COUNTING SOLUTIONS TO RANDOM CNF FORMULAS

ANDREAS GALANIS, LESLIE ANN GOLDBERG, HENG GUO, AND KUAN YANG

Abstract. We give the first efficient algorithm to approximately count the number of solutions in the random k-SAT model when the density of the formula scales exponentially with k. The best previous counting algorithm for the permissive version of the model was due to Montanari and Shah and was based on the correlation decay method, which works up to densities $(1+o_k(1))\frac{2\log k}{k}$, the Gibbs uniqueness threshold for the model. Instead, our algorithm harnesses a recent technique by Moitra to work for random formulas with much higher densities. The main challenge in our setting is to account for the presence of high-degree variables whose marginal distributions are hard to control and which cause significant correlations within the formula.

1. Introduction

Let $\Phi = \Phi(k, n, m)$ be a k-CNF formula on n Boolean variables with m clauses chosen uniformly at random where each clause has size $k \geq 3$. The random formula Φ shows an interesting threshold behaviour, where the asymptotic probability that Φ is satisfiable drops dramatically from 1 to 0 when the density $\alpha := m/n$ crosses a certain threshold $\alpha_\star$. There has been tremendous progress on establishing this phase transition and pinpointing the threshold $\alpha_\star$ [30, 23, 4, 5, 15, 18] guided by elaborate but non-rigorous methods in physics [33, 32]. The exact value of the threshold $\alpha_\star$ is established in [18] for sufficiently large k; it is known that $\alpha_\star = 2^k \ln 2 - \frac{1}{2}(1 + \ln 2) + o_k(1)$ as $k \to \infty$.

In contrast, the "average case" computational complexity of random k-CNF formulas remains elusive. It is a notoriously hard problem to design algorithms that succeed in finding a satisfying assignment when the density of the formula Φ is close to (but smaller than) the satisfiability threshold $\alpha_\star$. The best polynomial-time algorithm to find a satisfying assignment of Φ is due to Coja-Oghlan [10], which succeeds if $\alpha < (1 - o_k(1)) \cdot 2^k \ln k/k$. It is known that beyond this density bound $2^k \ln k/k$ the solution space of the formula undergoes a phase transition and becomes severely more complicated [2], so local algorithms are bound to fail to find a satisfying assignment in polynomial time (see for example [28, 11, 13]).

It is also a natural question to determine the number of satisfying assignments to Φ, denoted by $Z(\Phi)$, when the density is below the satisfying threshold. It has been shown that $\frac{1}{n}\log Z(\Phi)$ is concentrated around its expectation [1, 16] for $\alpha < (1 - o_k(1)) \cdot 2^k \ln k/k$. However, for the random k-SAT model, there is no known formula for the expectation $\mathbb{E}\frac{1}{n}\log Z(\Phi)$ (though see [3] and [40, 17] for progress along these lines for the case $k = 2$ and for more symmetric models of random formulas, respectively). Regarding the algorithmic question, Montanari and Shah [36] have given an efficient algorithm to approximate a closely related (permissive) version[1]

Date: 24th May 2021.

A preliminary short version of the manuscript (without proofs) appeared in the proceedings of ICALP 2020. The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013) ERC grant agreement no. 334828 and under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 947778). The paper reflects only the authors' views and not the views of the ERC or the European Commission. The European Union is not liable for any use that may be made of the information contained therein.

[1]More precisely, for the relevant densities α, [36] gives, for all reals $\beta > 0$, a PTAS for the partition function $\frac{1}{n}\log Z_\beta(\Phi)$ of a weighted model on all Boolean assignments σ, where the weight of an assignment σ is $e^{-\beta H(\sigma)}$ and $H(\sigma)$ is the number of unsatisfied clauses under σ. The number of satisfying assignments $Z(\Phi)$ corresponds to $Z_\beta(\Phi)$ when $\beta \to \infty$. Technically, the algorithm of [36] does not yield an algorithm for $\beta = \infty$, even though it applies to arbitrarily large β (in fact, even for $\beta \leq n^\delta$ for some small constant $\delta > 0$).

of $\frac{1}{n}\log Z(\Phi)$ if $\alpha \leq \frac{2\log k}{k}(1+o_k(1))$, based on the correlation decay method and the uniqueness threshold of the Gibbs distribution. Note that the threshold for the density α is exponentially lower than the satisfiability threshold, and no efficient algorithm was known to give a more precise approximation to $Z(\Phi)$ (rather than the log-partition function).

In this paper, we address the algorithmic counting problem by giving the first *fully polynomial-time approximation scheme* (FPTAS) for the number of satisfying assignments to random k-CNF formulas, if the density α is less than 2^{rk}, for sufficiently large k and some constant $r > 0$. Our bound is exponential in k and goes well beyond the uniqueness threshold of $\frac{2\log k}{k}(1+o_k(1))$ which is required by the correlation decay method.

Our result is related to other algorithmic counting results on random graphs such as counting colourings, independent sets, and other structures in random graphs, see for example [38, 19, 42, 20, 31, 8]. However, previous methods, such as Markov Chain Monte Carlo and Barvinok's method, appear to be difficult to apply to random formulas. Instead, our algorithm is the first adaptation of Moitra's method [35] to the random instance setting. We give a high level overview of the techniques in Section 1.3.

1.1. **The model and the main result.** For $k \geq 3$, let $\Phi = \Phi(k,n,m)$ denote a k-SAT formula chosen uniformly at random from the set of all k-SAT formulas with n variables and m clauses. Specifically, Φ has n variables $v_1, v_2, \ldots, v_n$ and m clauses $c_1, c_2, \ldots, c_m$. Each clause c_i has k literals $\ell_{i,1}, \ell_{i,2}, \ldots, \ell_{i,k}$ and each literal $\ell_{i,j}$ is chosen uniformly at random from $2n$ literals $\{v_1, v_2, \ldots, v_n, \neg v_1, \neg v_2, \ldots, \neg v_n\}$. Note that each clause has exactly k literals (repetitions allowed), so there are $(2n)^{km}$ possible formulas; we use $\Pr(\cdot)$ to denote the uniform distribution on the set of all such formulas. Throughout, we will assume that $m = \lfloor n\alpha \rfloor$, where $\alpha > 0$ is the density of the formula. We say that an event $\mathcal{E}$ holds *w.h.p.* if $\Pr(\mathcal{E}) = 1 - o(1)$ as $n \to \infty$.

For a k-SAT formula Φ, we let $\Omega = \Omega(\Phi)$ denote the set of satisfying assignments of Φ. If Ω is non-empty, we let π_Φ denote the uniform distribution on Ω.

Theorem 1. *There is a polynomial-time algorithm $\mathcal{A}$ and there are two constants $r > 0$ and $k_0 \geq 3$ such that, for all $k \geq k_0$ and all $\alpha < 2^{rk}$, the following holds w.h.p. over the choice of the random k-SAT formula $\Phi = \Phi(k,n,\lfloor \alpha n \rfloor)$. The algorithm $\mathcal{A}$, given as input the formula Φ and a rational $\varepsilon > 0$, outputs in time poly$(n, 1/\varepsilon)$ a number Z that satisfies $e^{-\varepsilon}|\Omega(\Phi)| \leq Z \leq e^{\varepsilon}|\Omega(\Phi)|$.*

Throughout this paper, we will assume that $k \geq k_0$ where k_0 is a sufficiently large constant. We will also assume that the density α of the formula Φ satisfies $\alpha < 2^{k/300}/k^3$, so r can be taken to be $1/301$ in Theorem 1. The constant 300 here is not optimised, but we do not expect to be able to use the current techniques to improve it substantially. Our main point is that for a density which is exponential in k, an FPTAS exists for random k-CNF formulas. Finally, we assume that $k^2\alpha \geq 1$, otherwise it is well-known (see, e.g., Theorem 3.6 in [39]) that w.h.p. every connected component of Φ, viewed as a hypergraph where variables correspond to vertices and clauses correspond to hyperedges, is of size $O(\log n)$. In this case we can count the number of satisfying assignments by brute force.

1.2. **New developments.** After this paper was written, Coja-Oghlan, Müller, and Ravelomanana [14] have given a conditional formula for the permissive version (analogously to [36], cf. Footnote 1) of the log-partition function $\mathbb{E}\frac{1}{n}\log Z(\Phi)$. Their condition captures a non-reconstruction property of the Gibbs distribution that is believed to hold up to densities $(1-o_k(1))2^k \ln k/k$ (though this is not known). As mentioned earlier, our algorithmic result applies to the non-permissive version and gives unconditional approximation guarantees for the partition function $Z(\Phi)$ with arbitrarily small relative error. For future work, it would be interesting to see whether the probabilistic approach of [14] can be combined with the algorithmic perspective of this paper.

1.3. **Algorithm overview.** We give a high-level overview of our algorithm here before giving the details. Approximately counting the satisfying assignments of a k-CNF formula has been a challenging problem using traditional algorithmic techniques, since the solution space (the set of satisfying assignments) is complicated and it is not connected, using the transitions of commonly-studied Markov chains. Recently some new approaches were introduced [35, 24]. Most notably, the breakthrough work of Moitra [35] gives the first (and so far the only) efficient deterministic algorithm that can approximately count the satisfying assignments of k-CNF formulas in which each variable appears in at most d clauses, if, roughly speaking, $d \lesssim 2^{k/60}$. Inspired by this, Feng et al. [22] have also given a MCMC algorithm which applies when $d \lesssim 2^{k/20}$. Subsequent to this paper, Jain et al. [29] improved this bound even further, showing how to approximately count when $d \lesssim 2^{k/5.741}$.

As our goal is to count satisfying assignments of sparse random k-CNF formulas, where these degree bounds do not hold, but average degrees are small, it is natural to also choose Moitra's method in the random instance setting. However, the first difficulty is that Moitra's method relies on the fact that the marginal probability of each variable (the probability that the variable is true in a uniformly-chosen satisfying assignment) is nearly $1/2$. This is necessary because Moitra's method involves solving a certain linear program (LP) and the size of this LP is polynomially-bounded only if a certain process couples quickly. The proof that the process couples quickly relies on the fact that the marginals are nearly $1/2$ (and certainly on the fact that they are bounded away from 0 and 1). In contrast, for a random k-CNF formula, although the *average* degree of variables is low, with high probability there are variables with degrees as high as $\Omega(\log n/\log\log n)$. In the presence of these high-degree variables, it is no longer true that the marginal probabilities of the variables are nearly $1/2$. In fact, they can be arbitrarily near 0 or 1.

Our solution to this issue is to separate out high-degree variables, as well as those that are heavily influenced by high-degree variables. To do this, we define a process to recursively label "bad" variables. At the start, all high-degree variables are bad. Then, all clauses containing more than $k/10$ bad variables are labelled bad, as are all variables that they contain. We run this process until no more bad clauses are found. We call the remaining variables and clauses of the formula "good". A key property is that all good variables have an upper bound on their degree and all good clauses contain at least $9k/10$ good variables; this allows us to show that the marginal probabilities of good variables are close to $1/2$.

The next step is to attempt to apply Moitra's method. The goal of Moitra's method is to compute more precise estimates for the marginal probabilities of variables; given accurate estimates on the marginal probabilities it is then relatively easy to approximate the number of satisfying assignments using refined self-reducibility techniques.

Of course, we need to modify the method to deal with the bad variables, which still appear in the formula. We first explain Moitra's method and then proceed with our modifications. The first step is to mark variables, so that every clause contains a good fraction of marked variables and a good fraction of unmarked variables. Then, for a particular marked variable v, we set up an LP. As noted earlier, the variables of the LP correspond to the states of a certain coupling process which couples two distributions on satisfying assignments using the marked variables — the first distribution over satisfying assignments in which v is true, and the second distribution over satisfying assignments in which v is false. Solving the LP recovers the transition probabilities of the coupling process and yields enough information to approximate the marginal probability of v.

In order to guarantee that the size of the LP is bounded by a polynomial in the size of the original CNF formula, we have to restrict the coupling process. The process can be viewed as a tree and it suffices to truncate this tree at a suitable level.

Thus, a crucial part of the proof (both in Moitra's case and in ours) is to show that the error caused by the truncation is sufficiently small. The reason that the error caused by the truncation is small is that, with high probability, branches of the coupling tree "die out" before reaching a large level. The reason for this is that the marginals of marked variables stay near $1/2$, even when conditioning on partial assignments.

In our case where Φ is a random formula, the marginals are not all near $1/2$, even without any conditioning. But the good variables do have marginals near $1/2$. So we only mark/unmark good variables and we "give up" on bad variables. Given that we don't have any control over the bad variables, we have to modify the coupling process. Thus, whenever we meet a bad variable in the coupling process, we have to assume the worst case and treat this variable and all bad variables connected to it as if they all have failed the coupling, meaning that the disagreement spreads quickly over bad components.

The most important part of our analysis is to upper bound the size of connected bad components and how often we encounter them during the coupling process. Given these upper bounds, we are able to show that the coupling still dies out sufficiently quickly, so the error caused by the truncation is not too large.

Solving the LP then allows us to estimate the marginals of the good variables. Given that the bad components have small size, this turns out to be enough information to allow us to estimate the number of satisfying assignments of the original formula (containing both good and bad clauses).

We conclude this summary by discussing the prospects for improving our work. Although we have given an efficient algorithm which works for densities that are exponentially large in k, the densities that we can handle are still small compared to the satisfiability threshold or to the threshold under which efficient search algorithms exist. Perhaps a modest start towards obtaining comparable thresholds for approximate counting algorithms would be to consider models whose state spaces are connected. For example, for monotone k-CNF formulas where each variable appears in at most d clauses, Hermon et al. [27] showed that efficient randomised algorithms exist if $d \leq c2^{k/2}$ for some constant $c > 0$, which is optimal up to the constant c due to complementing hardness results [7]. They also showed that the same algorithm works for *random regular* monotone k-CNF formulas, if the degree $d \leq c2^k/k$ for some $c > 0$. It remains open whether an average case bound of the same order can be achieved for random monotone k-CNF formulas.

2. Notation

To help keep track of the notation defined in this paper, the reader is referred to the table in Appendix (Section A).

3. High-degree and bad variables

We will apply the method of Moitra [35], which was introduced to approximately count the satisfying assignments of k-CNF formulas in which each literal appears a bounded number of times. The main difference between the formulas studied by Moitra and the random formulas that we study is that, in our formulas, some variables will occur many more times than the average.

Definition 2. *Let Φ be a k-SAT formula. We say that a variable v of Φ is* high-degree *if Φ contains at least $\Delta := 2^{k/300}$ occurrences of literals involving the variable v.*

In our algorithm, we will not be able to control these high degree variables or other variables that are affected by them. These variables contribute to the "bad" part of the formula Φ. Formally, denote the set of clauses of Φ by $\mathcal{C}$ and the set of variables by $\mathcal{V}$. For each $c \in \mathcal{C}$, let $\mathsf{var}(c)$ denote the set of variables in c. For each subset C of $\mathcal{C}$, let $\mathsf{var}(C) := \cup_{c\in C}\mathsf{var}(c)$. The *bad variables* and *bad clauses* of Φ are identified by running the following process:

(1) $\mathcal{V}_0$ (the initial bad variables) $\leftarrow$ the set of high-degree variables;
(2) $\mathcal{C}_0 \leftarrow$ the set of clauses with at least $k/10$ variables in $\mathcal{V}_0$;
(3) $i \leftarrow 0$;
(4) Do the following until $\mathcal{V}_i = \mathcal{V}_{i-1}$:
   - $i \leftarrow i + 1$;
   - $\mathcal{V}_i \leftarrow \mathcal{V}_{i-1} \cup \mathsf{var}(\mathcal{C}_{i-1})$;

- $\mathcal{C}_i \leftarrow \{c \in \mathcal{C} \mid \mathsf{var}(c) \cap \mathcal{V}_i \geq k/10\}$;

(5) $\mathcal{C}_{\text{bad}} \leftarrow \mathcal{C}_i$ and $\mathcal{V}_{\text{bad}} \leftarrow \mathcal{V}_i$;

(6) $\mathcal{C}_{\text{good}} \leftarrow \mathcal{C} \setminus \mathcal{C}_i$ and $\mathcal{V}_{\text{good}} \leftarrow \mathcal{V} \setminus \mathcal{V}_i$.

The method that we have used to define good variables and clauses is inspired by [15].

Observation 3. *For any $c \in \mathcal{C}_{good}$, $|\mathsf{var}(c) \cap \mathcal{V}_{bad}| < k/10$. For any $c \in \mathcal{C}_{bad}$, $\left|\mathsf{var}(c) \cap \mathcal{V}_{good}\right| = 0$.*

The formula Φ naturally corresponds to a bipartite "factor graph" where one side is variables and the other clauses. We will use the following two graphs G_Φ and H_Φ which are the dependency graph induced by the factor graph on clauses and variables, respectively.

Definition 4. *Let G_Φ be the graph with vertex set $\mathcal{C}$ in which two clauses c and c' are adjacent if and only if $\mathsf{var}(c) \cap \mathsf{var}(c') \neq \emptyset$. We say that a set $C \subseteq \mathcal{C}$ of clauses is connected if the induced subgraph $G_\Phi[C]$ is connected. Let $G_{\Phi,good}$ be the graph with vertex set $\mathcal{C}_{good}$ in which two clauses c and c' are adjacent if and only if $\mathsf{var}(c) \cap \mathsf{var}(c') \cap \mathcal{V}_{good} \neq \emptyset$, i.e., c and c' share a good variable.*

Since each $v \in \mathcal{V}_{\text{good}}$ has at most Δ literal occurrences, the maximum degree in $G_{\Phi,\text{good}}$ is at most $k(\Delta - 1)$.

Definition 5. *Let H_Φ be the graph with vertex set $\mathcal{V}$ in which two variables v and v' are adjacent if and only if there exists a clause $c \in \mathcal{C}$ such that $v, v' \in \mathsf{var}(c)$. We say that a set $V \subseteq \mathcal{V}$ of variables is connected if the induced subgraph $H_\Phi[V]$ is connected. Let $H_{\Phi,bad}$ be the graph with vertex set $\mathcal{V}_{bad}$ in which two variables v and v' are adjacent if and only if there exists a bad clause $c \in \mathcal{C}_{bad}$ such that $v, v' \in \mathsf{var}(c)$. We say that a set $V \subseteq \mathcal{V}$ of variables is a* bad component *if V is a connected component in $H_{\Phi,bad}$.*

4. The Lovász local lemma

The Lovász local lemma [21] is an important tool for our algorithm. In particular, we will need the asymmetric version (proved by Lovász and published in [41]). For convenience, we will specialize the local lemma to the variable setting.

Definition 6. *Let Ω^* be the set of all $2^{|\mathcal{V}|}$ assignments $\mathcal{V} \to \{\mathsf{T}, \mathsf{F}\}$. Given any subset $A \subseteq \Omega^*$, let μ_A be the uniform distribution on A. We say that a subset $A \subseteq \Omega^*$ depends on a variable $v \in \mathcal{V}$ if there exist $\sigma \in A$ and $\sigma' \in \Omega^* \setminus A$ such that σ and σ' differ only at v. We use $\mathsf{var}(A)$ to denote the set of variables on which A depends. Let $\Gamma(A) = \{c \in \mathcal{C} \mid \mathsf{var}(c) \cap \mathsf{var}(A) \neq \emptyset\}$ and $\Gamma_{good}(A) = \{c \in \mathcal{C} \mid \mathsf{var}(A) \cap \mathsf{var}(c) \cap \mathcal{V}_{good} \neq \emptyset\}$. Similarly, for any $c \in \mathcal{C}$, let $\Gamma(c) = \{c' \in \mathcal{C} \mid \mathsf{var}(c) \cap \mathsf{var}(c') \neq \emptyset\}$ and $\Gamma_{good}(c) = \{c' \in \mathcal{C} \mid \mathsf{var}(c) \cap \mathsf{var}(c') \cap \mathcal{V}_{good} \neq \emptyset\}$.*

When reading the following theorem, it is helpful to think of A_c as being the set of assignments that fail to satisfy the clause c (though we will also use the theorem in other ways).

Theorem 7 (The local lemma)**.** *For each $c \in \mathcal{C}$, let A_c be a subset of Ω^* such that $\mathsf{var}(A_c) \subseteq \mathsf{var}(c)$. If there exists a function $x \colon \mathcal{C} \to (0, 1)$ such that, for all $c \in \mathcal{C}$,*

$$\Pr_{\mu_{\Omega^*}}(A_c) \leq x(c) \prod_{b \in \Gamma(A_c)} \Big(1 - x(b)\Big), \tag{1}$$

then

$$\Pr_{\mu_{\Omega^*}}\Big(\bigwedge_{c \in \mathcal{C}} \overline{A_c}\Big) > 0.$$

Furthermore, for any $A \subseteq \Omega^*$, $\Pr_{\mu_{\Omega^*}}\Big(A \mid \bigwedge_{c \in \mathcal{C}} \overline{A_c}\Big) \leq \Pr_{\mu_{\Omega^*}}(A) \prod_{b \in \Gamma(A)} \Big(1 - x(b)\Big)^{-1}$.

The second part of Theorem 7 is due to Haeupler et al. [26]. It provides an upper bound on the probability of an event under the uniform distribution over satisfying assignments. It is also possible to find an assignment such that $\bigwedge_{c \in \mathcal{C}} \overline{A_c}$ holds in $O(|\mathcal{V}|)$ time by the algorithm of Moser and Tardos [37].

In the course of our algorithm it is useful to "mark" some of the variables in $\mathcal{V}_{\text{good}}$. For this, we use the approach of Moitra [35]. Formally, a "marking" is an assignment from $\mathcal{V}_{\text{good}}$ to $\{\text{marked}, \text{unmarked}\}$. All bad variables are unmarked. Using Theorem 7, we have the following lemma.

Lemma 8. *W.h.p. over the choice of Φ, there exists a marking on $\mathcal{V}_{good}$ such that:*

(1) every good clause has at least $3k/10$ marked variables and at least $k/4$ unmarked good variables;

(2) there is a partial assignment of bad (and thus unmarked) variables that satisfies all bad clauses.

Furthermore, such a marking can be found in deterministic polynomial time.

Proof. We apply Theorem 7 on $\mathcal{C}_{\text{good}}$ and $\mathcal{V}_{\text{good}}$. Let Ω^*_{good} be all possible markings: $\mathcal{V}_{\text{good}} \to \{\text{marked}, \text{unmarked}\}$, so $\mu_{\Omega^*_{\text{good}}}$ is the distribution in which each good variable is marked independently and uniformly at random. For $c \in \mathcal{C}_{\text{good}}$, let M be the number of marked good variables in $\mathsf{var}(c)$. Let A_c be the subset of Ω^*_{good} that $M < 3k/10$ or $M > 3k/5$. Since $c \in \mathcal{C}_{\text{good}}$, c contains at least $9k/10$ good variables (see the observation immediately following the process that defines bad variables and clauses). Thus $9k/20 \le \mathbb{E}_{\mu_{\Omega^*_{\text{good}}}} M \le k/2$, and the number of unmarked good variables in $\mathsf{var}(c)$ is at least $9k/10 - M$. By a Chernoff bound,

$$\begin{aligned}\Pr_{\mu_{\Omega^*_{\text{good}}}}(A_c) &= \Pr_{\mu_{\Omega^*_{\text{good}}}}(M > 3k/5) + \Pr_{\mu_{\Omega^*_{\text{good}}}}(M < 3k/10)\\ &\le e^{-k/110} + e^{-k/40} \le 2e^{-k/150}.\end{aligned}$$

Let $x(c) = \frac{1}{k\Delta}$ if $c \in \mathcal{C}_{\text{good}}$. Since we only consider assignments of $\mathcal{V}_{\text{good}}$, so $\mathsf{var}(A_c) \subseteq \mathsf{var}(c) \cap \mathcal{V}_{\text{good}}$ and thus $\Gamma(A_c) \subseteq \Gamma_{\text{good}}(c)$. Note that $\Gamma_{\text{good}}(c)$ is the set of neighbours of c in $G_{\Phi,\text{good}}$, and the maximum degree in $G_{\Phi,\text{good}}$ is at most $k(\Delta - 1)$, so we can verify (1) as follows, for any $c \in \mathcal{C}_{\text{good}}$ and sufficiently large k,

$$\begin{aligned}x(c) \prod_{b \in \Gamma(A_c)} \Big(1 - x(b)\Big) \ge x(c) \prod_{b \in \Gamma_{\text{good}}(c)} \Big(1 - x(b)\Big) &\ge \frac{1}{k\Delta}\left(1 - \frac{1}{k\Delta}\right)^{k\Delta} \ge \frac{1}{e^2 k\Delta}\\ &\ge e^{-k/300-2}/k > 2e^{-k/150} \ge \Pr_{\mu_{\Omega^*_{\text{good}}}}(A_c).\end{aligned}$$

Thus, there is a marking such that item (1) holds.

W.h.p. over the choice of Φ, item (2) always holds for any marking of $\mathcal{V}_{\text{good}}$. This is because our density is well below the critical threshold [18]. Thus there is at least one satisfying assignment to Φ. As $\mathsf{var}(c) \cap \mathcal{V}_{\text{good}} = \emptyset$ for any $c \in \mathcal{C}_{\text{bad}}$, the restriction of the satisfying assignment to $\mathcal{V}_{\text{bad}}$ satisfies the condition.

The marking can be found using the deterministic algorithm [9] by verifying

$$\Pr_{\mu_{\Omega^*_{\text{good}}}}(A_c) \le \left(x(c) \prod_{b \in \Gamma_{\text{good}}(c)} \Big(1 - x(b)\Big) \right)^{1.01} \qquad \square$$

We will assume from now on that the random formula Φ satisfies Lemma 8 and we will stick to an arbitrary marking given by Lemma 8. We use $\mathsf{marked}(c)$ to denote the marked variables in clause c and $\mathcal{V}_{\mathsf{marked}} := \cup_{c \in \mathcal{C}_{\text{good}}} \mathsf{marked}(c)$ be the set of all marked variables.

Let Ω be the set of satisfying assignments of Φ. We will be particularly interested in the uniform distribution $\mu_\Omega(\cdot)$. For any partial assignment Λ of some of the *good* variables of Φ, let Φ^Λ be the formula produced by simplifying Φ under Λ. In other words, we remove all clauses that are satisfied under Λ and we remove all false literals from all clauses. (Some clauses may become empty in Φ^Λ, in which case Φ^Λ cannot be satisfied.) We use $\mathcal{C}^\Lambda$ to denote the set of clauses of Φ^Λ, and similarly $\mathcal{V}^\Lambda$. We also define $\mathcal{V}^\Lambda_{\text{good}} = \mathcal{V}_{\text{good}} \cap \mathcal{V}^\Lambda$ and $\mathcal{C}^\Lambda_{\text{good}} = \mathcal{C}_{\text{good}} \cap \mathcal{C}^\Lambda$ to denote the sets of remaining good variables and clauses of Φ simplified under Λ. (Remark: Note that Λ does not contain bad variables so if we define $\mathcal{V}^\Lambda_{\text{bad}}$ and $\mathcal{C}^\Lambda_{\text{bad}}$ similarly then we will have

$\mathcal{V}_{\text{bad}}^{\Lambda} = \mathcal{V}_{\text{bad}}$ and $\mathcal{C}_{\text{bad}}^{\Lambda} = \mathcal{C}_{\text{bad}}$.) Let Ω^{Λ} denote the set of satisfying assignments of Φ^{Λ}, namely, those satisfying assignments of Φ consistent with Λ.

Let $s := 2^{k/4}/(ek\Delta)$. Under an arbitrary conditioning of marked variables, we have good control of events defined by good variables, and the marginal distribution of good variables.

Lemma 9. *Let $V \subseteq \mathcal{V}_{\mathsf{marked}}$. For any partial assignment $\Lambda : V \to \{\mathsf{T}, \mathsf{F}\}$ such that $\Omega^{\Lambda} \neq \emptyset$ and any subset $A \subseteq \Omega^{\Lambda}$ such that $\mathsf{var}(A) \subseteq \mathcal{V}_{good} \setminus V$,*

$$\Pr_{\mu_{\Omega^{\Lambda}}}(A) \leq \Pr_{\mu_{\mathcal{A}}}(A)\Big(1 - \frac{1}{sk\Delta}\Big)^{-|\Gamma(A)|},$$

where $\mathcal{A}$ is the set of all assignments of $\mathsf{var}(A)$. In particular, for $v \in \mathcal{V}_{\mathsf{marked}} \setminus V$,

$$\left(1 - \frac{1}{3s}\right)\frac{1}{2} \leq \Pr_{\mu_{\Omega^{\Lambda}}}(v \to \mathsf{T}) \leq \left(1 + \frac{1}{3s}\right)\frac{1}{2}.$$

Proof. Let Λ_{bad} be an arbitrary assignment on $\mathcal{V}_{\text{bad}}$ such that all bad clauses are satisfied. Such a Λ_{bad} exists because of Item (2) of Lemma 8.

We apply Theorem 7 to $\Phi^{\Lambda \cup \Lambda_{\text{bad}}}$. The remaining variables are $V' = \mathcal{V}_{\text{good}} \setminus V$ and clauses are $\mathcal{C}' := \mathcal{C}_{\text{good}}^{\Lambda \cup \Lambda_{\text{bad}}}$. Let Ω' be the set of all $2^{|V'|}$ assignments $V' \to \{\mathsf{T}, \mathsf{F}\}$. Let A_c be the subset of Ω' that c is not satisfied, for $c \in \mathcal{C}'$. Then,

$$\Pr_{\mu_{\Omega^{\Lambda}}}(\cdot \mid \Lambda_{\text{bad}}) = \Pr_{\mu_{\Omega'}}(\cdot \mid \bigwedge_{c \in \mathcal{C}'} \overline{A_c}). \tag{2}$$

We verify the condition (1) of Theorem 7 and show

$$\Pr_{\mu_{\Omega'}}(A \mid \bigwedge_{c \in \mathcal{C}'} \overline{A_c}) \leq \Pr_{\mu_{\mathcal{A}}}(A)\Big(1 - \frac{1}{sk\Delta}\Big)^{-|\Gamma(A)|}, \tag{3}$$

and

$$\left(1 - \frac{1}{3s}\right)\frac{1}{2} \leq \Pr_{\mu_{\Omega'}}(v \to \mathsf{T} \mid \bigwedge_{c \in \mathcal{C}'} \overline{A_c}) \leq \left(1 + \frac{1}{3s}\right)\frac{1}{2}. \tag{4}$$

The first part of the lemma follows directly and the second part follows from (2), (4), and the law of total probabilities over all choices of Λ_{bad}.

By Lemma 8, there are at least $k/4$ unmarked variables in each $c \in \mathcal{C}'$. Thus,

$$\Pr_{\mu_{\Omega'}}(A_c) \leq 2^{-k/4}.$$

Let $x(c) = \frac{1}{sk\Delta}$. Since Ω' is a set of assignments of good variables, we have $\mathsf{var}(A_c) \subseteq \mathsf{var}(c) \cap \mathcal{V}_{\text{good}}$, and thus $\Gamma(A_c) \subseteq \Gamma_{\text{good}}(c)$. Again, note that $\Gamma_{\text{good}}(c)$ is the set of neighbours of c in $G_{\Phi,\text{good}}$ and the maximum degree of $G_{\Phi,\text{good}}$ is at most $k(\Delta - 1)$, so we obtain

$$\begin{aligned} x(c) \prod_{b \in \Gamma(A_c)} \Big(1 - x(b)\Big) &\geq x(c) \prod_{b \in \Gamma_{\text{good}}(c)} \Big(1 - x(b)\Big) \geq \frac{1}{sk\Delta}\left(1 - \frac{1}{sk\Delta}\right)^{k(\Delta-1)} \\ &\geq \frac{e^{-1/s}}{sk\Delta} \geq \frac{1}{esk\Delta} = 2^{-k/4}. \end{aligned}$$

The two inequalities above verify condition (1) of Theorem 7. Thus, (3) follows directly since $\mathsf{var}(A) \subseteq V'$. Moreover, setting $A = v \to \mathsf{T}$, for sufficiently large k,

$$\Pr_{\mu_{\Omega'}}(v \to \mathsf{T} \mid \bigwedge_{c \in \mathcal{C}'} \overline{A_c}) \leq \frac{1}{2}\left(1 - \frac{1}{sk\Delta}\right)^{-\Delta} \leq \frac{1}{2} \cdot e^{\Delta/(sk\Delta - 1)} \leq \frac{1}{2}\left(1 + \frac{1}{3s}\right).$$

We get the same upper bound for the event $v \to \mathsf{F}$ by the same argument. The bound (4) follows by combining these two bounds. □

Moreover, we have the following lemma for a partial assignment that we will use to apply self-reducibility.

Lemma 10. *Let $\Phi = \Phi(k,n,m)$ and let $v_1, v_2, \ldots, v_n$ be the variables of Φ. In each clause, order the literals in the order induced by the indices of their variables. Then there is a partial assignment Λ^* of truth values to some subset of $\mathcal{V}_{\mathsf{marked}}$ with the property that every clause $c \in \mathcal{C}_{good}$ is satisfied by its first $k/20$ literals corresponding to marked variables. Moreover, Λ^* can be found in deterministic polynomial time.*

Proof. We replace every $c \in \mathcal{C}_{\text{good}}$ by its first $k/20$ literals corresponding to marked variables. Call the new clause c' and the new set $\mathcal{C}'_{\text{good}}$. This induces a new formula Φ' whose clause set is $\mathcal{C}'_{\text{good}}$ with no unmarked variables. We apply Theorem 7 to Φ'. Let A_c be the event that c is not satisfied where $c \in \mathcal{C}'_{\text{good}}$ and set $x(c) = \frac{1}{k\Delta}$ for all $c \in \mathcal{C}'_{\text{good}}$. Since Φ' is a smaller formula, the size of $\Gamma_{\text{good}}(c)$ is still at most $k(\Delta - 1)$. It is straightforward to verify that the condition (1) holds and Theorem 7 applies. Thus, the desired Λ^* exists.

To find Λ^*, once again we apply the deterministic algorithm for the local lemma [9]. $\square$

Recall Φ^Λ, which is Φ simplified under Λ. Under Λ^*, neither bad clauses nor bad variables will be removed. More importantly, we have the following corollary.

Corollary 11. *For any prefix Λ of Λ^* in Theorem 10, any remaining good clause c in Φ^Λ satisfies* $\mathsf{marked}(c) \geq k/4$.

Proof. Under Λ, for any $c \in \mathcal{C}_{\text{good}}$, either c has at least $k/20$ marked variables assigned, or c has at most $k/20$ marked variables assigned. In the first case, by Lemma 10, c is satisfied and thus is removed in Φ^Λ. In the second case, even if c is not satisfied, by Lemma 8, c has at least $(3/10 - 1/20)k = k/4$ marked variables unassigned. $\square$

5. The coupling tree

W.h.p. the random formul Φ satisfies Lemma 8 and Lemma 10 and from now on we focus on Φ for which this is true. We will use the marking from Lemma 8. Conditioned on a prefix Λ of the partial assignment Λ^* from Lemma 10, the main subroutine of our algorithm is to calculate the marginal probability of the next variable in Λ^*, say v^*. Let $\Omega_1^\Lambda = \Omega^{\Lambda \cup \{v^* \to \mathsf{T}\}}$ and $\Omega_2^\Lambda = \Omega^{\Lambda \cup \{v^* \to \mathsf{F}\}}$. Then our goal is to estimate $|\Omega_1^\Lambda|/|\Omega_2^\Lambda|$.

We will eventually set up a linear program to approximate $|\Omega_1^\Lambda|/|\Omega_2^\Lambda|$. Before introducing the linear program, we will define the so-called coupling tree, which represents a variable-by-variable greedy coupling process between $\mu_{\Omega_1^\Lambda}$ and $\mu_{\Omega_2^\Lambda}$. Denote by $\mathbb{T}^\Lambda$ the coupling tree. Each node ρ of $\mathbb{T}^\Lambda$ is a tuple $\rho = (\mathcal{A}_1, \mathcal{A}_2, V_I, V_{\text{set}}, \mathcal{C}_{\text{rem}}, \mathcal{F}, R)$ where

- V_I and V_{set} are subsets of $\mathcal{V}$;
- $\mathcal{C}_{\text{rem}}$ and $\mathcal{F}$ are subsets of $\mathcal{C}$;
- $\mathcal{A}_1$ and $\mathcal{A}_2$ are functions from V_{set} to $\{\mathsf{T}, \mathsf{F}\}$;
- R is a function from $\mathcal{C}$ to the subsets of $\{\mathsf{bad}, \mathsf{disagree}, 1, 2\}$.

We use the notation $\mathcal{A}_1(\rho)$, $\mathcal{A}_2(\rho)$, etc. to denote components of ρ. Note that the information in ρ is redundant, as some components can be deduced from others.

Intuitively, for a node ρ, V_{set} is the set of variables that we have tried to couple, and V_I is the set of variables that cause discrepancies, or for which we have given up. The set $\mathcal{C}_{\text{rem}}$ denotes remaining clauses, and $\mathcal{F}$ denotes "failed" clauses. The two partial assignments $\mathcal{A}_1$ and $\mathcal{A}_2$ assign truth values to the variables in V_{set}. Finally, the function $R(c)$ gives the reason why c is in $\mathcal{F}$. The possibilities are: (1) c is a bad clause; (2) $\mathcal{A}_1$ and $\mathcal{A}_2$ disagree on some variable in this clause; (3) c is not satisfied by the partial assignment $\mathcal{A}_1$ or $\mathcal{A}_2$ (or both). These reasons may not be mutually exclusive. If $c \notin \mathcal{F}$, then $R(c) = \emptyset$.

We will only consider a partial assignment Λ that is a prefix of Λ^* from Lemma 10. Thus, by Corollary 11, in any $c \in \mathcal{C}_{\text{good}}^\Lambda$, there are at least $k/4$ marked variables remaining.

We will (inductively) guarantee that every node $\rho = (\mathcal{A}_1, \mathcal{A}_2, V_I, V_{\text{set}}, \mathcal{C}_{\text{rem}}, \mathcal{F}, R)$ of the coupling tree satisfies the following properties:

(P1) $\{v^*\} \subseteq V_{\text{set}} \cap V_I$.

(P2) Every clause $c \in \mathcal{C}_{\text{rem}}$ satisfies one of the following: $\mathsf{var}(c) \subseteq V_I$, $\mathsf{var}(c) \subseteq \mathcal{V}^\Lambda \setminus V_I$, or $\mathsf{marked}(c) \setminus V_{\text{set}}$ is non-empty.
(P3) For any $c \in \mathcal{C}_{\text{rem}} \cap \mathcal{C}_{\text{bad}}$, $\mathsf{var}(c) \subseteq \mathcal{V}^\Lambda \setminus V_I$.
(P4) For any $c \in \mathcal{C}^\Lambda \setminus \mathcal{C}_{\text{rem}}$, at least one of the following is true:
- c is satisfied by both of the partial assignments $\mathcal{A}_1$ and $\mathcal{A}_2$;
- $\mathsf{var}(c) \subseteq V_I \cup V_{\text{set}}$.

(P5) $H_\Phi[V_I]$ is connected.
(P6) For any $u \in V_I$, $\exists c \in \mathcal{F}$ s.t. $u \in \mathsf{var}(c)$.
(P7) For any $u \in V_{\text{set}} \setminus V_I$, $\mathcal{A}_1(u) = \mathcal{A}_2(u)$.
(P8) $R(c)$ is non-empty if and only if $c \in \mathcal{F}$.
(P9)
 (P9.1) For any $c \in \mathcal{F}$, $\mathsf{bad} \in R(c)$ only if c is a bad clause;
 (P9.2) For any good $c \in \mathcal{F}$, $\mathsf{disagree} \in R(c)$ iff there is a variable $v \in \mathsf{var}(c) \cap V_I \cap V_{\text{set}}$ such that $\mathcal{A}_1(v) \neq \mathcal{A}_2(v)$;
 (P9.3) For any $c \in \mathcal{F}$ and $i \in \{1, 2\}$, $i \in R(c)$ only if the following three conditions hold: $\mathsf{var}(c) \subseteq V_I \cup V_{\text{set}}$, $\mathsf{marked}(c) \subseteq V_{\text{set}}$, and $\mathcal{A}_i$ does not satisfy c.
(P10) For any $c \in \mathcal{C}^\Lambda_{\text{good}}$, $\mathsf{marked}(c) \cap V_I \subseteq V_{\text{set}}$.

The root of the coupling tree is the node ρ^* with $V_{\text{set}}(\rho^*) = V_I(\rho^*) = \{v^*\}$, $\mathcal{C}_{\text{rem}}(\rho^*) = \mathcal{C}^\Lambda$, and $\mathcal{F}(\rho^*)$ assigned to the set of clauses containing v^*. The assignment $\mathcal{A}_1(\rho^*)$ sets v^* to T and the assignment $\mathcal{A}_2(\rho^*)$ sets v^* to F. The function R maps every clause in $\mathcal{F}(\rho^*)$ to $\{\mathsf{disagree}\}$ and every other clause to $\emptyset$.

It is straightforward to see that ρ^* satisfies the coupling-tree properties. Property (P3) follows from the fact that v^* is good, so it is not in a bad clause (Observation 3). To see Property (P2) note that each bad clause c has $\mathsf{var}(c) \subseteq \mathcal{V}^\Lambda \setminus V_I$ and each good clause has at least $k/4 > 1$ marked variables so $\mathsf{marked}(c) \setminus V_{\text{set}}(\rho)$ is non-empty. The rest are straightforward.

In order to ensure that the size of the coupling tree is bounded from above by a polynomial in the size of the formula Φ, we will set a truncation depth $L := C_0(3k^2\Delta)\lceil \log(n/\varepsilon)\rceil$ for some sufficiently large absolute constant C_0, where $n = |\mathcal{V}|$.

Definition 12 (leaf, truncating node)**.** *A node ρ of the coupling tree is* a leaf *if $|V_I(\rho)| \leq L$ and every $c \in \mathcal{C}_{rem}(\rho)$ has the property that $\mathsf{var}(c) \subseteq V_I(\rho)$ or $\mathsf{var}(c) \subseteq \mathcal{V}^\Lambda \setminus V_I(\rho)$. If $|V_I(\rho)| > L$, then ρ is a* truncating node*. We denote the set of leaves by $\mathcal{L}$, the set of truncating nodes by $\mathcal{T}$, and their union by $\mathcal{L}^* := \mathcal{L} \cup \mathcal{T}$.*

Suppose that ρ is not a leaf or a truncating node. Then we define its children as follows. Since ρ is not a leaf, there is a clause $c \in \mathcal{C}_{\text{rem}}(\rho)$ such that $\mathsf{var}(c) \cap V_I(\rho) \neq \emptyset$ and $\mathsf{var}(c) \cap (\mathcal{V}^\Lambda \setminus V_I(\rho)) \neq \emptyset$. By (P3), c must be a good clause. Let c be the first such clause. By (P2), let u be the first variable in $\mathsf{marked}(c) \setminus V_{\text{set}}(\rho)$. By (P10), $u \notin V_I(\rho)$. Since it is in $\mathsf{marked}(c)$, u is a good variable. We refer to c as the *"first clause of ρ"* and u as the *"first variable of ρ"*.

We now define the four children of ρ in the coupling tree. For each of the four pairs (τ_1, τ_2) where τ_1 and τ_2 are assignments from $\{u\}$ to $\{\mathsf{T}, \mathsf{F}\}$, we create a child ρ_{τ_1,τ_2} of ρ using Algorithm 1. The following lemma shows that the coupling tree properties are maintained.

Lemma 13. *If ρ satisfies the coupling tree properties, then so does ρ_{τ_1,τ_2}.*

Proof. Property (P1) holds trivially for all nodes of the coupling tree. Property (P2) holds because clauses not satisfying the conditions are removed in the while loop of Line 15 which is repeated after the loop in Line 23 until there are no further changes. Property (P3) holds because clauses not satisfying the conditions are removed in the while loop of Line 23. For Property (P4), if a clause is removed from $\mathcal{C}_{\text{rem}}$, then it is removed in Line 13, 21, or 27. The first case satisfies the first condition of (P4), and the other two satisfy the second condition of (P4). For Property (P5), since $H_\Phi[V_I(\rho)]$ is connected, we just need to verify that when V_I expands, the new vertices are connected with the old V_I. The set V_I expands in Line 6 and Line 20 and 25. All cases can be verified straightforwardly. Property (P6) holds because all variables are added to V_I in Line 6 and Line 20 and 25. In all three cases the corresponding

Algorithm 1 Create the child ρ_{τ_1,τ_2}

1: $V_{\text{set}}(\rho_{\tau_1,\tau_2}) \leftarrow V_{\text{set}}(\rho) \cup \{u\}$;
2: $\mathcal{A}_1(\rho_{\tau_1,\tau_2}) \leftarrow$ combine $\mathcal{A}_1(\rho)$ with τ_1;
3: $\mathcal{A}_2(\rho_{\tau_1,\tau_2}) \leftarrow$ combine $\mathcal{A}_2(\rho)$ with τ_2;
4: $(V_I, \mathcal{C}_{\text{rem}}, \mathcal{F}, R) \leftarrow (V_I(\rho), \mathcal{C}_{\text{rem}}(\rho), \mathcal{F}(\rho), R(\rho))$;
5: **if** $\tau_1(u) \neq \tau_2(u)$ **then**
6: $\quad V_I \leftarrow V_I \cup \{u\}$;
7: $\quad$ **for** $c' : u \in \mathsf{var}(c')$ **do**
8: $\quad\quad \mathcal{F} \leftarrow \mathcal{F} \cup \{c'\}$;
9: $\quad\quad R(c') \leftarrow R(c') \cup \{\mathsf{disagree}\}$;
10: $\quad$ **end for**
11: **end if**
12: **for** $c' \in \mathcal{C}_{\text{rem}}$ **s.t.** c' is satisfied by both $\mathcal{A}_1(\rho_{\tau_1,\tau_2})$ and $\mathcal{A}_2(\rho_{\tau_1,\tau_2})$ **do**
13: $\quad \mathcal{C}_{\text{rem}} \leftarrow \mathcal{C}_{\text{rem}} \setminus \{c'\}$;
14: **end for**
15: **while** $\exists c' \in \mathcal{C}_{\text{rem}}$ with $\mathsf{var}(c') \cap V_I \neq \emptyset$, $\mathsf{var}(c') \cap (\mathcal{V}^\Lambda \setminus V_I) \neq \emptyset$, and $\mathsf{marked}(c') \setminus V_{\text{set}}(\rho_{\tau_1,\tau_2}) = \emptyset$ **do**
16: $\quad$ (* For some $i \in \{1,2\}$, c' is not satisfied by $\mathcal{A}_i(\rho_{\tau_1,\tau_2})$. Otherwise, c'
17: $\quad$ would have been removed from $\mathcal{C}_{\text{rem}}$ in the for loop of Line 12. *)
18: $\quad \mathcal{F} \leftarrow \mathcal{F} \cup \{c'\}$;
19: $\quad R(c') \leftarrow R(c') \cup \{i \mid c'$ is not satisfied by $\mathcal{A}_i(\rho_{\tau_1,\tau_2})\}$;
20: $\quad V_I \leftarrow V_I \cup (\mathsf{var}(c') \setminus V_{\text{set}}(\rho_{\tau_1,\tau_2}))$;
21: $\quad \mathcal{C}_{\text{rem}} \leftarrow \mathcal{C}_{\text{rem}} \setminus \{c'\}$;
22: **end while**
23: **while** $\exists c' \in \mathcal{C}_{\text{rem}} \cap \mathcal{C}_{\text{bad}}$ with $\mathsf{var}(c') \cap V_I \neq \emptyset$ **do**
24: $\quad \mathcal{F} \leftarrow \mathcal{F} \cup \{c'\}$;
25: $\quad V_I \leftarrow V_I \cup (\mathsf{var}(c') \setminus V_{\text{set}}(\rho_{\tau_1,\tau_2}))$;
26: $\quad R(c') \leftarrow R(c') \cup \{\mathsf{bad}\}$;
27: $\quad \mathcal{C}_{\text{rem}} \leftarrow \mathcal{C}_{\text{rem}} \setminus \{c'\}$;
28: **end while**
29: **if** $\exists c' \in \mathcal{C}_{\text{rem}}$ with $\mathsf{var}(c') \cap V_I \neq \emptyset$, $\mathsf{var}(c') \cap (\mathcal{V}^\Lambda \setminus V_I) \neq \emptyset$, and $\mathsf{marked}(c') \setminus V_{\text{set}}(\rho_{\tau_1,\tau_2}) = \emptyset$ **then** Goto Line 15
30: **end if**
31: $(V_I(\rho_{\tau_1,\tau_2}), \mathcal{C}_{\text{rem}}(\rho_{\tau_1,\tau_2}), \mathcal{F}(\rho_{\tau_1,\tau_2}), R(\rho_{\tau_1,\tau_2})) \leftarrow (V_I, \mathcal{C}_{\text{rem}}, \mathcal{F}, R)$;

clauses are added to $\mathcal{F}$. For Property (P7), V_{set} is only expanded in Line 1 and the property is guaranteed by Line 6. Property (P8) follows from the way that R and $\mathcal{F}$ are updated by the algorithm.

Property (P9.1) follows from the fact that the loop in Line 23 only applies to bad clauses. To see Property (P9.2), assume first that $c \in \mathcal{F}$ has $\mathsf{disagree} \in R(c)$. This must happen in Line 9 and the loop guarantees that there is a variable $u \in \mathsf{var}(c) \cap V_I \cap V_{\text{set}}$ such that $\mathcal{A}_1(u) \neq \mathcal{A}_2(u)$. For the other direction consider a good clause c' that is added to $\mathcal{F}$ in Step 18. All of the marked variables in c' are in $V_{\text{set}}(\rho_{\tau_1,\tau_2})$ so if there were a variable $v \in \mathsf{var}(c') \cap V_{\text{set}}(\rho_{\tau_1,\tau_2})$ such that $\mathcal{A}_1(\rho_{\tau_1,\tau_2})$ and $\mathcal{A}_2(\rho_{\tau_1,\tau_2})$ disagree on v, then at the time that v was set (either in ρ_{τ_1,τ_2} or in a parent) $\mathsf{disagree}$ would have been added to $R(c')$. Property (P9.3) follows by considering the loop in Line 15.

For Property (P10) note that marked vertices can only be added to V_I in Lines 6 and Line 20. The vertex added in Line 6 is also in V_{set}. The marked vertices that are added in Line 20 are also all in V_{set}. $\square$

5.1. **Key property of the coupling tree for a random formula.** The following property will be useful. Its proof is deferred to Section 8.5.

Lemma 14. *W.h.p. over the choice of Φ, for every prefix Λ of Λ^*, every node ρ in $\mathbb{T}^\Lambda$ has the property that $|V_{set}(\rho)| \leq 3k^3\alpha L + 1$.*

Remark. *The number of nodes in the coupling tree is a polynomial in n since the depth of the tree does not exceed $\max_{\rho\in\mathbb{T}^\Lambda} |V_{set}(\rho)| \leq 3k^3\alpha L + 1 = O(\log \frac{n}{\varepsilon})$ and each node has at most 4 children.*

6. The linear program

Before introducing the linear program which we will use to estimate $|\Omega_1^\Lambda|/|\Omega_2^\Lambda|$, we define one more piece of notation. In particular, for each node ρ of the coupling tree we define a quantity $r(\rho)$ as follows.

Let ρ be a node of the coupling tree. Let $\mathcal{C}_I(\rho)$ be the set of clauses $c \in \mathcal{C}^\Lambda$ such that $\mathsf{var}(c) \subseteq V_I(\rho) \cup V_{\text{set}}(\rho)$. For $i \in \{1,2\}$, let $N_i(\rho)$ be the number of assignments τ to $V_I(\rho)\setminus V_{\text{set}}(\rho)$ such that every clause in $\mathcal{C}_I(\rho)$ is satisfied by $\tau \cup \mathcal{A}_i(\rho)$.

We will use the following lemma.

Lemma 15. *If ρ is a node in the coupling tree, then $N_i(\rho) \neq 0$ for any $i \in \{1,2\}$.*

Proof. Since our coupling tree is based on the marking from Lemma 8, there is a partial assignment of bad variables that satisfies all $c \in \mathcal{C}_{\text{bad}}$. Let τ_{bad} be such an assignment and let $\tau_{\text{bad}}(\rho)$ be the restriction of τ_{bad} to the set $(V_I(\rho) \setminus V_{\text{set}}(\rho)) \cap \mathcal{V}_{\text{bad}}^\Lambda$. Note that $V_{\text{set}}(\rho) \subseteq \mathcal{V}_{\text{good}}^\Lambda$, so (by Observation 3) any bad clause $c \in \mathcal{C}_I(\rho)$ has the property that $\mathsf{var}(c) \subseteq (V_I(\rho) \setminus V_{\text{set}}(\rho)) \cap \mathcal{V}_{\text{bad}}$, which implies that $\tau_{\text{bad}}(\rho)$ satisfies all bad clauses in $\mathcal{C}_I(\rho)$.

Next we claim that there is an partial assignment $\tau_{\text{good}}(\rho) : (V_I(\rho) \setminus V_{\text{set}}(\rho)) \cap \mathcal{V}_{\text{good}}^\Lambda \to \{\mathsf{T}, \mathsf{F}\}$ such that $\tau_{\text{good}}(\rho)$ satisfies all good clauses in $\mathcal{C}_I(\rho)$. Let c be a good clause in $\mathcal{C}_I(\rho)$. Again by Lemma 8, c has at least $k/4$ unmarked good variables. Note that $\mathsf{var}(c) \subseteq V_I(\rho) \cup V_{\text{set}}(\rho)$ and $V_{\text{set}}(\rho)$ consists only of marked variables, so $\left|\mathsf{var}(c) \cap (V_I(\rho) \setminus V_{\text{set}}(\rho)) \cap \mathcal{V}_{\text{good}}^\Lambda\right| \geq k/4$. Denote by A_c the set of assignments in Ω^* for which the restriction on $(V_I(\rho)\setminus V_{\text{set}}(\rho))\cap\mathcal{V}_{\text{good}}^\Lambda$ does not satisfy c. Thus, $\Pr_{\mu_{\Omega^*}}(A_c) \leq 2^{-k/4}$. Also, by the definition of A_c, we obtain that $\mathsf{var}(A_c) \subseteq \mathsf{var}(c)$ and $\mathsf{var}(A_c) \subseteq \mathcal{V}_{\text{good}}^\Lambda$. So $\Gamma(A_c) \subseteq \Gamma_{\text{good}}(c)$. Let $x(c) = \frac{1}{k\Delta}$. Since $\Gamma_{\text{good}}(c)$ is the set of neighbours of c in $G_{\Phi,\text{good}}$, and the maximum degree in $G_{\Phi,\text{good}}$ is at most $k(\Delta-1)$, we conclude our claim by applying Theorem 7 and verifying

$$x(c) \prod_{b\in\Gamma(A_c)} \Big(1 - x(b)\Big) \geq \frac{1}{k\Delta}\left(1 - \frac{1}{k\Delta}\right)^{k\Delta} \geq \frac{1}{e^2 k\Delta} > 2^{-k/4} \geq \Pr_{\mu_{\Omega^*}}(A_c).$$

Now let $\tau = \tau_{\text{good}}(\rho) \cup \tau_{\text{bad}}(\rho)$. Then every clause in $\mathcal{C}_I(\rho)$ is satisfied by τ, which yields that $N_i(\rho) > 0$. ☐

Define $r(\rho) := N_1(\rho)/N_2(\rho)$. Lemma 15 implies that $r(\rho)$ is always well-defined.

Observation 16. *The quantity $r(\rho)$ can be computed in $m \cdot 2^{O(|V_I(\rho)\setminus V_{set}(\rho)|)}$ time by considering all assignments τ to $V_I(\rho)\setminus V_{set}(\rho)$. If ρ is a leaf then $|V_I(\rho)| \leq L$, so the time taken is polynomial in n/ε.*

The importance of $r(\rho)$ comes from the following lemma.

Lemma 17. *If ρ is a leaf, then $r(\rho) = |\Omega^{\mathcal{A}_1(\rho)\cup\Lambda}|/|\Omega^{\mathcal{A}_2(\rho)\cup\Lambda}|$.*

Proof. Suppose $\rho = (\mathcal{A}_1, \mathcal{A}_2, V_I, V_{\text{set}}, \mathcal{C}_{\text{rem}}, \mathcal{F}, R)$. Since ρ is a leaf, by Definition 12, for any $c \in \mathcal{C}_{\text{rem}}$, either $\mathsf{var}(c) \subseteq V_I \cup V_{\text{set}}$ (i.e., $c \in \mathcal{C}_I(\rho)$), or $\mathsf{var}(c) \subseteq \mathcal{V}^\Lambda \setminus V_I$ (or both). Denote by $\mathcal{C}_O(\rho)$ the set of clauses in $\mathcal{C}_{\text{rem}} \setminus \mathcal{C}_I(\rho)$ such that $\mathsf{var}(c) \subseteq \mathcal{V}^\Lambda \setminus V_I$. Denote by $\mathcal{C}_{\text{other}}(\rho)$ the set of clauses in $\mathcal{C}^\Lambda \setminus (\mathcal{C}_{\text{rem}} \cup \mathcal{C}_I(\rho))$. Thus, the clauses in $\mathcal{C}^\Lambda$ split into the disjoint sets $\mathcal{C}_I(\rho)$, $\mathcal{C}_O(\rho)$ and $\mathcal{C}_{\text{other}}(\rho)$.

By (P4), every clause in $\mathcal{C}_{\text{other}}(\rho)$ is satisfied by $\mathcal{A}_1(\rho)$ and $\mathcal{A}_2(\rho)$.

Let $M(\rho)$ be the number of assignments $\sigma\colon \mathcal{V}^\Lambda \setminus (V_I \cup V_{\text{set}}) \to \{\mathsf{T},\mathsf{F}\}$ such that all clauses in $\mathcal{C}_O$ are satisfied by $\sigma \cup \mathcal{A}_1(\rho)$ or equivalently, by (P7), satisfied by $\sigma \cup \mathcal{A}_2(\rho)$. Recall that $N_i(\rho)$ is the number of assignments τ to $V_I(\rho) \setminus V_{\text{set}}(\rho)$ such that every clause in $\mathcal{C}_I(\rho)$ is satisfied by $\tau \cup \mathcal{A}_i(\rho)$. Then for $i \in \{1,2\}$, $|\Omega^{\mathcal{A}_i(\rho)\cup\Lambda}| = M(\rho)N_i(\rho)$, which implies the lemma, since $r(\rho) = N_1(\rho)/N_2(\rho)$. ☐

We will use a binary search to approximate the quantity $\left|\Omega_1^\Lambda\right| / \left|\Omega_2^\Lambda\right|$. Our linear program relies on two constants r_{lower} and r_{upper}. We will move these closer and closer together by binary search. For each node ρ of the coupling tree, we introduce two variables $P_{1,\rho}$ and $P_{2,\rho}$. The idea is that a solution of the LP should have the property that

$$\frac{\left|\Omega_1^\Lambda\right|}{\left|\Omega_2^\Lambda\right|} = \frac{P_{1,\rho}}{P_{2,\rho}} \cdot \frac{\left|\Omega^{\mathcal{A}_1(\rho)\cup\Lambda}\right|}{\left|\Omega^{\mathcal{A}_2(\rho)\cup\Lambda}\right|}.$$

We now introduce the constraint sets of the LP.

Constraint Set 0. For every node ρ of the coupling tree and every $i \in \{1,2\}$ we add the constraint $0 \le P_{i,\rho} \le 1$.

Constraint Set 1. If $\rho \in \mathcal{L}$ then we add the following constraints.

$$r_{\mathsf{lower}}\, P_{2,\rho} \le P_{1,\rho}\, r(\rho)$$
$$P_{1,\rho}\, r(\rho) \le r_{\mathsf{upper}}\, P_{2,\rho}$$

Remark. *The purpose of these constraints is to guarantee*

$$r_{\mathsf{lower}} \le \frac{P_{1,\rho}}{P_{2,\rho}}\, r(\rho) \le r_{\mathsf{upper}}.$$

Constraint Set 2. For the root ρ^* of the coupling tree, we add the constraints

$$P_{1,\rho^*} = P_{2,\rho^*} = 1.$$

For every node ρ of the coupling tree that is not in $\mathcal{L}^*$, let u be the first variable of ρ. Add constraints as follows. For each $X \in \{\mathsf{T},\mathsf{F}\}$ add the following constraints.

$$P_{1,\rho} = P_{1,\rho_{u\to X, u\to \mathsf{T}}} + P_{1,\rho_{u\to X, u\to \mathsf{F}}}$$
$$P_{2,\rho} = P_{2,\rho_{u\to \mathsf{T}, u\to X}} + P_{2,\rho_{u\to \mathsf{F}, u\to X}}$$

These constraints imply the following lemma.

Lemma 18. *Suppose that the LP variables satisfy all of the constraints in Constraint Set 2. Then for any $i \in \{1,2\}$ and any $\sigma \in \Omega_i^\Lambda$,*

$$\sum_{\rho \in \mathcal{L}^* : \sigma \in \Omega^{\mathcal{A}_i(\rho)\cup\Lambda}} P_{i,\rho} = 1$$

Proof. For $i \in \{1,2\}$, we will maintain a set Ψ_i of nodes in the coupling tree with the invariant that $\sum_{\rho\in\Psi_i} P_{i,\rho} = 1$ and every node $\rho \in \Psi_i$ has $\mathcal{A}_i(\rho)$ agree with σ. For the base case, we let ρ^* be the root of the coupling tree and we take $\Psi_i = \{\rho^*\}$. If every node in Ψ_i is in $\mathcal{L}^*$ then we are finished. Otherwise, let ρ be some node in Ψ_i that is not in $\mathcal{L}^*$. Let u be the first variable of ρ. Let ρ' and ρ'' be the two children of ρ such that $\mathcal{A}_i(\rho')$ and $\mathcal{A}_i(\rho'')$ both map u to $\sigma(u)$. Replace ρ in Ψ_i with ρ' and ρ''. The constraints guarantee that $P_{i,\rho} = P_{i,\rho'} + P_{i,\rho''}$, so the invariant is maintained. ☐

Constraint Set 3. For every node ρ of the coupling tree that is not in $\mathcal{L}^*$, every $X \in \{\mathsf{T},\mathsf{F}\}$, and every $i \in \{1,2\}$, let u be the first variable of ρ and add the constraint $P_{i,\rho_{u\to X, u\to\neg X}} \le \frac{1}{s}\, P_{i,\rho}$.

The intuition behind this set of constraints is Lemma 9.

7. Analysis of the linear program

In Section 7.1, we show that whenever $r_{\mathsf{lower}} \le |\Omega_1^\Lambda|/|\Omega_2^\Lambda| \le r_{\mathsf{upper}}$, a solution to the LP exists. We call this "completeness" of the LP. In the remaining subsections of this section, we show "soundness" — namely, that whenever a solution exists, r_{lower} and r_{upper} are valid bounds for the quantity $|\Omega_1^\Lambda|/|\Omega_2^\Lambda|$, up to small errors.

7.1. Completeness.

Recall that we use the marking from Lemma 8 and that Λ is a prefix of the partial assignment Λ^* from Lemma 10.

We use the following lemma.

Lemma 19. *If ρ is a node in the coupling tree then, for any $i \in \{1,2\}$, $\Omega^{\mathcal{A}_i(\rho)\cup\Lambda}$ is non-empty.*

Proof. Let ρ be a node in the coupling tree and fix $i \in \{1,2\}$. Since our coupling tree is based on the marking from Lemma 8, there is a partial assignment of bad variables that satisfies all $c \in \mathcal{C}_{\mathrm{bad}}$. Let τ_{bad} be such an assignment. Note that $\mathcal{A}_i(\rho)\cup\Lambda$ is a partial assignment of marked (good) variables so it does not assign any variables in common with τ_{bad}. Let V' be the set of unmarked good variables. It remains to show that there is a partial assignment of variables in V' that satisfies all clauses in $\mathcal{C}_{\mathrm{good}}$. To do this we apply Theorem 7 to $\mathcal{V}_{\mathrm{good}}$ and $\mathcal{C}_{\mathrm{good}}$ in the same way as the proof of Lemma 9. □

Let ρ be a node of the coupling tree with first variable u. For $X \in \{\mathsf{T},\mathsf{F}\}$, we use the notation

$$\psi_{\rho,X,1} := \frac{|\Omega^{\mathcal{A}_1(\rho_{u\to X,u\to X})\cup\Lambda}|}{|\Omega^{\mathcal{A}_1(\rho)\cup\Lambda}|} = \frac{|\Omega^{\mathcal{A}_1(\rho_{u\to X,u\to\neg X})\cup\Lambda}|}{|\Omega^{\mathcal{A}_1(\rho)\cup\Lambda}|}. \tag{5}$$

This is well-defined since $\mathcal{A}_1(\rho_{u\to X,u\to X}) = \mathcal{A}_1(\rho_{u\to X,u\to\neg X})$. In other words, $\psi_{\rho,X,1}$ is the probability that u is assigned value X under $\mu_{\Omega^{\mathcal{A}_1(\rho)\cup\Lambda}}$. Thus, $\psi_{\rho,X,1} + \psi_{\rho,\neg X,1} = 1$. We similarly define

$$\psi_{\rho,X,2} := \frac{|\Omega^{\mathcal{A}_2(\rho_{u\to X,u\to X})\cup\Lambda}|}{|\Omega^{\mathcal{A}_2(\rho)\cup\Lambda}|} = \frac{|\Omega^{\mathcal{A}_2(\rho_{u\to\neg X,u\to X})\cup\Lambda}|}{|\Omega^{\mathcal{A}_2(\rho)\cup\Lambda}|},$$

noting that $\psi_{\rho,X,2} + \psi_{\rho,\neg X,2} = 1$.

We will next give an inductive definition of a function Q from nodes of the coupling tree to real numbers in $[0,1]$. The way to think about this is as follows — we will implicitly define a probability distribution over paths from the root of the coupling tree to $\mathcal{L}^*$. For each node ρ, $Q(\rho)$ will be the probability that ρ is included in a path drawn from this distribution.

Any such path starts at the root, so we define $Q(\rho^*) = 1$. Once we have defined $Q(\rho)$ for a node ρ that is not in $\mathcal{L}^*$ we can define $Q(\cdot)$ on the children of ρ as follows. Let u be the first variable of ρ and consider the four children $\rho_{u\to\mathsf{T},u\to\mathsf{T}}, \rho_{u\to\mathsf{T},u\to\mathsf{F}}, \rho_{u\to\mathsf{F},u\to\mathsf{T}}, \rho_{u\to\mathsf{F},u\to\mathsf{F}}$. Define the values of Q as follows.

$$\begin{aligned}
Q(\rho_{u\to\mathsf{T},u\to\mathsf{T}}) &:= Q(\rho)\min\{\psi_{\rho,\mathsf{T},1},\psi_{\rho,\mathsf{T},2}\}.\\
Q(\rho_{u\to\mathsf{T},u\to\mathsf{F}}) &:= Q(\rho)(\psi_{\rho,\mathsf{T},1} - \min\{\psi_{\rho,\mathsf{T},1},\psi_{\rho,\mathsf{T},2}\}).\\
Q(\rho_{u\to\mathsf{F},u\to\mathsf{F}}) &:= Q(\rho)\min\{1-\psi_{\rho,\mathsf{T},1},1-\psi_{\rho,\mathsf{T},2}\}.\\
Q(\rho_{u\to\mathsf{F},u\to\mathsf{T}}) &:= Q(\rho)((1-\psi_{\rho,\mathsf{T},1}) - \min\{1-\psi_{\rho,\mathsf{T},1},1-\psi_{\rho,\mathsf{T},2}\}).
\end{aligned} \tag{6}$$

Observation 20. *For any $X \in \{\mathsf{T},\mathsf{F}\}$,*

$$\begin{aligned}
Q(\rho_{u\to X,u\to\mathsf{T}}) + Q(\rho_{u\to X,u\to\mathsf{F}}) &= Q(\rho)\psi_{\rho,X,1}, \text{and}\\
Q(\rho_{u\to\mathsf{T},u\to X}) + Q(\rho_{u\to\mathsf{F},u\to X}) &= Q(\rho)\psi_{\rho,X,2}.
\end{aligned}$$

Also,

$$Q(\rho_{u\to\mathsf{T},u\to\mathsf{T}}) + Q(\rho_{u\to\mathsf{T},u\to\mathsf{F}}) + Q(\rho_{u\to\mathsf{F},u\to\mathsf{T}}) + Q(\rho_{u\to\mathsf{F},u\to\mathsf{F}}) = Q(\rho).$$

This implies that $\sum_{\rho\in\mathcal{L}^*} Q(\rho) = 1$.

We use Q to define a feasible solution to the LP. Indeed, this definition explains what the variables in the LP were meant to represent.

Definition 21 (The LP variables)**.** *For each node ρ of the coupling tree and each $i \in \{1,2\}$, define $P_{i,\rho} := Q(\rho)|\Omega_i^\Lambda|/|\Omega^{\mathcal{A}_i(\rho)\cup\Lambda}|$.*

Theorem 19 ensures the values in Definition 21 are well defined. Motivated by Definition 21, we give the following upper bound for $Q(\rho)$.

Lemma 22. *For all nodes ρ in the coupling tree and all $i \in \{1,2\}$, $Q(\rho) \le |\Omega^{\mathcal{A}_i(\rho)\cup\Lambda}|/|\Omega_i^\Lambda|$.*

Proof. The proof is by induction — having established the claim for a node ρ, we then establish it for the children of ρ using Observation 20. For the base case, $Q(\rho^*) = 1 = |\Omega^{\mathcal{A}_i(\rho^*)\cup\Lambda}|/|\Omega_i^\Lambda|$. For the inductive step, consider a node ρ (for which the claim is established) and let u be the first variable of ρ. Consider any child $\rho_{u\to X,u\to Y}$ of ρ. We show the lemma for $i = 1$ and the other case is similar. By Observation 20, Equation (5), and the induction hypothesis,

$$\begin{aligned} Q(\rho_{u\to X,u\to Y}) &\le Q(\rho)\cdot\psi_{\rho,X,1} = Q(\rho)\cdot\frac{\left|\Omega^{\mathcal{A}_1(\rho_{u\to X,u\to Y})\cup\Lambda}\right|}{\left|\Omega^{\mathcal{A}_1(\rho)\cup\Lambda}\right|} \\ &\le \frac{\left|\Omega^{\mathcal{A}_1(\rho)\cup\Lambda}\right|}{\left|\Omega_i^\Lambda\right|}\cdot\frac{\left|\Omega^{\mathcal{A}_1(\rho_{u\to X,u\to Y})\cup\Lambda}\right|}{\left|\Omega^{\mathcal{A}_1(\rho)\cup\Lambda}\right|} = \frac{\left|\Omega^{\mathcal{A}_1(\rho_{u\to X,u\to Y})\cup\Lambda}\right|}{\left|\Omega_i^\Lambda\right|}. \end{aligned}$$ □

The next lemma relates the values of the LP variables, as defined by Definition 21.

Lemma 23. *Let $\rho \notin \mathcal{L}^*$ be a node in the coupling tree. Let u be the first variable of ρ. For any $X \in \{\mathsf{T},\mathsf{F}\}$ and $i \in \{1,2\}$, $P_{i,\rho_{u\to X,u\to\neg X}} \le P_{i,\rho}/s$.*

Proof. We first assume $X = \mathsf{T}$ and $i = 1$. Definition 21 implies that

$$\frac{P_{i,\rho_{u\to\mathsf{T},u\to\mathsf{F}}}}{P_{i,\rho}} = \frac{Q(\rho_{u\to\mathsf{T},u\to\mathsf{F}})}{Q(\rho)}\,\frac{\left|\Omega^{\mathcal{A}_1(\rho)\cup\Lambda}\right|}{\left|\Omega^{\mathcal{A}_1(\rho_{u\to\mathsf{T},u\to\mathsf{F}})\cup\Lambda}\right|}. \tag{7}$$

From the definition (6) (for the case where $\psi_{\rho,T,2} \le \psi_{\rho,T,1}$) along with Observation 20 (for the other case), we have that

$$\frac{Q(\rho_{u\to\mathsf{T},u\to\mathsf{F}})}{Q(\rho)} \le |\psi_{\rho,\mathsf{T},1} - \psi_{\rho,\mathsf{T},2}|.$$

Recall that $\psi_{\rho,\mathsf{T},i}$ is the probability that u is assigned the value T under $\mu_{\Omega^{\mathcal{A}_i(\rho)\cup\Lambda}}$. By Lemma 19, $\Omega^{\mathcal{A}_i(\rho)\cup\Lambda}$ is non empty, so we can apply Lemma 9 to this partial assignment. From the second part of the lemma, we have

$$\frac{1}{2}\left(1-\frac{1}{3s}\right) \le \psi_{\rho,\mathsf{T},i} \le \frac{1}{2}\left(1+\frac{1}{3s}\right).$$

Since k is sufficiently large and so is s,

$$|\psi_{\rho,\mathsf{T},1} - \psi_{\rho,\mathsf{T},2}| \le \frac{1}{s}\cdot\psi_{\rho,\mathsf{T},i}.$$

The claim in the lemma follows for $X = \mathsf{T}$ and $i = 1$ since $\psi_{\rho,\mathsf{T},1} = \frac{\left|\Omega^{\mathcal{A}_1(\rho_{u\to\mathsf{T},u\to\mathsf{F}})\cup\Lambda}\right|}{\left|\Omega^{\mathcal{A}_1(\rho)\cup\Lambda}\right|}$.

For $X = \mathsf{F}$ and $i = 1$, note that

$$\frac{Q(\rho_{u\to\mathsf{F},u\to\mathsf{T}})}{Q(\rho)} \le |(1-\psi_{\rho,\mathsf{T},1}) - (1-\psi_{\rho,\mathsf{T},2})| = |\psi_{\rho,\mathsf{T},1} - \psi_{\rho,\mathsf{T},2}|,$$

and use $\psi_{\rho,\mathsf{F},1} = \frac{\left|\Omega^{\mathcal{A}_1(\rho_{u\to\mathsf{F},u\to\mathsf{T}})\cup\Lambda}\right|}{\left|\Omega^{\mathcal{A}_1(\rho)\cup\Lambda}\right|}$ in the end.

For $i = 2$, the proof is similar. □

Now we are ready to show the completeness.

Lemma 24. *Suppose $r_{\mathsf{lower}} \le |\Omega_1^\Lambda|/|\Omega_2^\Lambda| \le r_{\mathsf{upper}}$. The variables $\mathbf{P} = \{P_{i,\rho}\}$ defined in Definition 21 satisfy all constraints of the LP.*

Proof. **Constraint Set 0**: The fact that the LP variables satisfy these constraints follows from the definition of the variables in Definition 21 (which guarantees that they are all non-negative) and from Lemma 22.

Constraint Set 1: Definition 21 implies that for any node ρ in the coupling tree,

$$\frac{P_{1,\rho}}{P_{2,\rho}} \frac{|\Omega^{\mathcal{A}_1(\rho)\cup\Lambda}|}{|\Omega^{\mathcal{A}_2(\rho)\cup\Lambda}|} = \frac{|\Omega_1^\Lambda|}{|\Omega_2^\Lambda|}. \tag{8}$$

If ρ is a leaf, then by Lemma 17, $r(\rho) = |\Omega^{\mathcal{A}_1(\rho)\cup\Lambda}|/|\Omega^{\mathcal{A}_2(\rho)\cup\Lambda}|$. Equation (8) implies that

$$\frac{P_{1,\rho}}{P_{2,\rho}} r(\rho) = \frac{|\Omega_1^\Lambda|}{|\Omega_2^\Lambda|},$$

so as long as $r_{\mathsf{lower}} \le |\Omega_1^\Lambda|/|\Omega_2^\Lambda| \le r_{\mathsf{upper}}$, the LP variables satisfy the constraints in **Constraint Set 1**, as required.

Constraint Set 2: For the root ρ^* of the coupling tree, it is easy to see from Definition 21 that $P_{i,\rho^*} = 1$.

Let $\rho \notin \mathcal{L}^*$ be a node in the coupling tree and let u be the first variable of ρ. For $X \in \{\mathsf{T}, \mathsf{F}\}$ and $i = 1$ we wish to establish $P_{1,\rho} = P_{1,\rho_{u\to X,u\to\mathsf{T}}} + P_{1,\rho_{u\to X,u\to\mathsf{F}}}$. Plugging in Definition 21 and dividing by $|\Omega_1^\Lambda|$, the constraint is equivalent (for any $Y \in \{\mathsf{T}, \mathsf{F}\}$) to

$$\frac{Q(\rho)}{|\Omega^{\mathcal{A}_1(\rho)\cup\Lambda}|} = \frac{Q(\rho_{u\to X,u\to\mathsf{T}})}{|\Omega^{\mathcal{A}_1(\rho_{u\to X,u\to\mathsf{T}})\cup\Lambda}|} + \frac{Q(\rho_{u\to X,u\to\mathsf{F}})}{|\Omega^{\mathcal{A}_1(\rho_{u\to X,u\to\mathsf{F}})\cup\Lambda}|} = \frac{Q(\rho_{u\to X,u\to\mathsf{T}})}{|\Omega^{\mathcal{A}_1(\rho_{u\to X,u\to Y})\cup\Lambda}|} + \frac{Q(\rho_{u\to X,u\to\mathsf{F}})}{|\Omega^{\mathcal{A}_1(\rho_{u\to X,u\to Y})\cup\Lambda}|},$$

where we used again the fact that $\mathcal{A}_1(\rho_{u\to X,u\to X}) = \mathcal{A}_1(\rho_{u\to X,u\to\neg X})$. The equation above follows from Observation 20 using (5). The other three constraints are similar.

Constraint Set 3: This case directly follows from Lemma 23. □

7.2. **ℓ-wrong assignments.** There are two kinds of errors which cause solutions of the LP to differ from the ratio $|\Omega_1^\Lambda|/|\Omega_2^\Lambda|$. The first kind of error involves a notion that we call "ℓ-wrong assignments". To define them, we need some graph-theoretic notation.

Definition 25. *Given a graph G with vertices u and v in $V(G)$, let $\mathrm{dist}_G(u,v)$ be the distance between u and v in G — that is, the number of edges in a shortest path from u to v. Given a subset $T \subseteq V(G)$ and a vertex $v \in V(G)$, let $\mathrm{dist}_G(u,T)$ be $\min_{v\in T} \mathrm{dist}_G(u,v)$. For any positive integer k, let G^k be the graph with vertex set $V(G)$ in which vertices u and v are connected if and only if there is a length-k path from u to v in G. Let $G^{\le k}$ be the graph with vertex set $V(G)$ in which vertices u and v are connected if and only if there is a path from u to v in G of length at most k.*

The main combinatorial structure that we use is a set $\mathcal{D}(G_\Phi)$, which is based on Alon's "2,3-tree" [6]. Similar structures were subsequently used in [35, 25]. The main difference between our definition and previous ones is that we take into account whether clauses are connected via good variables.

Definition 26. *Given the graph G_Φ, let $\mathcal{D}(G_\Phi)$ be the set of subsets $T \subseteq V(G_\Phi)$ such that the following hold:*

(1) For any $c_1, c_2 \in T$, $\mathsf{var}(c_1) \cap \mathsf{var}(c_2) \cap \mathcal{V}_{good} = \emptyset$;
(2) The graph $G_\Phi^{\le 4}[T]$, which is the subgraph of $G_\Phi^{\le 4}$ induced by T, is connected.

The following two lemmas regarding $\mathcal{D}(G_\Phi)$ will be useful. We defer their proofs to Section 8.6.

Lemma 27. *Let ℓ be an integer which is at least $\log n$. W.h.p. over the choice of Φ, every clause $c \in \mathcal{C}_{good}^\Lambda$ has the property that the number of size-ℓ subsets $T \in \mathcal{D}(G_\Phi)$ containing c is at most $(18k^2\alpha)^{4\ell}$.*

Recall that the coupling tree $\mathbb{T}^\Lambda$ is defined with respect to a prefix Λ of the partial assignment Λ^* from Lemma 10. Let c^* be the first clause of the root node ρ^*.

Lemma 28. *W.h.p. over the choice of Φ, every node ρ in $\mathbb{T}^\Lambda$ with $|V_I(\rho)| \geq L$ has the property that there is a set $T \subseteq \mathcal{F}(\rho)$ containing c^* such that $T \in \mathcal{D}(G_\Phi)$, $|T| = C_0\lceil \log(n/\varepsilon)\rceil$ and $|T \cap \mathcal{C}_{bad}| \leq |T|/3$.*

We now define ℓ-wrong assignments.

Definition 29. *An assignment $\sigma \in \Omega_i^\Lambda$ is ℓ-wrong if there is a size-ℓ set $T \in \mathcal{D}(G_\Phi)$ such that*

- *$c^* \in T$,*
- *$|T \cap \mathcal{C}_{good}^\Lambda| \geq 2\,|T|\,/3$, and*
- *there is a size $\lceil \ell/2\rceil$ subset S of $T \cap \mathcal{C}_{good}^\Lambda$ such that the restriction of σ to marked variables in clauses in S does not satisfy any clause in S. (Formally, taking U to be $\cup_{c\in S}\mathsf{marked}(c)$, the condition is that $\sigma[U]$ does not satisfy any clauses in S.)*

Otherwise σ is ℓ-correct.

The following is similar to [25, Lemma 4.8].

Lemma 30. *Fix $i \in \{1,2\}$. Let $\ell = L/(3k^2\Delta)$. Then the fraction of assignments in Ω_i^Λ that are ℓ-wrong is at most $(k\Delta)^{-9\ell}$.*

Proof. Assume $i = 1$, as the other case is similar. We want to show that

$$\mathrm{Pr}_{\sigma\sim\mu_{\Omega_1^\Lambda}}(\sigma \text{ is } \ell\text{-wrong}) \leq (k\Delta)^{-9\ell}.$$

If every assignment in $\mu_{\Omega_1^\Lambda}$ is ℓ-correct, then we are finished. Otherwise, consider a size-ℓ set $T = \{c_1, c_2, \ldots, c_\ell\}$ in $\mathcal{D}(G_\Phi)$ such that $c_\ell = c^*$ and $\ell_T \geq 2\ell/3$ where $\ell_T := |T \cap \mathcal{C}_{\mathrm{good}}^\Lambda|$.

Let $V := \mathcal{V}^\Lambda \setminus \{v^*\}$ be the set of unassigned variables in ρ^*, and let Ω' be the set of all assignments $\sigma : V \to \{\mathsf{T}, \mathsf{F}\}$. Let $U_T := V \cap \left(\cup_{i=1}^{\ell}\mathsf{marked}(c_i)\right)$ be the set of all marked variables in V that are in clauses in T. Given an assignment $\sigma \in \Omega'$, let $Z_T(\sigma)$ be the number of clauses in $T \cap \mathcal{C}_{\mathrm{good}}^\Lambda$ that are not satisfied by $\sigma[U_T]$.

Suppose that σ is drawn from $\mu_{\Omega'}$. By Corollary 11, every $c \in \mathcal{C}_{\mathrm{good}}^\Lambda$ has at least $k/4-1$ marked variables in V. Thus, for every $c \in T \cap \mathcal{C}_{\mathrm{good}}^\Lambda$, the probability that c is not satisfied by $\sigma[U_T]$ is at most $p^* := 2^{1-k/4}$. For $c \in T \cap \mathcal{C}_{\mathrm{good}}^\Lambda$, let A_c be the event that c is not satisfied by $\sigma[U_T]$. Because $T \in \mathcal{D}(G_\Phi)$, clauses in T do not share good variables (so they do not share marked variables). Thus, the events A_c are mutually independent. Since $\ell_T = |T \cap \mathcal{C}_{\mathrm{good}}^\Lambda|$, we have that the event $Z_T(\sigma) \geq \lceil \ell/2\rceil$ is dominated above by the probability that $\widehat{Z} \geq \lceil \ell/2\rceil$ where $\widehat{Z}$ is a sum of ℓ_T independent Bernoulli r.v.s with parameter p^*. Setting $\gamma := \ell/(2p^*\ell_T) - 1 \geq 1/(2p^*) - 1$ and applying a Chernoff bound to $\widehat{Z}$, we obtain that

$$\begin{aligned}\mathrm{Pr}_{\mu_{\Omega'}}(Z_T(\sigma) \geq \lceil \ell/2\rceil) &= \mathrm{Pr}_{\mu_{\Omega'}}(Z_T(\sigma) \geq \ell/2) = \mathrm{Pr}_{\mu_{\Omega'}}(Z_T(\sigma) \geq (1+\gamma)p^*\ell_T)\\ &\leq \mathrm{Pr}(\widehat{Z} \geq (1+\gamma)p^*\ell_T) \leq \left(\frac{e^\gamma}{(1+\gamma)^{1+\gamma}}\right)^{p^*\ell_T} \leq \left(2ep^*\right)^{\ell_T/2} \leq \left(2ep^*\right)^{\ell/3},\end{aligned}$$

where the second-to-last inequality follows by substituting in the lower bound $1/(2p^*) - 1$ for γ.

Let $\mathcal{E}_T$ be the event that $Z_T(\sigma) \geq \lceil \ell/2\rceil$, so $\mathsf{var}(\mathcal{E}_T) = U_T$. We apply Lemma 9 with the partial assignment $\Lambda \cup \{v^* \to \mathsf{T}\}$ and with the event A from Lemma 9 being $\mathcal{E}_T$. A clause c is in $\Gamma(\mathcal{E}_T)$ if and only if $\mathsf{var}(c)$ intersects U_T. Since all variables in U_T are marked, each $c_i \in T \cap \mathcal{C}_{\mathrm{good}}^\Lambda$ has at most $3k/4$ variables in U_T. Moreover, for each $v \in U_T$, there are at most Δ clauses containing v. Thus, $|\Gamma(\mathcal{E}_T)| \leq 3k\ell\Delta/4$.

Plugging everything into Lemma 9, we obtain

$$\mathrm{Pr}_{\mu_{\Omega_1^\Lambda}}(\mathcal{E}_T) \leq \mathrm{Pr}_{\mu_{\Omega'}}(\mathcal{E}_T)\Big(1 - \frac{1}{sk\Delta}\Big)^{-3k\ell\Delta/4} \leq \left(2ep^*\right)^{\ell/3} e^{\ell/s},$$

where the value s (defined just before Lemma 9) is $s = 2^{k/4}/(ek\Delta)$.

Note from the definition of L that $\ell = C_0\lceil\log(n/\varepsilon)\rceil$ for a sufficiently large constant C_0, so $\ell \geq \log n$. Applying Lemma 27 to the good clause c^*, w.h.p. over the choice of Φ, the number of size-ℓ sets $T \in \mathcal{D}(G_\Phi)$ containing c^* is at most $(18k^2\alpha)^{4\ell}$. Hence, by a union bound,

$$\begin{aligned}
\Pr_{\mu_{\Omega_1^\Lambda}}(\sigma \text{ is } \ell\text{-wrong}) &\leq \sum_{T\in\mathcal{D}(G_\Phi):|T|=\ell,c^*\in T,\ell_T\geq 2\ell/3} \Pr_{\mu_{\Omega_1^\Lambda}}(\mathcal{E}_T) \\
&\leq (18k^2\alpha)^{4\ell}\left(2ep^*\right)^{\ell/3} e^{\ell/s} = \left(18^4k^8\alpha^4 2^{1/3}e^{1/3}2^{1/3}2^{-k/12}e^{1/s}\right)^{\ell} \\
&\leq (k\Delta)^{-9\ell},
\end{aligned}$$

where in the last inequality we used that $\alpha \leq 2^{k/100}$ and $\Delta = 2^{k/300}$. □

7.3. **Errors from truncating nodes.** We have already noted that the existence of ℓ-wrong assignments can be viewed as "errors" which cause solutions of the LP to differ from the ratio $\left|\Omega_1^\Lambda\right|/\left|\Omega_2^\Lambda\right|$. However, even if an assignment $\sigma \in \Omega_i^\Lambda$ is ℓ-correct, it may still induce some error in the solution of the LP due to the truncation of the coupling tree. We account for this kind of error in this section.

Fix an assignment $\sigma \in \Omega_1^\Lambda$. Given a solution $\mathbf{P} = \{P_{i,\rho}\}$ to the LP, we consider the following stochastic process, which is a process (depending on σ) for sampling from $\mathcal{L}^*$. The process is defined for $i = 1$ rather than for $i = 2$. That is it starts with $\sigma \in \Omega_1^\Lambda$ and it uses the LP values $P_{1,\rho}$. (We will later require a similar process for $i = 2$).

Definition 31 (Sampling conditioned on $\sigma \in \Omega_1^\Lambda$)**.** *Fix $\sigma \in \Omega_1^\Lambda$. Here is a method for choosing a node $\rho \in \mathcal{L}^*$. Start by setting ρ to be the root ρ^* of the coupling tree. From every node ρ that is not in $\mathcal{L}^*$, proceed to a child, as follows. Let u be the first variable of ρ. For each $X \in \{\mathsf{T}, \mathsf{F}\}$, go to $\rho_{u\to\sigma(u),u\to X}$ with probability $P_{1,\rho_{u\to\sigma(u),u\to X}}/P_{1,\rho}$.*

This process is well-defined because $\mathbf{P}$ satisfies **Constraint Set 2**. If ρ is a leaf such that $\mathcal{A}_1(\rho)$ agrees with σ, then the probability of reaching ρ is $P_{1,\rho}$. The following definition concerns the truncating nodes $\rho \in \mathcal{T}$ where the process can also stop.

Lemma 32. *Fix $r_{\mathsf{lower}} \leq r_{\mathsf{upper}}$. Let $\ell = L/(3k^2\Delta)$. W.h.p. over the choice of Φ, the following holds. Let σ be any ℓ-correct assignment in Ω_1^Λ. If the LP has a solution using r_{lower} and r_{upper} then $\sum_{\rho\in\mathcal{T}:\sigma\in\Omega^{\Lambda\cup\mathcal{A}_1(\rho)}} P_{1,\rho} \leq (k\Delta)^{-8\ell}$.*

Proof. Let π_σ be the distribution on nodes in $\mathcal{L}^*$ from the sampling procedure in Definition 31. Let $\Upsilon_\sigma = \{\rho \in \mathcal{T} \mid \sigma \in \Omega^{\mathcal{A}_1(\rho)\cup\Lambda}\}$. The probability that the sampling procedure reaches any node $\rho \in \mathcal{L}^*$ with $\sigma \in \Omega^{\mathcal{A}_1(\rho)\cup\Lambda}$ is $P_{1,\rho}$, so the probability that it reaches Υ_σ is $\sum_{\rho\in\Upsilon_\sigma} P_{1,\rho}$, which is what we want to bound.

From the definition of L, note that $\ell = C_0\lceil\log(n/\varepsilon)\rceil$ for a sufficiently large constant C_0. Suppose that the formula Φ satisfies the condition in Lemma 28 (which it does, w.h.p.). Any node $\rho \in \Upsilon_\sigma$ satisfies $|V_I(\rho)| > L$, so by Lemma 28, there is a set $T \subseteq \mathcal{F}(\rho)$ containing c^* such that $T \in \mathcal{D}(G_\Phi)$, $|T| = \ell$ and $|T \cap \mathcal{C}_{\text{bad}}| \leq |T|/3$. This implies that

$$\left|T \cap C_{\text{good}}^\Lambda\right| \geq \frac{2\,|T|}{3}. \tag{9}$$

Let W be the set of all size-ℓ sets $T \in \mathcal{D}(G_\Phi)$ containing c^* such that (9) holds. Then the probability that the sampling procedure reaches Υ_σ is at most

$$\Pr_{\rho\sim\pi_\sigma}(\exists T \in W \text{ such that } T \subseteq \mathcal{F}(\rho)).$$

By a union bound we have

$$\sum_{\rho\in\Upsilon_\sigma} P_{1,\rho} \leq \sum_{T\in W} \Pr_{\rho\sim\pi_\sigma}(T \subseteq \mathcal{F}(\rho)),$$

so to finish we will show

$$\sum_{T\in W} \Pr_{\rho\sim\pi_\sigma}(T\subseteq \mathcal{F}(\rho)) \le (k\Delta)^{-8\ell}. \tag{10}$$

Consider a set $T \in W$ and any $\rho = (\mathcal{A}_1, \mathcal{A}_2, V_I, V_{\text{set}}, \mathcal{C}_{\text{rem}}, \mathcal{F}, R)$ such that ρ is in the support of π_σ and $T \subseteq \mathcal{F}(\rho)$. By the definition of π_σ, $\sigma \in \Omega^{\mathcal{A}_1(\rho)\cup\Lambda}$. By (P8), every $c \in T$ has $R(c)$ non-empty.

Let $T_{\text{good}} = T \cap \mathcal{C}^{\Lambda}_{\text{good}}$ and note that $|T_{\text{good}}| \ge 2\ell/3$. By (P9.1), every $c \in T_{\text{good}}$ has $R(c) \subseteq \{\mathsf{disagree}, 1, 2\}$.

Let $S = \{c \in T_{\text{good}} \mid 1 \in R(c)\}$. By (P9.3), every $c \in S$ has $\mathsf{marked}(c) \subseteq V_{\text{set}}$ and $\mathcal{A}_1$ does not satisfy c. Since σ is ℓ-correct, $|S| < \lceil \ell/2 \rceil$ so (by integrality) $|S| < \ell/2$. Thus, $|T_{\text{good}} \setminus S| \ge (2/3 - 1/2)\ell = \ell/6$.

We claim that for any $c \in T_{\text{good}} \setminus S$, $\mathsf{disagree} \in R(c)$. We have already seen that $R(c)$ is non-empty and that it is contained in $\{\mathsf{disagree}, 2\}$. If $2 \in R(c)$, then by (P9.3), c is satisfied by $\Lambda \wedge \mathcal{A}_1$ but not $\Lambda \wedge \mathcal{A}_2$, which implies that $\mathsf{disagree} \in R(c)$ by (P9.2).

Since $(T_{\text{good}} \setminus S) \subseteq T \in \mathcal{D}(G_\Phi)$, Definition 26 ensures that the clauses in $T_{\text{good}} \setminus S$ do not share good variables. Thus by (P9.2), there is a set $V \subseteq \mathcal{V}^\Lambda$ with $|V| \ge \ell/6$ such that for every $u \in V$, $\mathcal{A}_1(u) \ne \mathcal{A}_2(u)$.

By the definition of the sampling procedure (Definition 31), the probability that such a disagreement occurs when the first variable u of a node ρ' is set is at most $P_{1,\rho'_{u\to X, u\to\neg X}}/P_{1,\rho'}$ for some $X \in \{\mathsf{T}, \mathsf{F}\}$. By **Constraint Set 3**, this ratio is at most $1/s$. Also, these events are independent for the variables $u \in V$. Thus,

$$\Pr_{\rho\sim\pi_\sigma}(T \subseteq \mathcal{F}(\rho)) \le s^{-\ell/6}.$$

By the choice of C_0, ℓ is at least $\log n$. Lemma 27 implies that w.h.p., over the choice of Φ, $|W| \le (18k^2\alpha)^{4\ell}$. By a union bound (using the fact that k is sufficiently large and $\alpha < 2^{k/300}$),

$$\sum_{T\in W} \Pr_{\rho\sim\pi_\sigma}(T\subseteq\mathcal{F}(\rho)) \le (18k^2\alpha)^{4\ell} s^{-\ell/6} = \left(18^4 k^8 \alpha^4 s^{-1/6}\right)^{\ell} \le (k\Delta)^{-8\ell}. \qquad \square$$

The following lemma is the same as Lemma 32, except that we take $i = 2$ rather than $i = 1$. The proof is exactly the same as that of Lemma 32 except that the sampling procedure (analogous to the one from Definition 31) is conditioned on $\sigma \in \Omega_2^\Lambda$ and the transition from ρ to $\rho_{u\to X, u\to\sigma(u)}$ is with probability $P_{2,\rho_{u\to X,u\to\sigma(u)}}/P_{2,\rho}$.

Lemma 33. *Fix $r_{\mathsf{lower}} \le r_{\mathsf{upper}}$. Let $\ell = L/(3k^2\Delta)$. W.h.p. over the choice of Φ, the following holds. Let σ be any ℓ-correct assignment in Ω_2^Λ. If the LP has a solution using r_{lower} and r_{upper} then $\sum_{\rho\in\mathcal{T}:\sigma\in\Omega^{\Lambda\cup\mathcal{A}_2(\rho)}} P_{2,\rho} \le (k\Delta)^{-8\ell}$.*

7.4. **Soundness.** In this section we show the "soundness" of the LP, namely, that whenever a solution to the LP exists, it yields a bound on $|\Omega_1^\Lambda|/|\Omega_2^\Lambda|$.

Lemma 34. *Fix $r_{\mathsf{lower}} \le r_{\mathsf{upper}}$. W.h.p. over the choice of Φ, the following holds. If the LP has a solution $\mathbf{P}$ using r_{lower} and r_{upper}, then $e^{-\varepsilon/(3n)} r_{\mathsf{lower}} \le |\Omega_1^\Lambda|/|\Omega_2^\Lambda| \le e^{\varepsilon/(3n)} r_{\mathsf{upper}}$.*

Proof. By Lemma 18, the constraints in **Constraint Set 2** guarantee that, for any $i \in \{1,2\}$ and $\sigma \in \Omega_i^\Lambda$,

$$\sum_{\rho\in\mathcal{L}^*:\sigma\in\Omega^{\mathcal{A}_i(\rho)\cup\Lambda}} P_{i,\rho} = 1.$$

Thus,

$$|\Omega_i^\Lambda| = \sum_{\sigma\in\Omega_i^\Lambda} 1 = \sum_{\sigma\in\Omega_i^\Lambda} \sum_{\rho\in\mathcal{L}^*:\sigma\in\Omega^{\mathcal{A}_i(\rho)\cup\Lambda}} P_{i,\rho}.$$

Let $\ell = L/(3k^2\Delta)$. We start by defining Z_i, Z'_i and Z''_i as follows for $i \in \{1,2\}$.

$$Z_i = \sum_{\sigma\in\Omega_i^\Lambda} \sum_{\rho\in\mathcal{L}:\sigma\in\Omega^{\mathcal{A}_i(\rho)\cup\Lambda}} P_{i,\rho},$$

$$Z'_i = \sum_{\sigma\in\Omega_i^\Lambda,\ \sigma \text{ is } \ell\text{-wrong}} \sum_{\rho\in\mathcal{L}^*:\sigma\in\Omega^{\mathcal{A}_i(\rho)\cup\Lambda}} P_{i,\rho},$$

$$Z''_i = \sum_{\sigma\in\Omega_i^\Lambda,\ \sigma \text{ is } \ell\text{-correct}} \sum_{\rho\in\mathcal{T}:\sigma\in\Omega^{\mathcal{A}_i(\rho)\cup\Lambda}} P_{i,\rho}.$$

Thus $Z_i \le \left|\Omega_i^\Lambda\right| \le Z_i + Z'_i + Z''_i$. The proof consists of three parts — as we will see soon, the statement of the lemma follows directly from Equations (11), (12), and (13).

Part 1: Showing

$$r_{\text{lower}} \le \frac{Z_1}{Z_2} \le r_{\text{upper}}. \tag{11}$$

Part 2: Showing, for $i \in \{1,2\}$,

$$\frac{Z'_i}{\left|\Omega_i^\Lambda\right|} \le \frac{1-e^{-\varepsilon/(3n)}}{2}. \tag{12}$$

Part 3: Showing, for $i \in \{1,2\}$,

$$\frac{Z''_i}{\left|\Omega_i^\Lambda\right|} \le \frac{1-e^{-\varepsilon/(3n)}}{2}. \tag{13}$$

We now present the three parts of the proof.

Part 1. This part is straightforward. Exchanging the order of summation in the definition of Z_i, we have

$$Z_i = \sum_{\rho\in\mathcal{L}} \sum_{\sigma\in\Omega_i^\Lambda:\sigma\in\Omega^{\mathcal{A}_i(\rho)\cup\Lambda}} P_{i,\rho} = \sum_{\rho\in\mathcal{L}} P_{i,\rho}\cdot|\Omega^{\mathcal{A}_i(\rho)\cup\Lambda}|. \tag{14}$$

Since $\rho\in\mathcal{L}$, Lemma 17 guarantees that $r(\rho) = |\Omega^{\mathcal{A}_1(\rho)\cup\Lambda}|/|\Omega^{\mathcal{A}_2(\rho)\cup\Lambda}|$ and **Constraint Set 1** guarantees that

$$r_{\text{lower}} \le \frac{P_{1,\rho}\cdot\left|\Omega^{\mathcal{A}_1(\rho)\cup\Lambda}\right|}{P_{2,\rho}\cdot\left|\Omega^{\mathcal{A}_2(\rho)\cup\Lambda}\right|} = \frac{P_{1,\rho}\cdot r(\rho)}{P_{2,\rho}\cdot} \le r_{\text{upper}}.$$

Plugging in (14), we get (11), as required.

Part 2. Note that for any σ, $\sum_{\rho\in\mathcal{L}^*:\sigma\in\Omega^{\mathcal{A}_i(\rho)\cup\Lambda}} P_{i,\rho} \le 1$. By Lemma 30, we have that

$$\frac{Z'_i}{\left|\Omega_i^\Lambda\right|} \le \frac{\left|\{\sigma\in\Omega_i^\Lambda : \sigma \text{ is } \ell\text{-wrong}\}\right|}{\left|\Omega_i^\Lambda\right|} \le (k\Delta)^{-9\ell}.$$

Since $\ell = L/(3k^2\Delta) = C_0\lceil\log(n/\varepsilon)\rceil$, we verify that

$$(k\Delta)^{-9\ell} \le (n/\varepsilon)^{-9C_0\log(k\Delta)} \le (1-e^{-\varepsilon/(3n)})/2,$$

which implies (12). This finishes Part 2.

Part 3. By Lemmas 32 and 33, W.h.p. over the choice of Φ, for every ℓ-correct $\sigma\in\Omega_i^\Lambda$, $\sum_{\rho\in\mathcal{T}:\sigma\in\Omega^{\mathcal{A}_i(\rho)\cup\Lambda}} P_{i,\rho} \le (k\Delta)^{-8\ell}$. Hence we have

$$\frac{Z''_i}{\left|\Omega_i^\Lambda\right|} \le \frac{Z''_i}{\left|\{\sigma\in\Omega_i^\Lambda : \sigma \text{ is } \ell\text{-correct}\}\right|} \le (k\Delta)^{-8\ell}.$$

Again, $\ell = L/(3k^2\Delta) = C_0\lceil\log(n/\varepsilon)\rceil$ implies (13). This finishes Part 3.

Having finished the three parts, we now complete the proof. Combining (12) and (13) with the fact that $Z_i \le \left|\Omega_i^\Lambda\right| \le Z_i + Z'_i + Z''_i$, we get

$$e^{-\varepsilon/(3n)} \le \frac{Z_i}{\left|\Omega_i^\Lambda\right|} \le 1.$$

Plugging in (11) we obtain

$$e^{-\varepsilon/(3n)} r_{\mathsf{lower}} \le \frac{|\Omega_1^\Lambda|}{|\Omega_2^\Lambda|} \le e^{\varepsilon/(3n)} r_{\mathsf{upper}}. \qquad \square$$

8. Properties of the random formula

We still need to prove Lemmas 14, 27, and 28. All of these lemmas depend on properties of the random formula. The main intuition behind the proofs is that individual variables within the formula might have unbounded degrees, but once we consider sets of logarithmically many vertices (that form connected sets) they behave analogously to bounded-degree sets. In Section 8.1, we prove some rather standard facts about the random formula Φ, concerning the number of high-degree variables and its expansion properties. Then, in Section 8.2, we show how to bound the number of connected sets of variables in the formula Φ, and formalise the intuition above by showing Lemma 40, from which Lemma 27 is derived in Section 8.6. Then, in Section 8.3, we show Lemmas 41 and 42 which bound the number of high-degree variables in connected sets; in Section 8.4, we extend these to bounds for the number of bad variables in connected sets using expansion properties of the formula. These are the key ingredients for the proofs of Lemmas 14 and 28, which are given in Sections 8.5 and 8.6, respectively.

Throughout this section, we use $\Pr_\Phi(\cdot)$ to denote the distribution for choosing Φ.

8.1. Bounding the number of high-degree variables.

Recall from Section 3 that $\mathcal{V}_0$ is the set of high-degree variables. Based on the average-degree assumption and the randomness of the formula, it is standard to show the following.

Lemma 35. *W.h.p. over the choice of Φ, the size of $\mathcal{V}_0$ is at most $n/2^{k^{10}}$.*

Proof. The degrees of the variables in Φ have the same distribution as a balls-and-bins experiment with km balls and n bins. Let $D_1, \dots, D_n$ be a set of independent Poisson variables with parameter $k\alpha$, denoted $\mathrm{Poi}(k\alpha)$. It follows by well-known facts (see, e.g., [34, Chapter 5.4]) that the degrees of the variables in Φ have the same distribution as $\{D_1, \dots, D_n\}$ conditioned on the event $\mathcal{E}$ that $D_1 + \dots + D_n = km$, and that $\Pr(\mathcal{E}) = O(1/\sqrt{n})$. Let $U = \{i \in [n] : D_i \ge \Delta\}$, then

$$\mathbb{E}[|U|] = n \Pr(\mathrm{Poi}(k\alpha) \ge \Delta) \le n e^{-\Delta \cdot} \le n/2^{k^{10}+1},$$

where the first inequality follows from $k\alpha \le \Delta/k^2$ and using standard bounds for the tails of the Poisson distribution (see, e.g., [34, Theorem 5.4]). A Chernoff bound therefore yields that $\Pr(|U| \ge n/2^{k^{10}}) = \exp(-\Omega(n))$. It follows that

$$\Pr_\Phi(|\mathcal{V}_0| \ge n/2^{k^{10}}) = \Pr(|U| \ge n/2^{k^{10}} \mid \mathcal{E}) \le \exp(-\Omega(n)). \qquad \square$$

We will also need the following expansion properties of the random formula Φ; these bound the number of clauses that can contain more than b variables from a relatively small set of variables Y.

Lemma 36. *Let $2 \le b \le k$ be an integer and $t = \frac{2}{b-1}$. W.h.p. over the choice of Φ, for every set of variables Y such that $2 \le |Y| \le n/2^k$, the number of clauses that contain at least b variables from Y is at most $t|Y|$.*

Proof. Let y be an integer between b and $n/2^k$. There are $\binom{n}{y}$ ways to choose a set Y of y variables Y and $\binom{m}{\lceil ty \rceil}$ ways to choose a set Z of $\lceil ty \rceil$ clauses. The probability that a clause contains at least b variables from Y is bounded by $\binom{k}{b}\left(\frac{y}{n}\right)^b \le (ky/n)^b$. By a union bound over the choices of Y and Z, we therefore obtain that

$$\Pr_\Phi(\exists Y, Z) \le \sum_{b \le y \le n/2^k} \binom{n}{y}\binom{m}{\lceil ty \rceil}\left(\frac{ky}{n}\right)^{b\lceil ty \rceil} \le \sum_{b \le y \le n/2^k} \left(\frac{en}{y}\right)^y \left(\frac{e\alpha n}{ty}\left(\frac{ky}{n}\right)^b\right)^{\lceil ty \rceil}.$$

Note (by taking k to be its upper bound) that the quantity taken to the $\lceil ty \rceil$ power is at most

$$\frac{(b-1)e\alpha 2^k}{2}\left(\frac{k}{2^k}\right)^b$$

This is maximised at $b = 2$, so it is always at most 1 (given that $\alpha < 2^{rk}$ for some $r < 1$ and that k is sufficiently large). Thus, the quantity is maximised by removing the ceiling, so

$$\Pr_\Phi(\exists Y, Z) \leq \sum_{b \leq y \leq n/2^k} \left(\frac{en}{y}\left(\frac{e\alpha n}{ty}\right)^t \left(\frac{ky}{n}\right)^{bt}\right)^y = \sum_{b \leq y \leq n/2^k} \left(\frac{e^{1+t}k^{bt}\alpha^t y}{t^t n}\right)^y = o(1).$$

The last estimate follows from observing the inequalities $\left(\frac{e^{1+t}k^{bt}\alpha^t y}{t^t n}\right)^y \leq 1/n$ for $2 \leq y \leq \log n$, and $\frac{e^{1+t}k^{bt}\alpha^t y}{t^t n} < 1/10$ for $\log n < y \leq n/2^k$, which hold for all sufficiently large n. ☐

Applying Lemma 36 with $b = t = 2$ and with $b = \lceil k/10 \rceil$, $t = 2/(b-1) < 30/k$ gives the following two corollaries, respectively.

Corollary 37. *W.h.p. over the choice of Φ, for every set of variables Y such that $2 \leq |Y| \leq n/2^k$, the number of clauses that contain at least 2 variables from Y is at most $2|Y|$.*

Corollary 38. *W.h.p. over the choice of Φ, for every set of variables Y such that $2 \leq |Y| \leq n/2^k$, the number of clauses that contain at least $k/10$ variables from Y is at most $\frac{30}{k}|Y|$.*

8.2. **Bounding the number of connected sets of clauses.** In this section, we bound the number of connected sets of clauses which we will use in the upcoming sections. Recall that a set Y of clauses is connected if $G_\Phi[Y]$ is connected.

Lemma 39. *For any labelled tree T on a subset of clauses in Φ, the probability that T is a subgraph of G_Φ is at most $(k^2/n)^{|V(T)|-1}$.*

Proof. For a tree T on a subset of clauses in Φ, we use $T \subseteq G_\Phi$ to denote that T is a subgraph of G_Φ. We prove our claim by induction on the size of T. If $|V(T)| = 1$, then T contains only an isolated clause, so $\Pr(T \subseteq G_\Phi) = 1$. Now suppose that the claim holds for all trees with size $|V(T)| - 1$. Let c be a leaf in T and c' be its neighbour in T. Then we have

$$\Pr_\Phi(T \subseteq G_\Phi) = \Pr_\Phi\big((T \setminus c) \subseteq G_\Phi\big)\,\Pr_\Phi\big(\mathsf{var}(c) \cap \mathsf{var}(c') \neq \emptyset \mid (T \setminus c) \subseteq G_\Phi\big).$$

For any fixed c', we have that

$$\Pr(\mathsf{var}(c) \cap \mathsf{var}(c') \neq \emptyset) \leq \sum_{v \in \mathsf{var}(c')} \Pr(v \in \mathsf{var}(c)) \leq \frac{k^2}{n}.$$

Note that the events $(T \setminus c) \subseteq G_\Phi$ and $\mathsf{var}(c) \cap \mathsf{var}(c') \neq \emptyset$ are independent, so

$$\Pr\big(\mathsf{var}(c) \cap \mathsf{var}(c') \neq \emptyset \mid (T \setminus c) \subseteq G_\Phi\big) \leq \frac{k^2}{n}\,.$$

Since $T \setminus c$ is a tree of size $|V(T)| - 1$, by the induction hypothesis we have that $\Pr((T \setminus c) \subseteq G_\Phi) \leq (k^2/n)^{|V(T)|-2}$. We conclude that $\Pr(T \subseteq G_\Phi) \leq (k^2/n)^{|V(T)|-1}$. ☐

Lemma 40. *W.h.p. over the choice of Φ, for any clause c, the number of connected sets of clauses in G_Φ with size $\ell \geq \log n$ containing c is at most $(9k^2\alpha)^\ell$.*

Proof. Let c be an arbitrary clause. Let U be a size-ℓ set of clauses containing c and let T_U be the set of all labelled trees on the set U; note that $|\mathsf{T}_U| = \ell^{\ell-2}$. For any tree $T \in \mathsf{T}_U$, by Lemma 39, the probability that $T \subseteq G_\Phi$ is at most $(k^2/n)^{\ell-1}$. Thus

$$\Pr_\Phi(G_\Phi[U] \text{ is connected}) \leq \sum_{T \in \mathsf{T}_U} \Pr_\Phi(T \subseteq G_\Phi) = \ell^{\ell-2}(k^2/n)^{\ell-1}\,.$$

Let Z_c be the number of connected sets of clauses with size ℓ containing c. Then,

$$\begin{aligned}\mathbb{E}_\Phi[Z_c] &= \sum_{U\subseteq\mathcal{C};c\in U,|U|=\ell} \Pr_\Phi(G_\Phi[U] \text{ is connected})\\ &\le \binom{m-1}{\ell-1}\ell^{\ell-2}\left(\frac{k^2}{n}\right)^{\ell-1} \le \left(\frac{e(m-1)}{\ell-1}\right)^{\ell-1}\ell^{\ell-2}\left(\frac{k^2}{n}\right)^{\ell-1}\\ &\le \left(\frac{emk^2}{n}\cdot\frac{\ell^{\frac{\ell-2}{\ell-1}}}{\ell-1}\right)^{\ell-1} \le (ek^2\alpha)^{\ell-1}.\end{aligned}$$

As $\ell \ge \log n$, by Markov's inequality, we obtain that $\Pr_\Phi\left(Z_c \ge (9k^2\alpha)^{\ell-1}\right) \le \left(\frac{e}{9}\right)^{\ell-1} = o(1/n)$. By a union bound over all clauses c (note that there are $O(n)$ for them), we obtain the conclusion of the lemma. ☐

8.3. **Bounding the number of high-degree variables in connected sets.** In this section, we bound the number of high-degree variables in connected sets of variables.

Recall that $\mathcal{V}_0$ is the set of high-degree variables. For every set S of variables, let $\mathsf{HD}(S) = \mathcal{V}_0 \cap S$ be the set of high-degree variables in S. With this notation, $\mathcal{V}_0 = \mathsf{HD}(\mathcal{V})$. For a set $Y \subseteq \mathcal{C}$ of clauses, let $\mathsf{var}(Y) := \cup_{c\in Y}\mathsf{var}(c)$.

Lemma 41. *Let $\delta_0 > 0$ and $\theta_0 \ge \min\{k^2\alpha, 2\}$ be constants such that $\delta_0\theta_0\log(\theta_0/k^2\alpha) > \log\alpha + 3\log k$. Then, w.h.p. over the choice of Φ, there do not exist sets Y, Z of clauses and a set U of variables such that:*

(1) $|Y| \ge \log n$, $|U| \ge \delta_0 |Y|$, $|Z| \ge \theta_0 |U|$, and $Y \cap Z = \emptyset$;

(2) $G_\Phi[Y]$ is connected, $U \subseteq \mathsf{var}(Y)$, and every clause in Z contains at least one variable from U.

Proof. Let $\mathcal{E}$ the event that there exist sets Y, Z, U satisfying conditions (1) and (2).

Call a tuple (y, δ, θ) *feasible* if $y, \delta y, \theta\delta y$ are all integers, where $y \ge \log n$, $\delta \ge \delta_0$ and $\theta \ge \theta_0$. Fix a feasible tuple (y, δ, θ) and three sets of indices $I_Y \in \binom{[m]}{y}, I_U \in \binom{I_Y\times[k]}{\delta y}, I_Z \in \binom{[m]\setminus I_Y}{\theta\delta y}$. Define

$S_Y = \{c_i \mid i \in I_Y\}$

$S_U = \{\mathsf{var}(\ell_{i,j}) \mid (i,j) \in I_U\}$ where $\mathsf{var}(\ell_{i,j})$ is the variable corresponding to the literal $\ell_{i,j}$,

$S_Z = \{c_i \mid i \in I_Z\}$.

Denote by $\mathcal{E}_U$ the event that $|S_U| = \delta y$, by $\mathcal{E}_Y$ the event that $G_\Phi[S_Y]$ is connected and by $\mathcal{E}_Z$ the event that every clause in S_Z contains at least one variable from S_U. For any labelled tree T on vertex set S_Y, by Lemma 39, the probability that T is a subgraph of G_Φ is at most $(k^2/n)^{y-1}$. We have y^{y-2} such trees, so by a union bound

$$\Pr_\Phi(\mathcal{E}_Y) \le y^{y-2}\left(\frac{k^2}{n}\right)^{y-1}.$$

Moreover, by the independence of clauses, we have that

$$\Pr_\Phi(\mathcal{E}_Z \mid \mathcal{E}_U \wedge \mathcal{E}_Y) = \Pr_\Phi(\mathcal{E}_Z \mid \mathcal{E}_U) \le \left(\frac{k\delta y}{n}\right)^{\theta\delta y}.$$

Note that

$$\Pr_\Phi(\mathcal{E}_U \wedge \mathcal{E}_Y \wedge \mathcal{E}_Z) \le \Pr_\Phi(\mathcal{E}_Y)\Pr_\Phi(\mathcal{E}_Z \mid \mathcal{E}_Y \wedge \mathcal{E}_U),$$

so by a union bound over the choice of the tuple (y, δ, θ) and the sets I_Y, I_U, I_Z we have that

$$\Pr_\Phi(\mathcal{E}) \le \sum_{\text{feasible }(y,\delta,\theta)}\ \sum_{I_Y\in\binom{[m]}{y},I_U\in\binom{I_Y\times[k]}{\delta y},I_Z\in\binom{[m]\setminus I_Y}{\theta\delta y}} \Pr_\Phi(\mathcal{E}_Y)\Pr_\Phi(\mathcal{E}_Z \mid \mathcal{E}_Y \wedge \mathcal{E}_U).$$

It follows that

$$\Pr_\Phi(\mathcal{E}) \le \sum_{\text{feasible } (y,\delta,\theta)} \binom{m}{y}\binom{ky}{\delta y}\binom{m}{\theta\delta y} y^{y-2}\left(\frac{k^2}{n}\right)^{y-1}\left(\frac{k\delta y}{n}\right)^{\theta\delta y}$$

$$\le n \sum_{\text{feasible } (y,\delta,\theta)} \left(\frac{(ek)^{2+\delta+\theta\delta}\alpha^{\theta\delta+1}}{\delta^\delta\theta^{\theta\delta}}\right)^y. \tag{15}$$

Note that $\delta\theta\log(\theta/k^2\alpha) \ge \delta_0\theta_0\log(\theta_0/k^2\alpha) \ge \log\alpha + 3\log k$ and hence $\theta^{\delta\theta} \ge (k^2\alpha)^{\delta\theta}k^3\alpha$. It follows that

$$\frac{(ek)^{2+\delta+\theta\delta}\alpha^{\theta\delta+1}}{\delta^\delta\theta^{\theta\delta}} \le \frac{e^{2+\delta+\delta\theta}k^\delta}{\delta^\delta k^{\theta\delta+1}} = \frac{e^{2+\delta}k^\delta}{\delta^\delta(k/e)^{\theta\delta}k} \le \frac{2e^{2+\delta}k^\delta}{k(k/e)^{2\delta}} \le \frac{2e^2}{k} < \frac{1}{e^6},$$

where the last few inequalities hold for sufficiently large k combined with the fact that $\delta^\delta \ge 1/2$ for all $\delta > 0$ and our assumption $\theta \ge \theta_0 \ge 2$. Plugging this estimate into (15) and noting that there are $O(n^3)$ feasible tuples (y,δ,θ) and $y \ge \log n$, we obtain that $\Pr_\Phi(\mathcal{E}) = o(1)$, as needed. ☐

Lemma 42. *W.h.p. over the choice of Φ, every connected set U of variables with size at least $21600k\log n$ satisfies that $|\mathsf{HD}(U)| \le \frac{|U|}{21600}$.*

Proof. Let $\delta_0 = \frac{1}{21600}$ and $\theta_0 = \Delta - 2(k+1)$. Note that $\delta_0\theta_0\log\frac{\theta_0}{k^2\alpha} \ge 3\log k + \log\alpha$ for all sufficiently large k, so w.h.p. we have that Φ satisfies Lemma 41. Moreover, w.h.p. we have that Φ satisfies Corollary 37 and Lemma 35. We will show the conclusion of the lemma whenever Φ satisfies these properties.

For the sake of contradiction, suppose that U is a connected set of variables with $|U| \ge (k/\delta_0)\log n$ such that $h > \delta_0|U|$ where $h = |\mathsf{HD}(U)|$ is the number of high-degree variables in U. Recall that the factor graph of Φ is a bipartite graph where one side corresponds to variables and the other to clauses (whose edges join variables to clauses in the natural way). We next show that there is a tree T in the factor graph of Φ of size at most $2|U|$ such that

(1) every vertex in T is either a variable in U or a clause in Φ, all variables in $\mathsf{HD}(U)$ are vertices in T, and T contains at most $|U|$ clauses;
(2) every edge in T joins a variable and a clause, and for any variable v and clause c, (v,c) is an edge in T only if c contains v;
(3) $T_L \subseteq \mathsf{HD}(U)$, where T_L is the set of leaves of T;
(4) if a clause $c \in T$ contains any variable from $\mathsf{HD}(U)$, then at least one of its neighbours in T is a variable from $\mathsf{HD}(U)$.

Since $H_\Phi[U]$ is connected, there is a tree T' of size at most $2|U|$ that satisfies Items (1) and (2) (for example, we may take the Steiner tree with terminals $\mathsf{HD}(U)$ in the subgraph of the factor graph induced by U and its adjacent clauses). We now prune T' so that it satisfies Items (4) and (3) as well. For any clause c in T' such that c contains at least one variable from $\mathsf{HD}(U)$ but none of its neighbours in T' is from $\mathsf{HD}(U)$, let v be a variable from $\mathsf{var}(c) \cap \mathsf{HD}(U)$, and u be the neighbour of c on the path from c to v. Then we remove the edge (c,u) from T' and add the edge (c,v). Run this process until there is no such clause c. Now T' is a tree that satisfies Items (1), (2), (4). If T' has a leaf node which is not in $\mathsf{HD}(U)$, remove it from T'. Run this process until all leaf nodes are in $\mathsf{HD}(U)$ and let T be the remaining tree. Note that removing leaf nodes that are not in $\mathsf{HD}(U)$ does not affect Items (1), (2) and (4). We thus obtain a tree T satisfying all of these four items.

Let C_T be the set of clauses in T. From Item (1), we have $h/k \le |C_T| \le |U|$. Let t be the number of clauses in C_T that contain at least one variable from $\mathsf{HD}(U)$. By Item (4), we have

$$t \le \sum_{v \in \mathsf{HD}(U)} \mathsf{deg}_T(v).$$

Let $D = \sum_{v \in \mathsf{HD}(U) \setminus T_L} \mathsf{deg}_T(v)$. Because T is a tree and $\mathsf{HD}(U) \subseteq T$, we obtain that

$$|T_L| = 2 + \sum_{v \in T \setminus T_L} (\mathsf{deg}_T(v) - 2) \geq 2 + \sum_{v \in \mathsf{HD}(U) \setminus T_L} (\mathsf{deg}_T(v) - 2) = 2 + D - 2(h - |T_L|),$$

which yields that $D + |T_L| \leq 2h$. Thus we have $t \leq D + |T_L| \leq 2h$.

By our assumption on U, we have that the number of high-degree variables in U satisfies $h > \delta_0 |U| \geq k \log n$ and therefore $|C_T| \geq \log n$. Each variable in $\mathsf{HD}(U)$ is contained in at least Δ clauses. Moreover, by Lemma 35 we have $|\mathsf{HD}(U)| \leq |\mathcal{V}_0| \leq n/2^k$, so by Corollary 37 the number of clauses that contain at least 2 variables from $\mathsf{HD}(U)$ is at most $2h$. It follows that the number of clauses that contain at least one variable from $\mathsf{HD}(U)$ is at least $\Delta h - 2hk$. Since $t \leq 2h$, at most $2h$ of these clauses appear in T. Hence, there must exist a set Z of clauses of size at least $(\Delta - 2(k+1))h = \theta_0 h$ such that $Z \cap C_T = \emptyset$ and each clause in Z contains at least one variable from $\mathsf{HD}(U)$.

Note that $|C_T| \geq \log n$, $|\mathsf{HD}(U)| \geq \delta_0 |C_T|$ and $|Z| \geq \theta_0 |\mathsf{HD}(U)|$. Moreover, C_T is a connected set of clauses, $\mathsf{HD}(U) \subseteq \mathsf{var}(C_T)$ and every clause Z contains at least one variable from $\mathsf{HD}(U)$. This contradicts that Φ satisfies Lemma 41. Therefore, no such set U can exist, proving the lemma. $\square$

8.4. **Bounding the number of bad variables in connected sets.** In this section, we bound the number of bad variables in connected sets. Consider the following process P which we will use to work with bad components. The process, for every set S of variables, defines a set of variables $\mathsf{BC}(S)$.

(1) Let $\mathsf{BC}(S) = S$.
(2) While there is a clause c such that $|\mathsf{var}(c) \cap \mathsf{BC}(S)| \geq k/10$ and $\mathsf{var}(c) \setminus \mathsf{BC}(S) \neq \emptyset$ $\mathsf{BC}(S) := \mathsf{BC}(S) \cup \mathsf{var}(c)$

Note that $\mathcal{V}_{\text{bad}} = \mathsf{BC}(\mathcal{V}_0)$, where $\mathcal{V}_0$ is the set of high-degree variables. Recall from Section 3 that a bad component is a connected component of variables in $H_{\Phi,\text{bad}}$. The following lemma shows that the process P can be viewed as a "local" process for identifying bad components.

Lemma 43. *For every bad component S, we have $S = \mathsf{BC}(\mathsf{HD}(S))$.*

Proof. We run the process P starting from $\mathsf{HD}(S)$. Since $\mathsf{HD}(S)$ consists only of high-degree variables, $\mathsf{BC}(\mathsf{HD}(S)) \subseteq \mathcal{V}_{\text{bad}}$. By the construction of $\mathsf{BC}(\mathsf{HD}(S))$ and the definitions of $\mathcal{C}_{\text{bad}}$ and $H_{\Phi,\text{bad}}$, $H_{\Phi,\text{bad}}[\mathsf{BC}(\mathsf{HD}(S))]$ is connected. Since S is a connected component in $H_{\Phi,\text{bad}}$, we obtain that $\mathsf{BC}(\mathsf{HD}(S)) \subseteq S$.

Now we prove that $S \subseteq \mathsf{BC}(\mathsf{HD}(S))$. For the sake of contradiction, suppose that $S \setminus \mathsf{BC}(\mathsf{HD}(S)) \neq \emptyset$. Consider the process of identifying bad variables (cf. Section 3). Let i be the smallest number such that $V_i \cap (S \setminus \mathsf{BC}(\mathsf{HD}(S))) \neq \emptyset$ and let v be any variable from $V_i \cap (S \setminus \mathsf{BC}(\mathsf{HD}(S)))$. Thus there exists a clause $c \in C_{i-1}$ that contains v and at least $k/10$ variables in V_{i-1}. Let $U = \mathsf{var}(c) \cap V_{i-1}$. By Definition 5, variables in U are adjacent to v in $H_{\Phi,\text{bad}}$ and thus they are in S (since c becomes a bad clause after step i). Also by the choice of i, $U \cap (S \setminus \mathsf{BC}(\mathsf{HD}(S))) \subseteq V_{i-1} \cap (S \setminus \mathsf{BC}(\mathsf{HD}(S))) = \emptyset$, which implies that $U \subseteq \mathsf{BC}(\mathsf{HD}(S))$. Therefore, the clause c contains v and at least $k/10$ variables from $\mathsf{BC}(\mathsf{HD}(S))$, so v should also be in $\mathsf{BC}(\mathsf{HD}(S))$ according to the process P, which yields a contradiction. This finishes the proof. $\square$

To analyse the number of variables added by the process P, we will use an expansion property proved by Coja-Oghlan and Frieze [12], adapted for our purposes. First, we show the following slightly more quantitative version of [12, Lemma 2.4].

Lemma 44 ([12, Lemma 2.4])**.** *There exists a constant $k_0 > 0$ such that for all $k \geq k_0$ the following holds. With probability $1 - o(1/n)$ over the choice of the random formula Φ, for $\varepsilon > 0$ and $\lambda > 4$ satisfying $\varepsilon \leq k^{-3}$ and $\varepsilon^\lambda \leq \frac{1}{e}(2e)^{-4k}$, Φ has the following property.*

Let $Z \subset [m]$ be any set of size $|Z| \le \varepsilon n$. Let $i_1, \dots, i_\ell \in [m] \setminus Z$ be a sequence of pairwise distinct indices. For $s \in \{1, \dots, \ell\}$, define $N_s := \mathsf{var}(Z) \cup \bigcup_{j=1}^{s-1} \mathsf{var}(c_{i_j})$. If

$$|\mathsf{var}(c_{i_s}) \cap N_s| \ge \lambda \quad \textit{for all } s \in \{1, \dots, \ell\},$$

then $\ell \le \varepsilon n$.

Proof. The proof is almost identical to the proof in [12], though there ε is a constant (independent of n), whereas here we allow it to depend on n. [12, Equation (4)] shows that the probability that Φ does not have the desired property is bounded above by $p_n := [(2e)^{2k}\varepsilon^{\lambda/2}]^{\varepsilon n}$. If $\varepsilon n \ge 10 \log n$, then clearly $p_n = o(1/n)$ since $(2e)^{2k}\varepsilon^{\lambda/2} < 1/e^{1/2}$ by the assumption $\varepsilon^\lambda \le \frac{1}{e}(2e)^{-4k}$. If $1 \le \varepsilon n \le 10 \log n$, then $(2e)^{2k}\varepsilon^{\lambda/2} < n^{-3/2}$ for all sufficiently large n (using $\lambda > 4$) and hence $p_n = o(1/n)$. Finally, for $\varepsilon n < 1$, the lemma follows by the case $\varepsilon n = 1$. $\square$

We will use the following corollary of Lemma 44; note the slightly different conclusion $\ell \le |Z|$ in the end.

Corollary 45. *W.h.p. the random formula Φ has the following property. Let $Z \subset [m]$ be any set of size $|Z| \le 2n/2^{k^{10}}$. Let $i_1, \dots, i_\ell \in [m] \setminus Z$ be a sequence of pairwise distinct indices. For $s \in \{1, \dots, \ell\}$, define $N_s := \mathsf{var}(Z) \cup \bigcup_{j=1}^{s-1} \mathsf{var}(c_{i_j})$. If*

$$|\mathsf{var}(c_{i_s}) \cap N_s| \ge k/10 \quad \textit{for all } s \in \{1, \dots, \ell\},$$

then $\ell \le |Z|$.

Proof. For an integer z satisfying $1 \le z \le 2n/2^{k^{10}}$, let $\mathcal{E}_z$ be the event that there exists a set Z with $|Z| = z$ that does not satisfy the desired property. By Lemma 44 (applied with $\varepsilon = z/n$ and $\lambda = k/10$), we have $\Pr_\Phi(\mathcal{E}_z) = o(1/n)$. Taking a union bound over the values of z yields the corollary. $\square$

Corollary 45 allows us to control the number of bad variables.

Lemma 46. *W.h.p. over the choice of Φ, $|\mathcal{V}_{bad}| \le 4kn/2^{k^{10}}$.*

Proof. W.h.p. Φ satisfies the properties in Lemma 35 and Corollaries 37, 45. By Lemma 35, we have that $|\mathcal{V}_0| \le n/2^{k^{10}}$ and hence by Corollary 37 (applied to $Y = \mathcal{V}_0$), we obtain that $|\mathcal{C}_0| \le 2n/2^{k^{10}}$. By Lemma 44 (applied to $Z = \mathcal{C}_0$), we conclude that $|\mathcal{C}_{\mathrm{bad}}| \le 4n/2^{k^{10}}$ and hence $|\mathcal{V}_{\mathrm{bad}}| \le 4kn/2^{k^{10}}$. $\square$

We can in fact use Corollary 45 to prove the following lemma.

Lemma 47. *W.h.p. over the choice of Φ, for any bad component S, $|S| \le 60\,|\mathsf{HD}(S)|$.*

Proof. W.h.p. we have that Φ satisfies the properties in Lemma 35 and Corollaries 38 and 45. We will show the conclusion of the lemma whenever Φ satisfies these three properties.

Let S be a bad component. If S contains only an isolated variable, it must be a high-degree variable and hence $\mathsf{HD}(S) = S$ (so we are finished). Otherwise, since a bad component is a connected component of variables in $H_{\Phi,\mathrm{bad}}$, the definition of $H_{\Phi,\mathrm{bad}}$ ensures that the bad component has at least $k/10$ high-degree variables. Note that $|\mathsf{HD}(S)| \le |\mathcal{V}_0| \le n/2^{k^{10}}$ by Lemma 35. Applying Corollary 38 with $Y = \mathsf{HD}(S)$, we find that there are at most $\frac{30}{k}\,|\mathsf{HD}(S)|$ clauses that contain at least $k/10$ variables from $\mathsf{HD}(S)$.

Now, we run the process P starting with $\mathsf{HD}(S)$. Take Z to be the set of clauses that contain at least $k/10$ variables from $\mathsf{HD}(S)$ (so, from above, we have $|Z| \le \frac{30}{k}\,|\mathsf{HD}(S)| \le \frac{30}{k}\frac{n}{2^{k^{10}}}$). Applying Corollary 45, we find that the number of clauses c such that $\mathsf{var}(c) \subseteq \mathsf{BC}(\mathsf{HD}(S))$ is at most $2|Z| \le 60\,|\mathsf{HD}(S)|\,/k$. Since $S = \mathsf{BC}(\mathsf{HD}(S))$ by Lemma 43 and each variable in S is contained in some bad clause, we have

$$|S| \le \left| \bigcup_{c \in \mathcal{C}_{\mathrm{bad}}\colon\, \mathsf{var}(c) \cap S \neq \emptyset} \mathsf{var}(c) \right| \le 60\,|\mathsf{HD}(S)|\,. \qquad \square$$

Next, we show that there is no large bad component.

Lemma 48. *W.h.p. over the choice of Φ, every bad component S has size at most $21600k\log n$.*

Proof. Suppose there is a bad component S with size $|S| > 21600k\log n$. Since S is a connected component in $H_{\Phi,\mathrm{bad}}$, S is also a connected set in H_Φ. By Lemma 42, $|\mathsf{HD}(S)| \le \frac{|S|}{21600}$. However by Lemma 47, we have $|S| \le 60\,|\mathsf{HD}(S)|$, which gives a contradiction. □

The following lemma shows that every "large" connected set contains few bad variables.

Lemma 49. *W.h.p. over the choice of Φ, for every connected set S of variables with size at least $21600k\log n$, $|S\cap\mathcal{V}_{bad}| \le |S|/360$.*

Proof. W.h.p. we have that Φ satisfies the properties in Lemmas 42 and 47. We will show the conclusion of the lemma for all such Φ.

For the sake of contradiction, let S be a connected set of variables with size at least $21600k\log n$ and $|S\cap\mathcal{V}_{\mathrm{bad}}| > |S|/360 = 60\delta_0\,|S|$, where $\delta_0 = 1/21600$. Suppose that there are t bad components $S_1, S_2, \ldots, S_t$ intersecting S. Let $S' = S\cup S_1\cup\cdots\cup S_t$ and let $b = |S'\setminus S|/|S|$. Note that S' is a connected set of variables. Also, all variables in $S'\setminus S$ are bad, so $|S'\cap\mathcal{V}_{\mathrm{bad}}| > (60\delta_0 + b)\,|S|$. Thus, by Lemma 47, we have

$$\left|\mathsf{HD}(S')\right| = \sum_{i=1}^{t} |\mathsf{HD}(S_i)| \ge \sum_{i=1}^{t} \frac{|S_i|}{60} = \frac{|S'\cap\mathcal{V}_{\mathrm{bad}}|}{60} > \left(\delta_0 + \frac{b}{60}\right)|S| > \delta_0(1+b)\,|S| = \delta_0\left|S'\right|,$$

which contradicts Lemma 42. □

Lemma 50. *W.h.p. over the choice of Φ, for every connected set of clauses Y such that $|\mathsf{var}(Y)| \ge 21600k\log n$, it holds that $|Y\cap\mathcal{C}_{bad}| \le |Y|/12$.*

Proof. W.h.p. we have that Φ satisfies the properties in Corollary 38 and Lemmas 46 and 49. We will show the conclusion of the lemma for all such Φ.

Let Y be a connected set of clauses such that $|\mathsf{var}(Y)| \ge 21600k\log n$ and let $S = \mathsf{var}(Y)$. Then $|S| \le k\,|Y|$. Since Y is connected, so is S. Let $S_{\mathrm{bad}} = S\cap\mathcal{V}_{\mathrm{bad}}$ and note that, by Lemma 46, $|S_{\mathrm{bad}}| \le |\mathcal{V}_{\mathrm{bad}}| \le 4kn/2^{k^{10}}$. By Lemma 49, we also have that $|S_{\mathrm{bad}}| \le |S|/360$. Note that every $c\in Y\cap\mathcal{C}_{\mathrm{bad}}$ contains at least $k/10$ variables in S_{bad}. Applying Corollary 38 (with the "Y" in the corollary equal to S_{bad}),

$$|Y\cap\mathcal{C}_{\mathrm{bad}}| \le \frac{30\,|S_{\mathrm{bad}}|}{k} \le \frac{|S|}{12k} \le \frac{|Y|}{12}.$$ □

8.5. **Proofs for the coupling tree.** In this section, we prove Lemma 14. For $V\subseteq\mathcal{V}$, let $\Gamma_{H_\Phi}(V) = \cup_{v\in V}\Gamma_{H_\Phi}(v)$ be the neighbourhood of V in H_Φ. Let $\Gamma^+_{H_\Phi}(V) = V\cup\Gamma_{H_\Phi}(V)$ be the extended neighbourhood.

Lemma 51. *W.h.p. over the choice of Φ, every connected set of variables $V\subseteq\mathcal{V}$ satisfies $|\Gamma^+_{H_\Phi}(V)| \le 3k^3\alpha\max\{|V|, k\log n\}$.*

Proof. Let $\delta_0 = 1$, and $\theta_0 = 2k^2\alpha$. Since $\delta_0\theta_0\log(\theta_0/k^2\alpha) > \log\alpha + 3\log k$, w.h.p. we have that Φ satisfies the property in Lemma 41. We will show the conclusion of the lemma for all such Φ.

Let V be a connected set of variables and Y be the set of neighbours of V in the factor graph, i.e., $Y = \{c\in\mathcal{C} \mid \mathsf{var}(c)\cap V \neq \emptyset\}$. Clearly $|\Gamma^+_{H_\Phi}(V)| \le k\,|Y|$ and hence it suffices to show that $|Y| \le 3k^2\alpha\max\{|V|, k\log n\}$. There are two cases depending on the size of V.

- $|V| \ge k\log n$. Since V is a connected set of variables, there exists a set $Y'\subseteq Y$ such that $|V|/k \le |Y'| \le |V|$ and $V\cup Y'$ is connected in the factor graph of Φ. Hence, Y' is a connected set of clauses and $|Y'| \ge \log n$. Let $Z := Y\setminus Y'$. If $|Z| \ge \theta_0\,|V|$, then we obtain a contradiction to Lemma 41 (using the sets $U = V, Y', Z$). Thus, $|Z| \le \theta_0\,|V|$ and $|Y| \le |Y'| + |Z| \le 3k^2\alpha\,|V|$.

- Otherwise $|V| < k \log n$. If $|\Gamma^+_{H_\Phi}(V)| < \lceil k \log n \rceil$ then we are finished. Otherwise, consider an arbitrary connected $V' \supset V$ such that $|V'| = \lceil k \log n \rceil$. By the argument of the previous case, the set of neighbours of V' in the factor graph, denoted Y'', satisfies that $|Y''| \leq 3k^2\alpha\,|V'| \leq 3k^3\alpha \log n$. Thus, $|Y| \leq |Y''| \leq 3k^3\alpha \log n$.

This completes the proof. □

Now we can show Lemma 14, which we restate here for convenience. Recall that Λ^* is from Lemma 10.

Lemma 14. *W.h.p. over the choice of Φ, for every prefix Λ of Λ^*, every node ρ in $\mathbb{T}^\Lambda$ has the property that $|V_{set}(\rho)| \leq 3k^3\alpha L + 1$.*

Proof. W.h.p. we have that Φ satisfies the property in Lemma 51. We will show the conclusion of the lemma for all such Φ.

Let Λ be a prefix of Λ^* and ρ be a node in $\mathbb{T}^\Lambda$. We first claim that $V_{\text{set}}(\rho) \subseteq \Gamma^+_{H_\Phi}(V_I(\rho))$. To prove the claim, we'll consider any $u \in V_{\text{set}}(\rho) \setminus V_I(\rho)$ and we will show that there is a clause c containing u and containing a variable in $V_I(\rho)$.

We first rule out the case that $u = v^*$ by noting (via Property (P1)) that $v^* \in V_I(\rho) \cap V_{\text{set}}(\rho)$.

So consider $u \in V_{\text{set}}(\rho) \setminus V_I(\rho)$ and let ρ' be the ancestor of ρ in the coupling tree such that u is the first variable of ρ'. The definition of the coupling tree guarantees that ρ' is uniquely defined and that it is a proper ancestor of ρ — the definition of "first variable" guarantees that $u \notin V_{\text{set}}(\rho')$, but for all proper descendants ρ''' of ρ', $u \in V_{\text{set}}(\rho''')$.

Let ρ'' be the child of ρ' on the path to ρ. We will show that there is a clause c containing u and containing a variable in $V_I(\rho')$. The claim will then follow from the fact that $V_I(\rho)$ contains $V_I(\rho')$. The existence of such a clause c is immediate from the definition of "first variable" — indeed c is the "first clause" of ρ'. Thus, we have proved the claim.

By (P5), $V_I(\rho)$ is a connected set of variables. Thus, by Lemma 51 and the claim,

$$|V_{\text{set}}(\rho)| \leq |\Gamma^+_{H_\Phi}(V_I(\rho))| \leq 3k^3\alpha \max\{|V_I(\rho)|, k \log n\}.$$

If $\rho \notin \mathcal{T}$, then $|V_I(\rho)| \leq L$ and the lemma holds. Otherwise, apply the above to the parent of ρ, which finishes the proof. □

8.6. **Proofs for $\mathcal{D}(G_\Phi)$.** In this section we show Lemma 27 and Lemma 28.

Lemma 52. *Let G be a connected graph. For any connected induced subgraph $G' = (V', E')$ of $G^{\leq 4}$, there exists a connected induced subgraph of G with size at most $4\,|V'|$ containing all vertices in V'.*

Proof. We do an induction on $\ell = |V'|$. If $\ell = 1$ the claim holds since G' is also an induced subgraph of G. If $\ell > 1$, assume that the claim holds for all induced subgraphs of $G^{\leq 4}$ with at most $\ell - 1$ vertices. Let v be a vertex of G' such that $G'[V' \setminus \{v\}]$ is connected in $G^{\leq 4}$. Thus, by the induction hypothesis, there exists a connected induced subgraph $G'' = (V'', E'')$ of G such that $(V' \setminus \{v\}) \subseteq V''$ and $|V''| \leq 4(\ell - 1)$. Since G' is connected in $G^{\leq 4}$, there exists a vertex $u \in (V' \setminus \{v\})$ such that $\text{dist}_G(u, v) \leq 4$. Let $U = V'' \cup \{\text{vertices on the path from } u \text{ to } v \text{ in } G\}$. Then the induced subgraph in G whose vertex set is U is connected and $|U| \leq 4\ell$. Thus the claim holds for G'. □

Corollary 53. *Let G be a connected graph and $v \in V(G)$ be a vertex. Let $n_{G,\ell}(v)$ denote the number of connected induced subgraphs of G with size ℓ containing v. Then*

$$n_{G^{\leq 4},\ell}(v) \leq 2^{\ell'} n_{G,\ell'}(v) \text{ where } \ell' := \min\{4\ell, |V(G)|\}.$$

Proof. By Lemma 52, for any connected induced subgraph G' of $G^{\leq 4}$ with size ℓ containing v, there exists a connected induced subgraph G'' of G such that $V(G') \subseteq V(G'')$ and $|V(G'')| \leq 4\ell$. In fact we can further assume that $|V(G'')| = \ell'$ since otherwise we can keep adding vertices from neighbours of G'' into G'' until $|V(G'')| = \ell'$. For any such G'' the number of size ℓ subsets containing v (corresponding to potential graphs G' which would be mapped to G'' by the above construction) is at most $\binom{\ell'}{\ell-1} \leq 2^{\ell'}$, giving the desired upper bound. □

Lemma 27. *Let ℓ be an integer which is at least $\log n$. W.h.p. over the choice of Φ, every clause $c \in \mathcal{C}^{\Lambda}_{good}$ has the property that the number of size-ℓ subsets $T \in \mathcal{D}(G_\Phi)$ containing c is at most $(18k^2\alpha)^{4\ell}$.*

Proof. Just combine Corollary 53 with Lemma 40. □

In the remainder of this section, we will focus on showing Lemma 28. We will need the following ingredients.

Lemma 54. *For any set $Y \subseteq \mathcal{C}_{good}$ of good clauses, the size of a maximum independent set in $G_{\Phi,good}[Y]$ is at least $|Y|/(k\Delta)$.*

Proof. Let c be a clause in Y. Note that c contains at most k variables in $\mathcal{V}_{\mathrm{good}}$ and each variable in $\mathcal{V}_{\mathrm{good}}$ is contained in at most Δ clauses. So the degree of c in $G_{\Phi,\mathrm{good}}$ is at most $k(\Delta-1)$. The result follows since every n-vertex graph of maximum degree d contains an independent set of size at least $n/(d+1)$. □

We will also use the following properties of $\mathcal{F}(\rho)$.

Lemma 55. *If ρ is a node of the coupling tree, then the following properties hold.*

(1) $G_\Phi^{\leq 2}[\mathcal{F}(\rho)]$ is connected.
(2) $|\mathcal{F}(\rho)| \geq |V_I(\rho)|/k$.

Proof. We show Item (1) by induction on the size of $V_{\mathrm{set}}(\rho)$. The base case where $|V_{\mathrm{set}}(\rho)| = 1$ is trivial since, in this case, $\rho = \rho^*$ and $\mathcal{F}(\rho^*)$ is the set of clauses containing v^*. For the inductive step, we consider a node $\rho' = \rho_{\tau_1,\tau_2}$ being created as a new child of ρ by Algorithm 1 and we consider how clauses are added to $\mathcal{F}(\rho')$. We show that each part of the algorithm that adds clauses to $\mathcal{F}(\rho')$ maintains the property that $G_\Phi^{\leq 2}[\mathcal{F}(\rho')]$ is connected. Before Line 5, this holds by the inductive hypothesis.

- First, consider the addition of clauses in Line 8. All clauses c' that are added by this line contain the first variable u of ρ which is in the first clause c of ρ so to finish it suffices to show that $\mathcal{F}(\rho)$ has a clause which shares a variable with c. Since $\mathsf{var}(c) \cap V_I(\rho)$ is non-empty, it suffices to show that every variable in $V_I(\rho)$ is contained in a clause in $\mathcal{F}(\rho)$. This is true by Property (P6).
- Next, consider the addition of clauses in Line 18. It is important to note that, after the loop containing Line 8, Property (P6) has been re-established. That is, for any $u'' \in V_I$ there is a clause $c'' \in \mathcal{F}$ such that $u'' \in \mathsf{var}(c'')$. All clauses c' added to $\mathcal{F}$ in Line 18 have variables in V_I so the introduction of c' leaves $G_\Phi^{\leq 2}[\mathcal{F}]$ connected. Moreover, the subsequent addition of variables from $\mathsf{var}(c')$ to V_I maintains Property (P6).
- Finally, consider the addition of clauses in Line 24. As in the previous case, Property (P6) guarantees that the introduction of clauses to $\mathcal{F}$ leaves $G_\Phi^{\leq 2}[\mathcal{F}]$ connected. Moreover, the subsequent addition of variables to V_I maintains Property (P6).

Item (2) is a direct consequence of (P6). □

We also need the following expansion property (which is a strengthening of Lemma 36 in the case that $b = k$).

Lemma 56 ([12, Lemma 2.3])**.** *For all sufficiently large k, w.h.p. over the choice of Φ, for any $Y \subseteq \mathcal{C}$ such that $|Y| \leq n/k^2$, $|\mathsf{var}(Y)| \geq 0.9k\,|Y|$.*

We are now ready to prove Lemma 28, which we restate here.

Lemma 28. *W.h.p. over the choice of Φ, every node ρ in $\mathbb{T}^{\Lambda}$ with $|V_I(\rho)| \geq L$ has the property that there is a set $T \subseteq \mathcal{F}(\rho)$ containing c^* such that $T \in \mathcal{D}(G_\Phi)$, $|T| = C_0\lceil \log(n/\varepsilon)\rceil$ and $|T \cap \mathcal{C}_{bad}| \leq |T|/3$.*

Proof. W.h.p. we have that Φ satisfies the properties in Lemmas 50 and 56. We will show the conclusion of the lemma for all such Φ. Let $\rho \in \mathcal{T}$ be a node of $\mathbb{T}^{\Lambda}$ with $|V_I(\rho)| \geq L$. For a good clause c, let $\Gamma^+_{\text{good}}(c)$ be the set consisting of c and all of its neighbours in $G_{\Phi,\text{good}}$. Recall also from Section 5 that c^* is a good clause (being the first clause of the root node ρ^*).

Let $U = \mathcal{F}(\rho) \setminus (\mathcal{C}_{\text{bad}} \cup \Gamma^+_{\text{good}}(c^*))$ and I be a maximum independent set of $G_{\Phi,\text{good}}[U]$. We let $T = I \cup \{c^*\} \cup (\mathcal{F}(\rho) \cap \mathcal{C}_{\text{bad}})$. By Lemma 54 we have $|I| \geq |U|/(k\Delta)$. By construction T contains c^*.

Next, we show that $T \in \mathcal{D}(G_\Phi)$. Since $\Gamma^+_{\text{good}}(c^*)$ and U are disjoint, and I is an independent set of $G_{\Phi,\text{good}}[U]$, for any $c_1, c_2 \in T$ we have that $\mathsf{var}(c_1) \cap \mathsf{var}(c_2) \cap \mathcal{V}_{\text{good}} = \emptyset$ (note that clauses in $\mathcal{C}_{\text{bad}}$ only have bad variables). It therefore suffices to show that T is connected in $G_\Phi^{\leq 4}$.

Suppose for contradiction that T is not connected in $G_\Phi^{\leq 4}$. Then there exists a partition (S_1, S_2) of T such that $S_1 \cup S_2 = T$, $S_1 \cap S_2 = \emptyset$ and $\min_{c_1 \in S_1, c_2 \in S_2} \text{dist}_{G_\Phi}(c_1, c_2) \geq 5$. Let $S_i' = (\cup_{c \in S_i} \Gamma^+_{\text{good}}(c)) \cap \mathcal{F}(\rho)$ for $i = 1, 2$. Then we have $\min_{c_1 \in S_1', c_2 \in S_2'} \text{dist}_{G_\Phi}(c_1, c_2) \geq 3$.

Since I is a maximum independent set of $G_{\Phi,\text{good}}[U]$, every clause c' in U has $\Gamma^+_{\text{good}}(c') \cap I \neq \emptyset$. So $U \subseteq \cup_{c \in I} \Gamma^+_{\text{good}}(c)$. Thus,

$$S_1' \cup S_2' \supseteq U \cup (\mathcal{C}_{\text{bad}} \cap \mathcal{F}(\rho)) \cup (\Gamma^+_{\text{good}}(c^*) \cap \mathcal{F}(\rho)) = \mathcal{F}(\rho).$$

However, $G_\Phi^{\leq 2}[\mathcal{F}(\rho)]$ is connected by Item (1) of Lemma 55, which contradicts

$$\min_{c_1 \in S_1', c_2 \in S_2'} \text{dist}_{G_\Phi}(c_1, c_2) \geq 3.$$

Thus, we have finished showing that $T \in \mathcal{D}(G_\Phi)$.

Now, observe the following lower bound on the size of T:

$$\begin{aligned}|T| = |I| + |\mathcal{F}(\rho) \cap \mathcal{C}_{\text{bad}}| + 1 &\geq \frac{|\mathcal{F}(\rho)| - |\mathcal{F}(\rho) \cap \mathcal{C}_{\text{bad}}| - k\Delta}{k\Delta} + |\mathcal{F}(\rho) \cap \mathcal{C}_{\text{bad}}| + 1 \\ &\geq \frac{|\mathcal{F}(\rho)|}{k\Delta} \geq \frac{|V_I(\rho)|}{k^2\Delta} \geq C_0 \lceil \log(n/\varepsilon) \rceil,\end{aligned}$$

where in the second to last inequality we used Item (2) of Lemma 55. If $|T| > C_0 \lceil \log(n/\varepsilon) \rceil$, we make the size of T exactly equal to $C_0 \lceil \log(n/\varepsilon) \rceil$ by removing some clauses from it. Note that any subset of T satisfies Item (1) of the definition of $\mathcal{D}(G_\Phi)$ (cf. Definition 26). To maintain the connectedness of T in $G_\Phi^{\leq 4}$, consider an arbitrary spanning tree of the subgraph $G_\Phi^{\leq 4}[T]$; remove leaf vertices of the tree from T until until $|T| = C_0 \lceil \log(n/\varepsilon) \rceil$. By construction, the remaining T is still connected in $G_\Phi^{\leq 4}$ and hence is in $\mathcal{D}(G_\Phi)$.

Finally, we show that $|T \cap \mathcal{C}_{\text{bad}}| \leq \frac{|T|}{3}$. Since T is connected in $G_\Phi^{\leq 4}$, Lemma 52 implies that there exists a connected induced set T' of vertices of G_Φ such that $T \subseteq T'$ and $|T'| \leq 4\,|T|$. Lemma 56 implies that $|\mathsf{var}(T')| \geq 0.9k\,|T'| \geq 0.9k|T| = 0.9kC_0 \lceil \log(n/\varepsilon) \rceil > 21600k \log n$. Thus Lemma 50 implies that $|T' \cap \mathcal{C}_{\text{bad}}| \leq \frac{|T'|}{12}$. We conclude that

$$|T \cap \mathcal{C}_{\text{bad}}| \leq |T' \cap \mathcal{C}_{\text{bad}}| \leq \frac{|T'|}{12} \leq \frac{|T|}{3}.$$

This completes the proof. ∎

9. Proof of Theorem 1

In order to estimate $\Omega(\Phi)$, we use self-reducibility to calculate the marginal probability of the partial assignment Λ^* from Lemma 10. This marginal probability is $|\Omega^{\Lambda^*}|/|\Omega|$. By Lemma 48, w.h.p. over the choice of Φ, $|\Omega^{\Lambda^*}|$ can be computed in polynomial time. This is because Λ^* satisfies all of the good clauses (by Lemma 10). The remaining clauses are bad, and Lemma 48 guarantees that all bad components have $O(\log n)$ size, so their satisfying assignments can be counted by brute force.

Lemma 57. *There is a deterministic algorithm that takes as input a k-CNF formula Φ on n Boolean variables with m clauses, a partial assignment Λ of Φ and an accuracy parameter $\varepsilon > 0$. It returns a rational value p and runs in time $\mathrm{poly}(n, 1/\varepsilon)$. If $\Phi = \Phi(k, n, m)$ then, w.h.p., the guarantees of all of our lemmas apply. In this case, as long as Λ is a prefix of the partial assignment Λ^* from Lemma 10, the output satisfies $e^{-\varepsilon/n} p \leq |\Omega_1^\Lambda|/|\Omega_2^\Lambda| \leq e^{\varepsilon/n} p$.*

Proof. Let v^* be the first unassigned variable in $\Lambda^* \setminus \Lambda$. By Lemma 14, the depth of $\mathbb{T}^\Lambda$ is at most $3k^3\alpha L + 1 = O(\log(n/\varepsilon))$. Thus, the number of nodes of $\mathbb{T}^\Lambda$ is bounded by a polynomial in n/ε.

After constructing $\mathbb{T}^\Lambda$, the algorithm constructs the linear program from Section 6. Lemma 34 guarantees that, given bounds $r_{\mathsf{lower}} \leq r_{\mathsf{upper}}$, if the LP has a solution $\mathbf{P}$ using r_{lower} and r_{upper}, then $e^{-\varepsilon/(3n)} r_{\mathsf{lower}} \leq |\Omega_1^\Lambda|/|\Omega_2^\Lambda| \leq e^{\varepsilon/(3n)} r_{\mathsf{upper}}$. Thus, the task of the algorithm is to find initial bounds $r_{\mathsf{lower}} \leq r_{\mathsf{upper}}$ for which there is a feasible solution and then (by binary search) to bring r_{lower} and r_{upper} closer together, to obtain a more accurate estimate. This is done in Algorithm 2.

The algorithm is based on the premise that there is a feasible solution with $r_{\mathsf{lower}} = (3s-1)/(3s+1)$ and $r_{\mathsf{upper}} = (3s+1)/(3s-1)$, so we next establish this fact. This follows from Lemma 9, which guarantees that

$$\frac{1-\frac{1}{3s}}{1+\frac{1}{3s}} \leq \frac{|\Omega_1^\Lambda|}{|\Omega_2^\Lambda|} \leq \frac{1+\frac{1}{3s}}{1-\frac{1}{3s}}.$$

Algorithm 2 Estimate $|\Omega_1^\Lambda| \,/\, |\Omega_2^\Lambda|$

$p_{\mathsf{lower}} \leftarrow \frac{3s-1}{3s+1}$;
$p_{\mathsf{upper}} \leftarrow \frac{3s+1}{3s-1}$;
while $p_{\mathsf{upper}} > e^{\varepsilon/(3n)} p_{\mathsf{lower}}$ **do**
    $r_{\mathsf{lower}} \leftarrow p_{\mathsf{lower}}$;
    $r_{\mathsf{upper}} \leftarrow (p_{\mathsf{lower}} + p_{\mathsf{upper}})/2$;
    if the LP described in Section 6 has a feasible solution **then**
        $p_{\mathsf{upper}} \leftarrow r_{\mathsf{upper}}$;
    else
        $p_{\mathsf{lower}} \leftarrow r_{\mathsf{upper}}$;
    end if
end while
return $p \leftarrow (p_{\mathsf{lower}} + p_{\mathsf{upper}})/2$;

We next explore the accuracy of the output. Applying Lemma 24 and Lemma 34, after each step of the while loop of Algorithm 2, it holds that $e^{-\varepsilon/(3n)} p_{\mathsf{lower}} \leq |\Omega_1^\Lambda| \,/\, |\Omega_2^\Lambda| \leq e^{\varepsilon/(3n)} p_{\mathsf{upper}}$. Also, $p_{\mathsf{upper}} \leq e^{\varepsilon/(3n)} p_{\mathsf{lower}}$ when the algorithm terminates, so the output p satisfies $e^{-\varepsilon/n} p \leq \frac{|\Omega_1^\Lambda|}{|\Omega_2^\Lambda|} \leq e^{\varepsilon/n} p$.

Finally, we explore the running time of the algorithm. Since each iteration of the while loop takes polynomial time, the main issue is determining how many times the while loop executes. Note that in the beginning $p_{\mathsf{upper}} - p_{\mathsf{lower}} \leq 4/(3s-1)$, and just before the last iteration,

$$p_{\mathsf{upper}} - p_{\mathsf{lower}} \geq (e^{\varepsilon/(3n)} - 1) p_{\mathsf{lower}} \geq \frac{\varepsilon(3s-1)}{3n(3s+1)}.$$

This difference halves with each iteration. Thus the algorithm solves the LP at most $O(\log(n/\varepsilon))$ times. Since the size of LP is bounded by a polynomial in n/ε and the LP can be solved in polynomial time, the whole of the algorithm runs in polynomial time, as required. □

We now prove Theorem 1.

Theorem 1. *There is a polynomial-time algorithm $\mathcal{A}$ and there are two constants $r > 0$ and $k_0 \geq 3$ such that, for all $k \geq k_0$ and all $\alpha < 2^{rk}$, the following holds w.h.p. over the choice of the random k-SAT formula $\Phi = \Phi(k, n, \lfloor \alpha n \rfloor)$. The algorithm $\mathcal{A}$, given as input the formula Φ and a rational $\varepsilon > 0$, outputs in time $poly(n, 1/\varepsilon)$ a number Z that satisfies $e^{-\varepsilon}|\Omega(\Phi)| \leq Z \leq e^{\varepsilon}|\Omega(\Phi)|$.*

Proof. Let $\Phi = \Phi(k, n, m)$ be a random formula and $\Omega = \Omega(\Phi)$. The algorithm first computes $\mathcal{V}_{\text{good}}, \mathcal{V}_{\text{bad}}, \mathcal{C}_{\text{good}}$ and $\mathcal{C}_{\text{bad}}$ in time $poly(n)$. Then using Lemma 8 and Lemma 10, it can compute $\mathcal{V}_{\mathsf{marked}}$ and Λ^* in polynomial-time. W.h.p. over the choice of Φ, the guarantees of all of our lemmas apply. Let us suppose that this happens (otherwise, the algorithm fails and outputs an arbitrary number).

Suppose Λ^* gives values to the j variables $v_1, v_2, \ldots, v_j$. By Lemma 57, for any $i \in \{1, \ldots, j\}$ and any prefix $\Lambda_i \colon \{v_1, v_2, \ldots, v_{i-1}\} \to \{\mathsf{T}, \mathsf{F}\}$ of Λ^*, it takes $poly(n, 1/\varepsilon)$ time to compute a number p_i satisfying $e^{-\varepsilon/n} p_i \leq |\Omega_1^{\Lambda_i}|/|\Omega_2^{\Lambda_i}| \leq e^{\varepsilon/n} p_i$. If $\Lambda^*(v_i) = \mathsf{T}$ let $q_i = \frac{p_i}{1+p_i}$. Otherwise, let $q_i = \frac{1}{1+p_i}$. Let $\Lambda_{j+1} = \Lambda^*$.

Suppose that $\Lambda^*(v_i) = \mathsf{T}$. Then

$$\frac{|\Omega^{\Lambda_i}|}{|\Omega^{\Lambda_{i+1}}|} = \frac{|\Omega_1^{\Lambda_i}| + |\Omega_2^{\Lambda_i}|}{|\Omega_1^{\Lambda_i}|} = 1 + \frac{|\Omega_2^{\Lambda_i}|}{|\Omega_1^{\Lambda_i}|} \leq 1 + \frac{e^{\varepsilon/n}}{p_i}.$$

Thus,

$$\frac{|\Omega^{\Lambda_{i+1}}|}{|\Omega^{\Lambda_i}|} \geq \frac{p_i}{p_i + e^{\varepsilon/n}} \geq \frac{p_i}{e^{\varepsilon/n} p_i + e^{\varepsilon/n}} = e^{-\varepsilon/n} q_i.$$

A similar calculation for the case where $\Lambda^*(v_i) = \mathsf{F}$ and a similar calculation for the lower bound give the following.

$$e^{-\varepsilon/n} q_i \leq \frac{|\Omega^{\Lambda_{i+1}}|}{|\Omega^{\Lambda_i}|} \leq e^{\varepsilon/n} q_i,$$

and thus

$$e^{-\varepsilon} \prod_{i=1}^{j} q_i \leq \frac{|\Omega^{\Lambda^*}|}{|\Omega|} \leq e^{\varepsilon} \prod_{i=1}^{j} q_i \, .$$

Since all good clauses are satisfied by Λ^*, $\mathcal{C}^{\Lambda^*}$ consists only of bad clauses. Also, by Lemma 48, every bad component of variables has size at most $21600k \log n$, so $\mathcal{C}^{\Lambda^*}$ can be divided into disjoint subsets where each subset of clauses contains $O(\log n)$ variables. The algorithm can compute the number of satisfying assignments of each subset by brute force in time $poly(n)$. Then $|\Omega^{\Lambda^*}|$ is the product of these numbers.

Combining all of the above, our algorithm outputs $Z = |\Omega^{\Lambda^*}| \prod_{i=1}^{j} q_i^{-1}$, which satisfies $e^{-\varepsilon}|\Omega| \leq Z \leq e^{\varepsilon}|\Omega|$. ◻

Acknowledgement

We thank Andrei Constantinescu and anonymous referees for various helpful comments which improved the accuracy of the paper.

Appendix A. Notation reference

Name	Description	Reference
Formula related		
$\mathcal{C}$	The clause set of Φ where $m = \lvert\mathcal{C}\rvert = \lfloor n\alpha \rfloor$.	Section 3
$\mathcal{C}_{\text{good}}$,$\mathcal{C}_{\text{bad}}$	Good and bad clauses, a partition of $\mathcal{C}$	Section 3
$\mathcal{V}$	The variable set of Φ where $n = \lvert\mathcal{V}\rvert$.	Section 3
$\mathcal{V}_{\text{good}}$,$\mathcal{V}_{\text{bad}}$	Good and bad variables, a partition of $\mathcal{V}$	Section 3
Δ	The high degree threshold, set to $2^{k/300}$.	Definition 2
G_Φ	The dependency graph of $\mathcal{C}$, which contains a subgraph $G_{\Phi,\text{good}}$.	Definition 4
H_Φ	The dependency graph of $\mathcal{V}$, which contains a subgraph $H_{\Phi,\text{bad}}$.	Definition 5
$\mathcal{D}(G_\Phi)$	A set of subsets $T \subseteq V(G_\Phi)$ satisfying some properties	Definition 26
Local lemma		
Ω^*	All assignments $\mathcal{V} \to \{\mathsf{T}, \mathsf{F}\}$	Definition 6
μ_A	Uniform distribution over $A \subseteq \Omega^*$	Definition 6
$\mathsf{marked}(c)$	Marked variables in clause $c \in \mathcal{C}$	Section 4
$\mathcal{V}_{\mathsf{marked}}$	All marked variables	Section 4
Ω	The set of all satisfying assignments	Section 4
Φ^Λ	The formula Φ simplified under Λ	Section 4
$\mathcal{C}^\Lambda$	Remaining clauses under Λ. Similar notations include $\mathcal{V}^\Lambda$, $\mathcal{C}^\Lambda_{\text{good}}$. Note that $\mathcal{V}^\Lambda_{\text{bad}} = \mathcal{V}_{\text{bad}}$ and $\mathcal{C}^\Lambda_{\text{bad}} = \mathcal{C}_{\text{bad}}$.	Section 4
s	Local uniformity parameter $s := 2^{k/4}/(ek\Delta)$.	Lemma 9
Λ^*	A particular "nice" partial assignment.	Lemma 10
Coupling tree		
L	Truncation depth, set to $C_0(3k^2\Delta)\lceil\log(n/\varepsilon)\rceil$, where C_0 is a sufficiently large integer (independent of Φ, k and n).	Definition 12
$\mathcal{L}$	The set of leaves of the coupling tree.	Definition 12
$\mathcal{T}$	The set of truncating nodes of the coupling tree.	Definition 12
$\mathcal{L}^*$	$\mathcal{L}^* = \mathcal{L} \cup \mathcal{T}$.	Definition 12

References

[1] Emmanuel Abbe and Andrea Montanari. On the concentration of the number of solutions of random satisfiability formulas. *Random Struct. Algorithms*, 45(3):362–382, 2014. 1

[2] Dimitris Achlioptas and Amin Coja-Oghlan. Algorithmic barriers from phase transitions. In *FOCS*, pages 793–802. IEEE Computer Society, 2008. 1

[3] Dimitris Achlioptas, Amin Coja-Oghlan, Max Hahn-Klimroth, Joon Lee, Noela Müller, Manuel Penschuck, and Guangyan Zhou. The random 2-SAT partition function. *arXiv e-prints*, abs/2002.03690, 2020. 1

[4] Dimitris Achlioptas and Cristopher Moore. The asymptotic order of the random k-SAT threshold. In *FOCS*, pages 779–788. IEEE Computer Society, 2002. 1

[5] Dimitris Achlioptas and Yuval Peres. The threshold for random k-SAT is $2^k(\ln 2 - o(k))$. In *STOC*, pages 223–231. ACM, 2003. 1

[6] Noga Alon. A parallel algorithmic version of the Local Lemma. *Random Struct. Algorithms*, 2(4):367–378, 1991. 15

[7] Ivona Bezáková, Andreas Galanis, Leslie Ann Goldberg, Heng Guo, and Daniel Štefankovič. Approximation via correlation decay when strong spatial mixing fails. *SIAM J. Comput.*, 48(2):279–349, 2019. 4

[8] Antonio Blanca, Andreas Galanis, Leslie Ann Goldberg, Daniel Štefankovič, Eric Vigoda, and Kuan Yang. Sampling in uniqueness from the Potts and random-cluster models on random regular graphs. *SIAM Journal on Discrete Mathematics*, 34(1):742–793, 2020. 2

[9] Karthekeyan Chandrasekaran, Navin Goyal, and Bernhard Haeupler. Deterministic algorithms for the Lovász local lemma. *SIAM J. Comput.*, 42(6):2132–2155, 2013. 6, 8

[10] Amin Coja-Oghlan. A better algorithm for random k-SAT. *SIAM J. Comput.*, 39(7):2823–2864, 2010. 1

[11] Amin Coja-Oghlan. Belief propagation guided decimation fails on random formulas. *J. ACM*, 63(6):Art. 49, 55, 2017. 1

[12] Amin Coja-Oghlan and Alan Frieze. Analyzing Walksat on random formulas. *SIAM J. Comput.*, 43(4):1456–1485, 2014. 24, 25, 28

[13] Amin Coja-Oghlan, Amir Haqshenas, and Samuel Hetterich. `Walksat` stalls well below satisfiability. *SIAM J. Discrete Math.*, 31(2):1160–1173, 2017. 1

[14] Amin Coja-Oghlan, Noëla Müller, and Jean B. Ravelomanana. Belief propagation on the random k-sat model. *CoRR*, abs/2011.02303, 2020. 2

[15] Amin Coja-Oghlan and Konstantinos Panagiotou. The asymptotic k-SAT threshold. *Adv. Math.*, 288:985–1068, 2016. 1, 5

[16] Amin Coja-Oghlan and Daniel Reichman. Sharp thresholds and the partition function. *Journal of Physics: Conference Series*, 473:012015, 2013. 1

[17] Amin Coja-Oghlan and Nick Wormald. The number of satisfying assignments of random regular k-SAT formulas. *Combin. Probab. Comput.*, 27(4):496–530, 2018. 1

[18] Jian Ding, Allan Sly, and Nike Sun. Proof of the satisfiability conjecture for large k. In *STOC*, pages 59–68. ACM, 2015. 1, 6

[19] Charilaos Efthymiou. A simple algorithm for sampling colorings of $G(n, d/n)$ up to the Gibbs uniqueness threshold. *SIAM Journal on Computing*, 45(6):2087–2116, 2016. 2

[20] Charilaos Efthymiou, Thomas P. Hayes, Daniel Štefankovič, and Eric Vigoda. Sampling random colorings of sparse random graphs. In *SODA*, pages 1759–1771. SIAM, 2018. 2

[21] Paul Erdős and László Lovász. Problems and results on 3-chromatic hypergraphs and some related questions. *Infinite and finite sets, volume 10 of Colloquia Mathematica Societatis János Bolyai*, pages 609–628, 1975. 5

[22] Weiming Feng, Heng Guo, Yitong Yin, and Chihao Zhang. Fast sampling and counting k-SAT solutions in the local lemma regime. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2020, pages 854–867, 2020. 3

[23] Ehud Friedgut. Sharp thresholds of graph properties, and the k-sat problem. *J. Amer. Math. Soc.*, 12(4):1017–1054, 1999. With an appendix by Jean Bourgain. 1

[24] Heng Guo, Mark Jerrum, and Jingcheng Liu. Uniform sampling through the Lovász local lemma. *J. ACM*, 66(3):18:1–18:31, 2019. 3

[25] Heng Guo, Chao Liao, Pinyan Lu, and Chihao Zhang. Counting hypergraph colorings in the local lemma regime. *SIAM J. Comput.*, 48(4):1397–1424, 2019. 15, 16

[26] Bernhard Haeupler, Barna Saha, and Aravind Srinivasan. New constructive aspects of the Lovász local lemma. *J. ACM*, 58(6):Art. 28, 28, 2011. 5

[27] Jonathan Hermon, Allan Sly, and Yumeng Zhang. Rapid mixing of hypergraph independent sets. *Random Struct. Algorithms*, 54(4):730–767, 2019. 4

[28] Samuel Hetterich. Analysing survey propagation guided decimationon random formulas. In *ICALP*, volume 55 of *LIPIcs*, pages 65:1–65:12. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016. 1

[29] Vishesh Jain, Huy Tuan Pham, and Thuy Duong Vuong. On the sampling Lovász local lemma for atomic constraint satisfaction problems. *CoRR*, abs/2102.08342, 2021. 3

[30] Lefteris M. Kirousis, Evangelos Kranakis, Danny Krizanc, and Yannis C. Stamatiou. Approximating the unsatisfiability threshold of random formulas. *Random Structures Algorithms*, 12(3):253–269, 1998. 1

[31] Chao Liao, Jiabao Lin, Pinyan Lu, and Zhenyu Mao. Counting independent sets and colorings on random regular bipartite graphs. In *APPROX-RANDOM*, volume 145 of *LIPIcs*, pages 34:1–34:12. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. 2

[32] Marc Mézard, Thierry Mora, and Riccardo Zecchina. Clustering of solutions in the random satisfiability problem. *Phys. Rev. Lett.*, 94:197205, 2005. 1

[33] Marc Mézard, Giorgio Parisi, and Riccardo Zecchina. Analytic and algorithmic solution of random satisfiability problems. *Science*, 297(5582):812–815, 2002. 1

[34] Michael Mitzenmacher and Eli Upfal. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, 2005. 20

[35] Ankur Moitra. Approximate counting, the Lovász local lemma, and inference in graphical models. *J. ACM*, 66(2):Art. 10, 25, 2019. 2, 3, 4, 6, 15

[36] Andrea Montanari and Devavrat Shah. Counting good truth assignments of random k-SAT formulae. In *SODA*, pages 1255–1264. SIAM, 2007. 1, 2

[37] Robin A. Moser and Gábor Tardos. A constructive proof of the general Lovász local lemma. *J. ACM*, 57(2):Art. 11, 15, 2010. 5

[38] Elchanan Mossel and Allan Sly. Exact thresholds for Ising-Gibbs samplers on general graphs. *Ann. Probab.*, 41(1):294–328, 2013. 2

[39] Jeanette Schmidt-Pruzan and Eli Shamir. Component structure in the evolution of random hypergraphs. *Combinatorica*, 5(1):81–94, 1985. 2

[40] Allan Sly, Nike Sun, and Yumeng Zhang. The number of solutions for random regular NAE-SAT. In *FOCS*, pages 724–731. IEEE Computer Society, 2016. 1
[41] Joel Spencer. Asymptotic lower bounds for Ramsey functions. *Discrete Math.*, 20(1):69–76, 1977. 5
[42] Yitong Yin and Chihao Zhang. Sampling in Potts model on sparse random graphs. In *APPROX-RANDOM*, volume 60 of *LIPIcs*, pages 47:1–47:22. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016. 2

(Andreas Galanis, Leslie Ann Goldberg) Department of Computer Science, University of Oxford, OX1 3QD, United Kingdom.
Email address: `{andreas.galanis,leslie.goldberg}@cs.ox.ac.uk`

(Heng Guo) School of Informatics, University of Edinburgh, Informatics Forum, Edinburgh, EH8 9AB, United Kingdom.
Email address: `hguo@inf.ed.ac.uk`

(Kuan Yang) John Hopcroft Center for Computer Science, Shanghai Jiao Tong University, 800 Dongchuan Road, Shanghai, 200240, China.
Email address: `kuan.yang@sjtu.edu.cn`