Solving Non-uniform Planted and Filtered Random SAT Formulas Greedily

Tobias Friedrich[1], Frank Neumann[2], Ralf Rothenberger[1], and Andrew M. Sutton[3](✉)

[1] Hasso Plattner Institute, University of Potsdam, Potsdam, Germany
{tobias.friedrich,ralf.rothenberger}@hpi.de
[2] The University of Adelaide, Adelaide, Australia
frank.neumann@adelaide.edu.au
[3] University of Minnesota Duluth, Duluth, USA
amsutton@d.umn.edu

Abstract. Recently, there has been an interest in studying non-uniform random k-satisfiability (k-SAT) models in order to address the non-uniformity of formulas arising from real-world applications. While uniform random k-SAT has been extensively studied from both a theoretical and experimental perspective, understanding the algorithmic complexity of heterogeneous distributions is still an open challenge. When a sufficiently dense formula is guaranteed to be satisfiable by conditioning or a planted assignment, it is well-known that uniform random k-SAT is easy on average. We generalize this result to the broad class of non-uniform random k-SAT models that are characterized only by an ensemble of distributions over variables with a mild balancing condition. This balancing condition rules out extremely skewed distributions in which nearly half the variables occur less frequently than a small constant fraction of the most frequent variables, but generalizes recently studied non-uniform k-SAT distributions such as power-law and geometric formulas. We show that for all formulas generated from this model of at least logarithmic densities, a simple greedy algorithm can find a solution with high probability.

As a side result we show that the total variation distance between planted and filtered (conditioned on satisfiability) models is $o(1)$ once the planted model produces formulas with a unique solution with probability $1 - o(1)$. This holds for all random k-SAT models where the signs of variables are drawn uniformly and independently at random.

Keywords: Random k-SAT · Planted k-SAT · Non-uniform variable distribution · Greedy algorithm · Local search

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 416061626.

© Springer Nature Switzerland AG 2021
C.-M. Li and F. Manyà (Eds.): SAT 2021, LNCS 12831, pp. 188–206, 2021.
https://doi.org/10.1007/978-3-030-80223-3_13

1 Introduction

Propositional satisfiability is one of the most intensively studied topics in theoretical computer science and artificial intelligence. Motivated by the desire to understand the hardness of typical propositional formulas, random satisfiability models were developed [23]. The archetype of these random structures is uniform random k-SAT: a family of distributions over formulas, parameterized by length, in conjunctive normal form with k literals in each clause. A vast number of compelling algorithmic hardness results both theoretical [13–16] and experimental [32,35] has developed from this field.

Despite bringing our understanding of the working principles of SAT solvers into sharper focus, a major drawback of the uniform random model is that it does not typically produce formulas that are similar to ones that come from applications. Thus it is not always clear how hardness results on the uniform random model might translate to other distributions. Recently, an effort has emerged to bridge this gap between the homogeneity of uniform random formulas and heterogeneous models of random satisfiability [2,9,10,17,34]. Moreover, specific properties of industrial instances have been identified, and non-uniform distributions have been subsequently introduced to produce such structures. Notable examples include the community attachment model [27] to address modularity, and the popularity-similarity model [28] to address locality.

Ansótegui et al. [5] studied the constraint graphs of industrial propositional formulas, and found that many reveal a *power law* degree distribution, while the variable degrees of formulas drawn from the uniform random k-SAT model are distributed binomially. To address this, they introduced a non-uniform random power law model that induces power law degree distributions. Other researcher have also noted that real-world formulas (especially those derived from bounded model checking) exhibit such heavy-tailed degree distributions [9]. Moreover, empirical results suggest that solvers specialized for industrial instances tend to perform better on formulas drawn from a power law model than on formulas drawn from a uniform model [3–6,8]. Non-uniform random k-SAT models for which the degree distribution follows a geometric law have also been introduced [6].

It is often difficult to understand how algorithmic results on uniform distributions translate to non-uniform models. We use a general variable distribution framework: random k-SAT models are described by an arbitrary ensemble of variable distributions $(\vec{p}_n)_{n\in\mathbb{N}}$ and the clauses are constructed by drawing variables from $\vec{p}_n$. This framework has recently gained interest in the SAT community. For example, it was shown that under some mild conditions on $\vec{p}_n$, the well-known sharpness result of Friedgut [24] generalizes to the non-planted version of this framework [26]. This line of work can help us understand if k-SAT instances with non-uniform variable distributions are easier to solve. If so, which distributions make them easier and why? If not, which other features of industrial instances are important to make them easily solvable?

Results. In this paper, we show that a result for uniform *planted SAT* models, in which a satisfying assignment is hidden, generalizes to a planted version of the non-uniform framework described earlier. In particular, we generalize an early result of Koutsoupias and Papadimitriou [30] to non-uniform planted SAT distributions. We also improve their lower bound on the density threshold by an $n/\log n$ factor. Distributions for which our results hold include recently introduced non-uniform random satisfiability models such as those with power law degree distributions and geometric degree distributions [6]. For those two models in particular only $\Omega(n \log n)$ clauses suffice to find a satisfying assignment with a simple greedy algorithm with high probability.

Furthermore, we investigate the relation between planted and filtered models. Here, filtered means any random SAT model, where we condition on the generated formulas to be satisfiable. We show a result for all k-SAT models in which the signs of variables are chosen uniformly and independently at random for each clause. This result states that, if the planted model asymptotically almost surely[1] generates formulas with a unique solution (the planted solution) at some constraint density m/n, then the total variation distance between the planted and the filtered model at that density is $o(1)$. This means, our results for non-uniform planted SAT transfer to the corresponding filtered models.

1.1 Planted *k*-SAT

Planted distributions are a common modification to average-case problem distributions for combinatorial problems in which instances are generated together with a solution. A motivation for studying planted distributions is that if no efficient algorithms exist for solving an instance, then the instance-solution pairs comprise a one-way function [22], which has important implications for cryptography.

The planted 3-SAT model has been studied in the context of the warning propagation algorithm, and noted for its similarity to low-density parity check codes [19]. The authors show that warning propagation can solve planted 3-SAT formulas with constant constraint density. Berthet [7] considers the problem of *detecting* whether a formula is drawn from the uniform or the planted distribution in the context of hypothesis testing.

Achlioptas, Jia, and Moore [1] analyze a 2-planted model, where two satisfying assignments are hidden at maximum distance from each other. They experimentally show that in their setting the runtime of local search algorithms is comparable to the runtime on completely random instances. Hu, Luo, and Wang introduce a planted version of community attachement [29] and study it experimentally. Feldman, Perkins, and Vempala [20] study planted k-SAT with different distributions on the signs of clauses.

We consider the greedy algorithm (Algorithm 1) originally introduced by Koutsoupias and Papadimitriou [30] who proved its success on uniform planted

[1] We say that an event $\mathcal{E}$ holds *asymptotically almost surely* (a. a. s.) if, over a sequence of sets, $\Pr(\mathcal{E}) = 1$. In the context of this paper, this means $\Pr(\mathcal{E}) = 1 - o(1)$.

formulas with at least linear constraint density, i.e., the ratio of clauses to variables is $\Omega(n)$. Bulatov and Skvortsov [11] proved a phase transition in the uniform model for this algorithm. In particular, for constraint densities above $\frac{7}{6}\ln n$, Algorithm 1 succeeds with high probability. On the other hand, the algorithm fails w.h.p. on formulas with uniformly positive constraint densities below this threshold. More sophisticated algorithms based on spectral techniques have been shown to be successful down to constant densities with high probability [21] and in expected polynomial time [31] on the uniform planted model.

Algorithm 1: Greedy algorithm [30]

```
α ← an assignment chosen uniformly at random;
while ∃ i ∈ [n] such that changing α_i increases the number of satisfied clauses do
    α[i] ← 1 − α[i];
return α
```

2 Non-uniform Planted k-SAT

In this section we will introduce our model and relevant notation formally. We denote the Boolean variables by $x_1, \ldots, x_n$. A k-clause is a disjunction of k literals $\ell_1 \vee \ldots \vee \ell_k$, where each literal is a variable or its negation. For a literal ℓ_i let $|\ell_i|$ denote the index of its variable. A formula Φ in conjunctive normal form is a conjunction of clauses $C_1 \wedge \ldots \wedge C_m$. We interpret a clause C both as a Boolean formula and as a set of literals. We say that Φ is satisfiable if there exists an assignment of its variables such that the formula evaluates to 1.

Definition 1 (Non-Uniform Random k-SAT). *Let $(\vec{p}_n)_{n\in\mathbb{N}}$ be a set of probability distributions where $\vec{p}_n = (p_1, p_2, \ldots, p_n)$ is a probability distribution over n Boolean variables with $\Pr(X = x_i) = p_i$. The random model $\mathcal{D}(n, m, \vec{p}_n, k)$ can be described as follows.*

1. ***for*** $j \leftarrow 1$ ***to*** m*:*
   (a) *Sample k variables from the distribution $\vec{p}_n$ without repetition.*
   (b) *Choose one of the 2^k negation patterns uniformly at random.*

Definition 2 (Non-Uniform Planted k-SAT). *Let $(\vec{p}_n)_{n\in\mathbb{N}}$ be a set of probability distributions where $\vec{p}_n = (p_1, p_2, \ldots, p_n)$ is a probability distribution over n Boolean variables with $\Pr(X = x_i) = p_i$. The random planted model $\mathcal{F}(n, m, \vec{p}_n, k)$ can be described as follows.*

1. *Select a planted assignment $\alpha^\star \in \{0,1\}^n$ uniformly at random*
2. ***for*** $j \leftarrow 1$ ***to*** m*:*
   (a) *Sample k variables from the distribution $\vec{p}_n$ without repetition.*
   (b) *Choose one of the $2^k - 1$ negation patterns that force the resulting j-th clause to evaluate to true under $\alpha^\star$ uniformly at random.*

We will show in Sect. 3 that the greedy algorithm is successful on non-uniform planted k-SAT if the clause-variable ratio is high enough and if the variable probability distribution is well-behaved in some sense. Moreover, we will relate the two models in Sect. 4, which allows us to conclude that the greedy algorithm also succeeds on satisfiable instances of non-uniform random k-SAT. Our results in Sect. 4 hold for more general versions of those models, which are defined as follows.

Definition 3 (Random k-SAT with Independent Signs). *Let $\mathcal{N}$ denote* any *random k-SAT model where m clauses are drawn and the signs of variables for each clause are drawn independently and uniformly at random among the 2^k possibilities. Let $F \sim \mathcal{N}$ denote a random formula F drawn in the model N. This means the probability to draw a certain k-CNF f is*

$$\Pr_{F\sim\mathcal{N}}(F = f) = p_f \cdot 2^{-k\cdot m},$$

where p_f denotes the probability to draw the sets of variables that the clauses of f consist of. We call such a model a random k-SAT model with independent signs.

Definition 4 (Corresponding Planted Model). *Let $\mathcal{N}$ be a random k SAT model with independent signs. Now let $\mathcal{P}$ be the following planted model: First draw a planted assignment with probability 2^{-n}, then we draw m clauses in the same way as in $\mathcal{N}$, and draw the signs of variables for each clause independently and uniformly at random among the $2^k - 1$ possibilities that make the planted assignment satisfy the clause. If $X(f)$ denotes the number of satisfying assignments of a k-CNF f, then the probability to draw f is*

$$\Pr_{F\sim\mathcal{P}}(F = f) = p_f \cdot \frac{X(f)}{2^n} \cdot \left(\frac{1}{2^k - 1}\right)^m.$$

We call $\mathcal{P}$ the corresponding planted model *of $\mathcal{N}$.*

Note that the definition of a random k-SAT model with independent signs is very general. It encompasses random k-SAT models where formulas with m clauses over n variables are drawn according to *any* distribution, as long as the sign of each literal is drawn independently at random with probability 1/2. This includes the community attachment model by Giráldez-Cru and Levy [27] and the approach with given variable degrees of Omelchenko and Bulatov [33] and Levy [12]. Furthermore, it is easy to see that non-uniform random k-SAT is a random k-SAT model with independent signs and that non-uniform planted k-SAT is its corresponding planted model.

Throughout the paper, we will assume that $k \geq 3$ is a constant. Note that according to our models clauses can be drawn repeatedly. Furthermore, to simplify the proofs, we assume the variables are sampled with replacement. However, we remark that for k constant and p_n bounded away from 1 by a constant, this changes the clause probabilities by at most a constant factor (see, e.g., [25]).

In this setting, the probability to draw a legal clause $C = (\ell_1 \vee \ldots \vee \ell_k)$ is $\frac{k!}{2^k-1} \cdot \prod_{j=1}^{k} p_{|\ell_j|}$.

We denote $[n] := [1, n] \cap \mathbb{N}$. For a discrete probability distribution $\vec{p} = (p_1, \ldots, p_n)$ we assume $p_1 \leq p_2 \leq \cdots \leq p_n$. For a particular $\mathcal{F}(n, m, \vec{p}_n, k)$, we define the parameter $\gamma(\varepsilon) := \Pr(i \leq (1/2 - \varepsilon) \cdot n - (k-1))$. Here, i is a random variable with $\Pr(i = j) = p_j$ for $j \in [n]$. We will denote the Hamming distance between two assignments $\alpha, \beta \in \{0,1\}^n$ by $d(\alpha, \beta)$ and simply refer to it as the "distance".

3 The Greedy Algorithm on Non-uniform Planted k-SAT

In this section, we will show that for sufficiently high constraint densities Algorithm 1 asymptotically almost surely finds a satisfying assignment of non-uniform planted k-SAT if a condition on the probability distribution of the model is fulfilled. The condition that has to be satisfied is that there are constants $\varepsilon \in (0, 1/2)$ and $\varepsilon' \in (0, \varepsilon)$ such that

$$\Pr\left(i \leq (1/2 - \varepsilon') \cdot n - (k-1)\right) > c + \Pr\left(i > (1/2 + \varepsilon) \cdot n\right) \tag{1}$$

for some $c = \Omega\left(\left(n \cdot p_1 \cdot \gamma(\varepsilon)^{3(k-1)} / \ln n\right)^{1/2k}\right)$. If a probability distribution $\vec{p}$ satisfies this condition, we call it "well-behaved". Formally, we show the following.

Theorem 11. *For a formula F drawn from $\mathcal{F}(m, n, \vec{p}, k)$ with a well-behaved probability distribution $\vec{p}$ with parameters ε and ε', and $m \geq \frac{C \ln n}{\gamma(\varepsilon)^{3(k-1)} p_1}$, where $k \geq 3$ is a constant and $C > 0$ is some sufficiently large constant, Algorithm 1 succeeds with high probability.*

Note that the choice of ε in the well-behavedness condition influences the value of $\gamma(\varepsilon)$ in the number of clauses necessary for the algorithm to succeed. It generally holds that the more uniform the probability distribution is, the smaller we can choose ε and a smaller ε results in a smaller lower bound on the number of clauses.

We call an assignment $\alpha \in \{0,1\}^n$ *good* if it satisfies all clauses or if there is an assignment β with $|\{i : \alpha_i \neq \beta_i\}| = 1$ and β satisfies strictly more clauses than α. We will show that a. a. s. all assignments that Algorithm 1 finds are good. Thus, the assignment it returns must be satisfying. To this end we consider assignments at distances $(1/2+\varepsilon') \cdot n$ and $(1/2+\varepsilon) \cdot n$ from the planted assignment $\alpha^\star$. Here, ε' and ε are the parameters of the well-behavedness condition with $0 < \varepsilon' < \varepsilon < 1/2$. There are five ingredients to the proof: (1) two technical lemmas, Lemmas 6 and 7, (2) Lemma 8, which states that all assignments within distance $(1/2+\varepsilon) \cdot n$ of $\alpha^\star$ are good, (3) Lemma 9, which states that the random starting assignment is at distance at most $(1/2+\varepsilon') \cdot n$ from $\alpha^\star$, (4) Lemma 10, which states that any assignment at distance $(1/2+\varepsilon') \cdot n$ from $\alpha^\star$ satisfies at least as many clauses as any assignment at distance $(1/2+\varepsilon) \cdot n$ from $\alpha^\star$, and (5) Theorem 11, which puts these ingredients together.

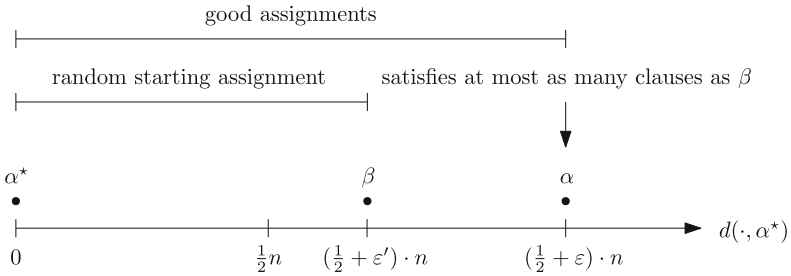

Fig. 1. Sketch of the assignment space with the planted assignment $\alpha^\star$ and the properties we show. (2) Lemma 8: All assignments within distance $(1/2+\varepsilon)\cdot n$ of $\alpha^\star$ are good, (3) Lemma 9: The random starting assignment is at distance at most $(1/2+\varepsilon')\cdot n$ from $\alpha^\star$, (4) Lemma 10: any assignment at distance $(1/2+\varepsilon')\cdot n$ from $\alpha^\star$ satisfies at least as many clauses as any assignment at distance $(1/2+\varepsilon)\cdot n$ from $\alpha^\star$

The argument now works as follows. Since the local search algorithm always picks an assignment that strictly increases the number of satisfied clauses, (4) implies that from an assignment at distance $(1/2+\varepsilon')\cdot n$, it will never reach one at distance $(1/2+\varepsilon)\cdot n$. Due to (3) the algorithm starts with an assignment within distance $(1/2+\varepsilon')\cdot n$ of $\alpha^\star$. Thus, all assignments found by the algorithm must remain within distance $(1/2+\varepsilon)\cdot n$ of $\alpha^\star$. Since all assignments within that distance to $\alpha^\star$ are good due to (2), all assignments found by the algorithm are good and the final assignment must be satisfying. Figure 1 visualizes the idea of the proof. Furthermore, Corollary 12 shows that some natural probability distributions are well-behaved for certain constants $\varepsilon, \varepsilon'$, which result in a constant $\gamma(\varepsilon)$. For instances of non-uniform planted k-SAT with these input distributions Algorithm 1 already works for logarithmic densities.

The efficiency of the greedy algorithm depends on the probability of sampling clauses over certain subsets of variables. We capture the probability of sampling a certain subset of variables in the following definition.

Definition 5. *Given any index set $I \subseteq [n]$, let $\mathcal{P}_l(I) = \{J \subseteq I : |J| = l\}$ denote the cardinality-l elements of the power set of I and define $Q_l(I) := \sum_{J\in\mathcal{P}_l(I)} \prod_{j\in J} p_j$ to be the probability of selecting l elements of I over $\vec{p}$.*

$Q_l(I)$ is the probability of choosing l variables with indices only from I. Note that $Q_l(I) \geq Q_{l'}(I)$ for $l \leq l'$. We want to lower-bound the probability $Q_l(I)$ for $|I| \geq (1/2-\varepsilon)\cdot n$. In the uniform planted model a lower bound would be roughly $(1/2-\varepsilon)^l$, where $0 < \varepsilon < 1/2$ is a constant. However, in our setting, where variable probabilities are non-uniform, $Q_l(I)$ depends on the total probability mass of the $(1/2-\varepsilon)\cdot n-(l-1)$ least probable variables. We underestimate and capture this probability mass in the parameter $\gamma(\varepsilon) = \Pr(i \leq (1/2-\varepsilon)\cdot n-(k-1))$. The following lemma now provides us with a lower bound on $Q_l(I)$ depending on $\gamma(\varepsilon)$.

Lemma 6. *If $\gamma(\varepsilon) > 0$ for some constant $0 < \varepsilon < 1/2$, then for any index set I with $|I| \geq (1/2 - \varepsilon) \cdot n$ and any natural number $l \leq k$, we have $Q_l(I) \geq \frac{\gamma(\varepsilon)^l}{l!}$.*

Proof. We can express $Q_l(I)$ as the following nested sum

$$\begin{aligned} Q_l(I) &= \frac{1}{l!} \sum_{i_1 \in I} \sum_{i_2 \in I \setminus \{i_1\}} \cdots \sum_{i_l \in I \setminus \{i_1, \ldots, i_{l-1}\}} \prod_{j=1}^{l} p_{i_j} \\ &= \frac{1}{l!} \sum_{i_1 \in I} p_{i_1} \sum_{i_2 \in I \setminus \{i_1\}} p_{i_2} \cdots \sum_{i_l \in I \setminus \{i_1, \ldots, i_{l-1}\}} p_{i_l}. \end{aligned}$$

This sum essentially captures the choices of elements we have for each term in $Q_l(I)$, where i_j is the j-th chosen element. Since we only forbid repetitions of elements, the j-th element can be anything from $I \setminus \{i_1, i_2, \ldots, i_{j-1}\}$. Since $|I| \geq (1/2 - \varepsilon) \cdot n$, we can always choose from at least $(1/2 - \varepsilon) \cdot n - (l-1)$ many elements. It holds that

$$Q_l(I) \geq \frac{1}{l!} \left(\sum_{i=1}^{(1/2-\varepsilon)\cdot n - (l-1)} p_i \right)^l = \frac{1}{l!} \Pr(i \leq (1/2 - \varepsilon) \cdot n - (l-1))^l \geq \frac{\gamma(\varepsilon)^l}{l!}.$$

as we assume the p_i to be in ascending order. □

The following technical lemma bounds the probability of making a random clause satisfied or unsatisfied by decreasing the Hamming distance to the planted solution. These bounds especially hold if the distance is decreased by only one, i. e. we flip the assignment of a single variable. The statements of this lemma will be used in order to show that assignments close to the planted solution are good.

Lemma 7. *Fix an assignment $\alpha \in \{0,1\}^n$ at Hamming distance $d(\alpha, \alpha^\star) < (1/2 + \varepsilon) \cdot n$ from the planted solution. For any assignment β with $\{i : \alpha_i = \alpha_i^\star\} \subseteq \{i : \beta_i = \alpha_i^\star\}$, denote $\pi_{\alpha\beta}$ as the probability over $\mathcal{F}(n, m, \vec{p}, k)$ that a clause is false under α and true under β. Analogously, we let $\pi_{\beta\alpha}$ denote the probability that a clause is false under β and true under α. With $I = \{i : \alpha_i \neq \beta_i \wedge \beta_i = \alpha_i^\star\}$ it holds that*

1. *$\pi_{\beta\alpha} \leq (1 - \frac{\gamma(\varepsilon)^{k-1}}{(k-1)!}) \cdot \pi_{\alpha\beta}$, and*
2. *$\pi_{\alpha\beta} \geq \frac{k \cdot \gamma(\varepsilon)^{k-1} \cdot |I| \cdot p_1}{2^k - 1}$.*

Proof. In addition to I, we will denote the set $J := \{i : \alpha_i = \alpha_i^\star\}$. Note that $|I| = d(\alpha, \beta) \leq d(\alpha, \alpha^\star)$ and that $d(\alpha, \alpha^\star) = n - |J|$. A clause changes from false to true between α and β if it (1) contains any variable indexed in I, and (2) the literals in the clause are set such that it evaluates to false under x. Note that the first condition implies $\alpha \neq \beta$ and $\alpha \neq \alpha^\star$. This is necessary in order for a clause

to evaluate to false under α and to evaluate differently under β. The probability for these events to occur is

$$\pi_{\alpha\beta} = \frac{k!}{2^k - 1} \sum_{\ell=1}^{k} Q_\ell(I) \cdot Q_{k-\ell}([n] \setminus I)$$

We have the same for $\pi_{\beta\alpha}$. Again, the clause must contain a variable from I for α and β to be different and this time the literals must evaluate to false under β. Additionally, we must take care that clauses that are false under $\alpha^\star$ are not allowed. In particular, if a clause contains only variables from $I \cup J$, i. e. only variables where β and $\alpha^\star$ do not differ, then the clause cannot evaluate to false under β. Thus, we must exclude such clauses from the probability mass. In particular,

$$\begin{aligned}
\pi_{\beta\alpha} &= \frac{k!}{2^k - 1} \sum_{\ell=1}^{k} Q_\ell(I) \cdot (Q_{k-\ell}([n] \setminus I) - Q_{k-\ell}(J)) \\
&\leq \frac{k!}{2^k - 1} \sum_{\ell=1}^{k} Q_\ell(I) Q_{k-\ell}([n] \setminus I) - \frac{k!}{2^k - 1} \sum_{\ell=1}^{k} Q_\ell(I) Q_{k-\ell}([n] \setminus I) Q_{k-\ell}(J) \\
&\leq \left(1 - \frac{\gamma^{k-1}}{(k-1)!}\right) \pi_{\alpha\beta}.
\end{aligned}$$

The final inequality comes from Lemma 6 and the fact that $|J| = n - d(\alpha, \alpha^\star) \geq (1/2 - \varepsilon) \cdot n$, which allows us to bound $Q_{k-\ell}(J) \geq Q_{k-1}(J) \geq \frac{\gamma(\varepsilon)^{k-1}}{(k-1)!}$.

The second statement holds since

$$\begin{aligned}
\pi_{\alpha\beta} &\geq \frac{k!}{2^k - 1} Q_1(I) \cdot Q_{k-1}([n] \setminus I) = \frac{k!}{2^k - 1} \sum_{i \in I} p_i \cdot Q_{k-1}([n] \setminus I) \\
&\geq \frac{k! \cdot |I| \cdot p_1}{2^k - 1} \cdot Q_{k-1}([n] \setminus I) \geq \frac{k \cdot \gamma(\varepsilon)^{k-1} \cdot |I| \cdot p_1}{2^k - 1}.
\end{aligned}$$

The final inequality comes from Lemma 6 and the fact that $|[n] \setminus I| = n - d(\alpha, \beta) \geq (1/2 - \varepsilon) \cdot n$. □

We will now show that w. h. p. assignments close to the planted assignment $\alpha^\star$ are good. This is the second ingredient of our argument. Remember that we call an assignment $\alpha \in \{0,1\}^n$ good if it satisfies all clauses or if there is an assignment β at distance one which satisfies strictly more clauses.

Lemma 8. *Let F be a formula drawn from $\mathcal{F}(m, n, \vec{p}, k)$, let $\varepsilon \in (0, 1/2)$ be a constant, and let $m \geq \frac{C \ln n}{\gamma(\varepsilon)^{3(k-1)} p_1}$, where $k \geq 3$ is a constant and $C > 0$ is some sufficiently large constant. Then all assignments α within distance $(1/2 + \varepsilon) \cdot n$ of the planted assignment $\alpha^\star$ are good with high probability.*

Proof. Fix an assignment α with $d(\alpha, \alpha^\star) < (1/2 + \varepsilon) \cdot n$. Denote the random variable X_{ij} that indicates that the j-th clause is false under α, but becomes

true by flipping the i-th variable. Similarly, denote as Y_{ij} the random variable that indicates that the j-th clause is true under α but becomes false by flipping the i-th variable. Define $X = \sum_{i:\alpha_i \neq \alpha_i^\star} \sum_{j=1}^{m} X_{ij}$ and $Y = \sum_{i:\alpha_i \neq \alpha_i^\star} \sum_{j=1}^{m} Y_{ij}$. By Lemma 7, $\mathrm{E}[X_{ij}] = \pi_{\alpha\beta}$ and $\mathrm{E}[Y_{ij}] = \pi_{\beta\alpha}$, where α and β differ only on $I = \{i\}$. Thus, $\mathrm{E}[Y] \leq (1 - \frac{\gamma(\varepsilon)^{k-1}}{(k-1)!})\mathrm{E}[X]$.

We want to use Chernoff bounds to show that the values of X and Y are concentrated around their expected values. First, we argue why Chernoff bounds can be applied. X and Y only consider assignments β that differ from α in one variable and are closer to $\alpha^\star$. Let $X_j = \sum_{i:\ \alpha_i \neq \alpha_i^\star} X_{ij}$ and $Y_j = \sum_{i:\ \alpha_i \neq \alpha_i^\star} Y_{ij}$. X_j denotes the number of those assignments, which make a clause true that is false under α, while Y_j denotes the number of those assignments that make a clause false that is true under α. It holds that $Y_j \leq 1$. If a clause is false under one assignment β, it must be true under all assignments that differ on that clause's variables. We know that the clause is true under α and since all other assignments $\beta' \neq \beta$ we consider differ from α in exactly one variable, as soon as they differ from α on one of the clause's variables, they must also differ from β on the clause's variables. Thus, the clause must be satisfiable on all assignments $\beta \neq \beta'$ we consider. $Y_j \leq 1$ implies that we can use a Chernoff bound on $Y = \sum_{j=1}^{m} Y_j$, since the Y_j are independent random variables with values in $[0, 1]$. Similarly, $X_j \leq k$, because if a clause is false under α, then all assignments that differ on that clause's variables will make the clause true. Thus, this holds for all assignments β that differ on one of the clause's variables. However, since we only consider those assignments β that differ from α by at most one variable, there are at most k such assignments, one for each variable of the k-clause. $X_j \leq k$ implies that we can use a Chernoff bound after resizing the variables X_j with a factor of $1/k$. This yields random variables whose values are independently distributed in $[0, 1]$. However, it means that the expected value in the exponent also has to be multiplied with $1/k$.

Applying the Chernoff bounds as stated, for any $\delta \in (0, 1)$, we have $\Pr(X \leq (1 - \delta)\mathrm{E}[X]) \leq e^{-\delta^2 \mathrm{E}[X]/(2 \cdot k)}$. For Y we choose δ' such that $(1 + \delta')\mathrm{E}[Y] = (1 + \delta)(1 - \frac{\gamma(\varepsilon)^{k-1}}{(k-1)!})\mathrm{E}[X]$. Then, $\delta' \geq \delta$ and $\delta' \cdot \mathrm{E}[Y] \geq \delta \cdot (1 - \frac{\gamma(\varepsilon)^{k-1}}{(k-1)!})\mathrm{E}[X]$. We can now apply a Chernoff bound to get

$$\Pr\left(Y \geq (1 + \delta')\mathrm{E}[Y]\right) \leq e^{-\delta'^2 \mathrm{E}[Y]/(2+\delta')} \leq e^{-\delta^2 (1 - \frac{\gamma(\varepsilon)^{k-1}}{(k-1)!})\mathrm{E}[X]/(2+\delta)}.$$

Taking a union bound, the probability of event $\{X \leq (1 - \delta)\mathrm{E}[X]\} \cup \{Y \geq (1+\delta)\left(1 - \frac{\gamma(\varepsilon)^{k-1}}{(k-1)!}\right)\mathrm{E}[X]\}$ is at most $\exp(-\delta^2 \left(1 - \frac{\gamma(\varepsilon)^{k-1}}{(k-1)!}\right)\mathrm{E}[X]/(k \cdot ((2+\delta)) + \ln 2)$. Setting $\delta = \kappa/(2 - \kappa)$ with $\kappa = \frac{\gamma(\varepsilon)^{k-1}}{(k-1)!}$, the event $\{X > Y\}$ occurs with probability at least

$$1 - \exp\left(-\frac{(1 - \kappa) \cdot \kappa^2}{k \cdot (2 - \kappa)(4 - \kappa)}\mathrm{E}[X] + \ln 2\right). \tag{2}$$

Remember that X only considers assignments which differ from α in one variable $\alpha_i \neq \alpha_i^\star$. Hence, $|I| = 1$ for α and any such assignment β. Thus, according to Lemma 7,

$$\mathrm{E}[X] \geq m \cdot d(\alpha, \alpha^\star) \cdot \frac{k \cdot \gamma(\varepsilon)^{k-1} \cdot p_1}{2^k - 1} = m \cdot d(\alpha, \alpha^\star) \cdot \frac{k! \cdot \kappa \cdot p_1}{2^k - 1}.$$

Substituting this into Eq. (2) and using $0 \leq \kappa = \frac{\gamma(\varepsilon)^{k-1}}{(k-1)!} \leq \frac{1}{(k-1)!}$ we get $\frac{(1-\kappa)\cdot\kappa^3}{(2-\kappa)(4-\kappa)} \geq (1 - 1/(k-1)!) \cdot \gamma(\varepsilon)^{3k-3}/(2 \cdot (k-1)!)^3 = \Omega(\gamma(\varepsilon)^{3(k-1)})$. Thus, the event $\{X > Y\}$ occurs for assignment α with probability at least

$$1 - \exp\left(-\Omega\left(\gamma(\varepsilon)^{3(k-1)} \cdot p_1 \cdot d(\alpha, \alpha^\star) \cdot m\right)\right).$$

This means the average count of clauses that go from false to true minus the count that go from true to false by flipping assignments in $\{i : \alpha_i \neq \alpha_i^\star\}$ is positive, and we can conclude that there exists at least one such flip that increases the total count of satisfied clauses. Hence, α is *good* with the above probability.

Taking a simple union bound over all $\binom{n}{d} \leq n^d$ assignments α at distance $d = d(\alpha, \alpha^\star)$, all assignments at this distance are good with probability at least

$$1 - n^d \exp\left(-\Omega\left(\gamma(\varepsilon)^{3(k-1)} \cdot p_1 \cdot d \cdot m\right)\right) \geq 1 - \exp(-\Omega(d \log n)) = 1 - n^{-C'd}$$

for some constant C' by choosing $m \geq \frac{C \cdot \ln n}{\gamma(\varepsilon)^{3(k-1)} \cdot p_1}$ with constant C large enough. A subsequent union bound over all such radius-d spheres yields that all assignments within distance $(1/2+\varepsilon)\cdot n$ of the planted solution are good with probability at least $1 - \sum_{d=1}^{\lfloor (1/2+\varepsilon)n \rfloor} n^{-C'd} \geq 1 - 1/\left(n^{C'} - 1\right)$, i. e. with high probability. □

Now we are going to show the third ingredient of our argument, i. e. that the random starting assignment is close to the planted assignment with high probability.

Lemma 9. *For any constant $\varepsilon' \in (0, 1/2)$ the random starting assignment is within distance at most $(1/2 + \varepsilon') \cdot n$ of the planted assignment $\alpha^\star$ with high probability.*

Proof. Since the starting assignment $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ is generated uniformly at random, each α_i differs from $\alpha_i^\star$ with probability $1/2$ independently at random. Let X_i denote the random variable indicating that $\alpha_i \neq \alpha_i^\star$ and let $X = \sum_{i=1}^n X_i$. We can see that $d(\alpha, \alpha^\star) = X$. It holds that $\mathrm{E}[X] = n/2$ and

$$\Pr(d(\alpha, \alpha^\star) > (1/2 + \varepsilon') \cdot n) = \Pr(X > (1 + 2 \cdot \varepsilon') \cdot \mathrm{E}[X]) \leq e^{-\frac{2 \cdot \varepsilon'^2 \cdot n}{2 + 2 \cdot \varepsilon'}}$$

due to a Chernoff bound. □

The last ingredient of our argument is to show that any assignment β at distance $(1/2+\varepsilon')\cdot n$ from $\alpha^\star$ satisfies at least as many clauses as any assignment α at distance $(1/2 + \varepsilon) \cdot n$ from $\alpha^\star$. In order to show this result, we require the variable probability distribution of our random model to be well-behaved. For β

and α well-behavedness essentially states that it is more probable to randomly sample a variable on which $\alpha^\star$ and β agree than it is to sample a variable on which $\alpha^\star$ and α agree. For a uniform probability distribution this is trivially true, since the number of those variables is much larger in β than it is in α due to β's smaller Hamming distance to $\alpha^\star$. However, for a non-uniform probability distribution, the property must be ensured. We will later see in Corollary 12 that uniform, power-law and geometric distributions are well-behaved.

Lemma 10. *Let $k \geq 3$ be a constant and let $\vec{p}$ be a probability distribution that is well-behaved for constants $\varepsilon \in (0, 1/2)$ and $\varepsilon' \in (0, \varepsilon)$. Further, let $m \geq \frac{C \ln n}{\gamma(\varepsilon)^{3(k-1)} p_1}$ for a sufficiently large constant $C > 0$, and let F be a formula drawn from $\mathcal{F}(m, n, \vec{p}, k)$. Then with high probability any assignment α with $d(\alpha, \alpha^\star) = (1/2 + \varepsilon) \cdot n$ satisfies at most as many clauses of F as any assignment β with $d(\beta, \alpha^\star) = (1/2 + \varepsilon') \cdot n$.*

Proof Sketch. The idea of the proof is to lower-bound the difference $\pi_{\alpha\beta} - \pi_{\beta\alpha}$, where $\pi_{\alpha\beta}$ is the probability that a random clause is not satisfied by α and satisfied by β. This difference depends on the probabilites of variables in I, the set of variables on which α and β differ. More precisely, it depends on the difference between the probability of sampling a variable from I for which α and $\alpha^\star$ disagree and the probability of sampling a variable from I for which α and $\alpha^\star$ agree. In the worst case the prior set of variables are those of minimal probabilities, while the latter are those of maximal probabilities according to the probability distribution $\vec{p}$. If we pessimistically assume this, the difference is minimized if I is of maximum size. Then, there are $(1/2 - \varepsilon') \cdot n$ variables in I on which α and $\alpha^\star$ disagree and $(1/2 - \varepsilon) \cdot n$ variables on which the assignments agree. However, the difference of the probabilites to sample those variables is lower bounded by $c = \Omega\left(\left(n \cdot p_1 \cdot \gamma(\varepsilon)^{3(k-1)} / \ln n\right)^{1/2k}\right)$ by the well-behavedness of $\vec{p}$ (Eq. 1).

By using a Chernoff bound, we can now show that the probability that α satisfies at least as many clauses as β is upper bounded by $\sim \exp(-m \cdot c^{2k}) \sim 2^{-\Omega(n)}$. Via a union bound we get that the probability is still exponentially small in n for all pairs of assignments α and β if C is sufficiently large. □

We can now put the ingredients of our argument together to get our main theorem.

Theorem 11. *For a formula F drawn from $\mathcal{F}(m, n, \vec{p}, k)$ with a well-behaved probability distribution $\vec{p}$ with parameters ε and ε', and $m \geq \frac{C \ln n}{\gamma(\varepsilon)^{3(k-1)} p_1}$, where $k \geq 3$ is a constant and $C > 0$ is some sufficiently large constant, Algorithm 1 succeeds with high probability.*

Proof. All statements in the proof hold with high probability. Lemma 9 tells us that the random starting assignment is within distance $(1/2+\varepsilon') \cdot n$ of the planted assignment $\alpha^\star$. The local search algorithm now considers assignments within Hamming distance one of the currently best assignment found. Furthermore,

the algorithm only accepts a new best assignment if it satisfies strictly more clauses than the previous best assignment. Thus, to reach an assignment α at distance $(1/2+\varepsilon)\cdot n$ from $\alpha^\star$, it first has to accept an assignment β at distance $(1/2+\varepsilon')\cdot n$ from $\alpha^\star$ and α has to satisfy strictly more clauses than β. However, Lemma 10 tells us that this is not possible. Therefore, any assignment found by the algorithm has to be within distance $(1/2+\varepsilon)\cdot n$ of $\alpha^\star$. Lemma 8 states that all those assignments are good. Thus, all assignments found by the algorithm are good and the final assignment must be satisfying. □

The $\gamma(\varepsilon)$ term in our proofs is a penalty incurred from having a potentially pathologically "light" tail in the variable distribution. If $\gamma(\varepsilon) = o(1)$, this means that most of the probability mass is concentrated around the $(1/2+\varepsilon)\cdot n$ most frequent variables, and the tail vanishes very quickly. In some sense, if the tail is at least as heavy as the uniform distribution, then $\gamma = \Theta(1)$. This is the case for most proposed classes of non-uniform variable distributions, as we formalize in Corollary 12.

The well-behavedness of the variable distribution intuitively states something similar. It also requires that not too much probability mass is concentrated around the most frequent variables. Note that ε denotes the same value in both requirements. We can see that increasing ε and decreasing ε' makes it easier to satisfy this prerequisite. However, increasing ε decreases $\gamma(\varepsilon)$ and thus increases the lower bound on the clause-variable ratio for which our main theorem holds.

Theorem 11 implies that the greedy algorithm already works at some logarithmic density if the variables of the planted model follow three well-known probability distributions: uniform, power-law, or geometric. We show this in the following corollary.

Corollary 12. *The greedy algorithm is successful over a $1-o(1)$ fraction of planted*

1. *uniform random k-SAT formulas,*
2. *power-law random k-SAT formulas with power-law exponent $\beta > 2$,*
3. *geometric random k-SAT formulas*[2] *with a base $b > 1$,*

with $\frac{m}{n} \geq C \ln n$, for constant $k \geq 3$ and a sufficiently large constant C.

Proof. The statement follows by application of Theorem 11, so it suffices to verify the minimum variable probability p_1, the γ term, and the well-behavedness of the distribution for each of the stated models.

1. Uniform: In the uniform k-SAT distribution, $p_1 = p_i = 1/n$ for all $i \in [n]$. Therefore, $\gamma(\varepsilon) = (1/2-\varepsilon) - (k-1)/n = \Theta(1)$ and

$$\begin{aligned}\Pr\left(i \leq (1/2-\varepsilon')\cdot n - (k-1)\right) &= (1/2-\varepsilon') - (k-1)/n \\ &\geq c + (1/2-\varepsilon) = c + \Pr\left(i > (1/2+\varepsilon)\cdot n\right)\end{aligned}$$

for $c = \varepsilon - \varepsilon' - (k-1)/n$. Thus, Algorithm 1 succeeds w. h. p. for clause-variable ratios $\frac{m}{n} \geq C \cdot \frac{\ln n}{n\cdot\gamma(\varepsilon)^{3(k-1)}\cdot p_1} = C\cdot \ln n$ for some sufficiently large constant $C > 0$.

[2] We refer to the geometric degree-distribution model introduced by Ansótegui et al. [6].

2. Power law: For the power-law distribution, $p_1 = (1/\sum_{i=1}^{n} \left(\frac{n}{i}\right)^{\frac{1}{\beta-1}}) = \Omega(1/n)$ and $p_n = \Theta(n^{-(\beta-2)/(\beta-1)})$. Thus, $\gamma(\varepsilon) = \Pr(i \leq (1/2-\varepsilon)\cdot n-(k-1)) = \Theta(1)$, since $\Pr(i \leq (1/2-\varepsilon)\cdot n-(k-1)) \geq (1/2-\varepsilon)\cdot n\cdot p_1-(k-1)\cdot p_n = \Omega(1)$. In order to validate the well-behavedness of the distribution, we can estimate $\Pr(i > (1/2+\varepsilon)\cdot n) \leq (1/2-\varepsilon)^{(\beta-2)/(\beta-1)}$ and, equivalently

$$\begin{aligned}\Pr(i \leq (1/2-\varepsilon')\cdot n-(k-1)) &= 1-\Pr(i > (1/2-\varepsilon')\cdot n-(k-1))\\ &\geq 1-(1/2+\varepsilon'+(k-1)/n)^{(\beta-2)/(\beta-1)}.\end{aligned}$$

Thus, for any $\varepsilon' \in (0,1/2)$ we can choose $\varepsilon > \max(\varepsilon', \varepsilon_0)$, where ε_0 is the solution of

$$1-(1/2+\varepsilon'+(k-1)/n)^{(\beta-2)/(\beta-1)} = (1/2-\varepsilon)^{(\beta-2)/(\beta-1)}.$$

Note that this lower bound on ε is always in $(0,1/2)$ and thus satisfies our requirements. As in the uniform case, this results in a lower bound of $\frac{m}{n} \geq C\cdot\ln n$ for some sufficiently large constant $C > 0$ in order for Algorithm 1 to succeed with high probability.

3. Geometric: In geometric random k-SAT, $p_i = \frac{1-b^{-1/n}}{b-1}\cdot b^{i/n}$. It now holds that $\sum_{i=1}^{(1/2-\varepsilon)n} p_i = \frac{b^{1/2-\varepsilon}-1}{b-1}$ and thus $\gamma(\varepsilon) = \frac{b^{1/2-\varepsilon-(k-1)/n}-1}{b-1} = \Theta(1)$. Furthermore,

$$p_1 = \frac{b^{1/n}-1}{b-1} = \frac{e^{\ln(b)/n}-1}{b-1} \geq \frac{1+\ln(b)/n-1}{b-1} = \frac{\ln(b)}{b-1}\cdot\frac{1}{n}.$$

For the requirement from Eq. 1, we get

$$\Pr(i > (1/2+\varepsilon)\cdot n) = 1-\Pr(i \leq (1/2+\varepsilon)\cdot n) = 1-\frac{b^{1/2+\varepsilon}-1}{b-1}.$$

This means, we need to ensure $\frac{b^{1/2-\varepsilon'-(k-1)/n}-1}{b-1} > 1-\frac{b^{1/2+\varepsilon}-1}{b-1}$ or, equivalently, $b^{1/2+\varepsilon} > b+1-b^{1/2-\varepsilon'-(k-1)/n}$. Note that $b^{1/2-\varepsilon'-(k-1)/n} > 1$. Thus, the right-hand side is a constant smaller than b. If we make $\varepsilon \in (0,1/2)$ sufficiently large, we can make the left-hand side by a constant bigger than the right-hand side. This is sufficient for the requirement from inequality 1. Again, we get that the greedy algorithm succeeds w. h. p. for $\frac{m}{n} \geq C\cdot\ln n$ and $C > 0$ sufficiently large. □

4 Relationship Between Planted and Filtered Instances

One interesting question is if the behavior of the greedy algorithm is an artifact of the instances being planted or if the same behavior emerges for satisfiable instances of the corresponding non-planted model. Thus, we now look at random k-SAT models with independent signs and their corresponding planted models. We show the following theorem, which is a generalization of a result by Doerr, Neumann, and Sutton [18].

Theorem 13. *Let $\mathcal{P} = \mathcal{F}(n, m, \vec{p}_n, k)$ be a non-uniform planted k-SAT model and let $\mathcal{N}$ be a non-uniform random k-SAT model on the same input parameters. Then for $m \geq \frac{(1+\varepsilon)\cdot(2^k-1)}{p_1} \cdot \ln n$ with any constant $\varepsilon > 0$ and for any event $\mathcal{E}$ it holds that $\Pr_{F\sim\mathcal{N}}(\mathcal{E} \mid X(F) \geq 1) = \Pr_{F\sim\mathcal{P}}(\mathcal{E}) \pm o(1)$.*

Proof Sketch. The proof follows the same lines as the one in [18]. We first show that for a random k-SAT model with independent signs and its planted equivalent the conditional probability to sample a certain formula is the same in both models if we condition on there being exactly one satisfying assignment. Then, we show that the probability to have exactly one satisfying assignment in the filtered model (conditioned on formulas being satisfiable) is at least as high as in the planted model. These two statements already imply a total variation distance that tends to zero as soon as the probability to have a unique satisfying assignment tends to one in the planted model. The last step of the proof consists of finding a number of clauses m for which formulas generated with non-uniform planted k-SAT a. a. s. only have one satisfying assignment. A first oder bound shows that this is case if $m \geq \frac{(1+\varepsilon)\cdot(2^k-1)}{p_1} \cdot \ln n$ for any constant $\varepsilon > 0$. □

Theorem 13 asserts that Theorem 11 also holds for the filtered non-uniform random k-SAT model. That means, for satisfiable formulas drawn from the non-uniform random k-SAT model the greedy algorithm also succeeds with probability $1 - o(1)$.

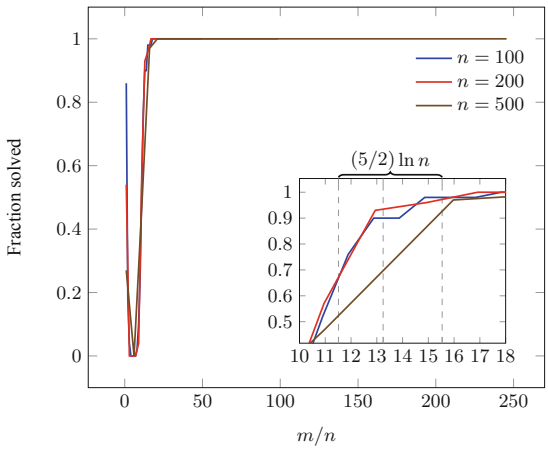

Fig. 2. Fraction of formulas solved by Algorithm 1 on the planted uniform 3-SAT distribution as a function of constraint density m/n for various n.

5 Experiments

We performed a number of experiments for the example distributions we consider in Corollary 12 to argue that the logarithmic lower bound in constraint density

for Algorithm 1 is likely to be tight asymptotically, and that the leading constants are small. For the uniform planted 3-SAT model, we sampled formulas at $n \in \{100, 200, 500\}$ with densities $1 \leq m/n \leq n^2/2$. For each n and m, we sampled 100 formulas and determined whether they could be solved by the greedy algorithm. We report the results as the fraction of formulas solved depending on the constraint density in Fig. 2.

As expected, above constraint densities of roughly $\Theta(\log n)$, the proportion of formulas solved by Algorithm 1 quickly goes to one. We see success rates of 70–90% already at $(5/2)\ln n$ for each n, but a more detailed analysis would be needed to get an accurate estimate for the true leading constant.

Non-uniform distributions typically have more parameters, and we are interested in the influence of these parameters on the success of the greedy algorithm. In particular, other than the minimum variable probability p_1, and the γ term for tail lightness, no other distribution parameter appears in our bound. To quantify the effect of constraint density and distribution parameter on geometric random 3-SAT and power-law random 3-SAT, we sampled 100 formulas for each value of the parameters across a range. We measured the proportion of these formulas that were solved by the greedy algorithm, and display the results in heat maps in Fig. 3. On the left, the fraction solved is shown as a function of density and base parameter b for the geometric distribution. On the right, the fraction solved is shown as a function of density and power law exponent β for power-law formulas. As reflected in our theoretical bounds, for the most part there is little influence of the distribution parameters b and β on the constraint density above which Algorithm 1 is successful. In the power law model, there appears to be a regime of the power law exponent β near 2 that seems to be influencing the lower bound. This might be due to hidden constant factors which depend on the power law exponent β. However, it is not clear how or whether this effect scales with n, and this is an avenue for future work. Of course, we cannot claim that our lower bound on the constraint density is tight for all possible well-behaved distributions. As we stated before, our lower bound only considers the smallest

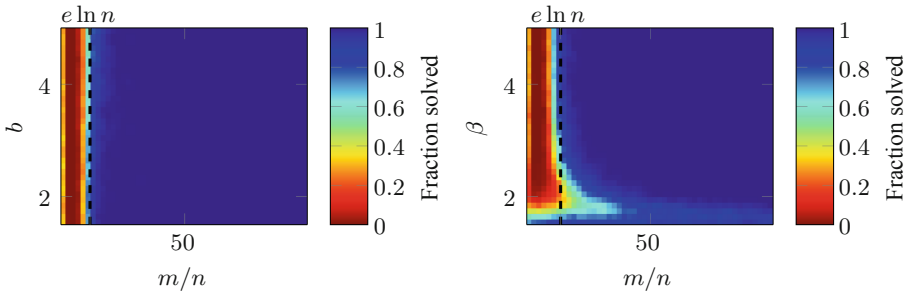

Fig. 3. Fraction of formulas solved by Algorithm 1 on geometric 3-SAT distribution as a function of constraint density m/n and base parameter b (left) and power law 3-SAT distribution as a function of constraint density m/n and power law exponent β (right) for $n = 200$ variables.

variable probability p_1, and the γ term, whereas the actual bound might have a more intricate relation to other distribution parameters.

6 Conclusions

Non-uniform k-SAT models have gained increased attention in recent years. With this paper, we contribute to the theoretical understanding of SAT problems with such non-uniform distributions by studying a greedy local search algorithm. We have shown that this algorithm is highly effective on planted SAT formulas drawn from k-SAT models realized by choosing variables from an arbitrary variable distribution, provided that the clauses are generated independently and that the variable distribution is not too skewed. Models with these properties include geometric and power-law random k-SAT [6].

Our experimental results reveal that for geometric and power-law distributions the exact parameters of the variable degree distribution have little influence on the success of the local search algorithm, at least in the planted setting. Moreover, our rigorous lower bounds on the clause-variable ratio necessary for the algorithm to succeed with high probability are asymptotically the same as for uniform planted k-SAT. This is somewhat surprising, as for state-of-the-art SAT solvers it is typically assumed that the non-uniform distributions we consider make instances easier to solve [5].

We also show that there is a correspondence between non-uniform planted k-SAT distributions and their *filtered* analogues, i.e., the non-planted distribution conditioned on satisfiability. We show that for large enough clause-variable ratios the total variation distance for events in filtered and their corresponding planted models vanishes in the limit. This result actually holds for all random k-SAT models, where the signs of literals are chosen independently at random without bias. It allows us to transfer our results for the greedy local search algorithm to filtered non-uniform models.

References

1. Achlioptas, D., Jia, H., Moore, C.: Hiding satisfying assignments: two are better than one. J. Art. Int. Res. 24 (2005). https://doi.org/10.1613/jair.1681
2. Achlioptas, D., Kirousis, L.M., Kranakis, E., Krizanc, D.: Rigorous results for random (2+p)-sat. Theor. Comput. Sci. **265**(1–2), 109–129 (2001)
3. Ansótegui, C., Bonet, M.L., Giráldez-Cru, J., Levy, J.: The fractal dimension of SAT formulas. In: Demri, S., Kapur, D., Weidenbach, C. (eds.) IJCAR 2014. LNCS (LNAI), vol. 8562, pp. 107–121. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-08587-6_8
4. Ansótegui, C., Bonet, M.L., Giráldez-Cru, J., Levy, J.: On the classification of industrial SAT families. In: Armengol, E., Boixader, D., Grimaldo, F. (eds.) 18th International Conference Catalan Association for Artificial Intelligence. Frontiers in Artificial Intelligence and Applications, vol. 277, pp. 163–172. IOS Press (2015). https://doi.org/10.3233/978-1-61499-578-4-163

5. Ansótegui, C., Bonet, M.L., Levy, J.: On the structure of industrial SAT instances. In: Gent, I.P. (ed.) CP 2009. LNCS, vol. 5732, pp. 127–141. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04244-7_13
6. Ansótegui, C., Bonet, M.L., Levy, J.: Towards industrial-like random SAT instances. In: Boutilier, C. (ed.) 21st International Joint Conference Artificial Intelligence (IJCAI), pp. 387–392 (2009). http://ijcai.org/Proceedings/09/Papers/072.pdf
7. Berthet, Q.: Optimal testing for planted satisfiability problems. CoRR **abs/1401.2205**http://arxiv.org/abs/1401.2205 (2014)
8. Bläsius, T., Friedrich, T., Sutton, A.M.: On the empirical time complexity of scale-free 3-SAT at the phase transition. In: Vojnar, T., Zhang, L. (eds.) TACAS 2019. LNCS, vol. 11427, pp. 117–134. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-17462-0_7
9. Boufkhad, Y., Dubois, O., Interian, Y., Selman, B.: Regular random k-sat: Properties of balanced formulas. J. Autom. Reasoning **35**(1–3), 181–200 (2005)
10. Bradonjic, M., Perkins, W.: On sharp thresholds in random geometric graphs. In: Jansen, K., Rolim, J.D.P., Devanur, N.R., Moore, C. (eds.) Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2014. LIPIcs, vol. 28, pp. 500–514. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik (2014). https://doi.org/10.4230/LIPIcs.APPROX-RANDOM.2014.500
11. Bulatov, A.A., Skvortsov, E.S.: Phase transition for local search on planted SAT. In: Italiano, G.F., Pighizzini, G., Sannella, D.T. (eds.) MFCS 2015. LNCS, vol. 9235, pp. 175–186. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-48054-0_15
12. Ansótegui, C., Maria Luisa Bonet, J.L.: Scale-free random SAT instances. CoRR **abs/1708.06805**http://arxiv.org/abs/1708.06805 (2017)
13. Chao, M., Franco, J.V.: Probabilistic analysis of two heuristics for the 3-satisfiability problem. SIAM J. Comput. **15**(4), 1106–1118 (1986)
14. Chao, M., Franco, J.V.: Probabilistic analysis of a generalization of the unit-clause literal selection heuristics for the k satisfiability problem. Inf. Sci. **51**(3), 289–314 (1990)
15. Chvátal, V., Reed, B.A.: Mick gets some (the odds are on his side). In: 33rd Symposium Foundations of Computer Science (FOCS), pp. 620–627. IEEE Computer Society (1992). https://doi.org/10.1109/SFCS.1992.267789
16. Chvátal, V., Szemerédi, E.: Many hard examples for resolution. J. ACM **35**(4), 759–768 (1988)
17. Coja-Oghlan, A., Wormald, N.: The number of satisfying assignments of random regular k-SAT formulas. Comb. Prob. Comput. **27**(4), 496–530 (2018)
18. Doerr, B., Neumann, F., Sutton, A.M.: Time complexity analysis of evolutionary algorithms on random satisfiable k-CNF formulas. Algorithmica **78**(2), 561–586 (2017)
19. Feige, U., Mossel, E., Vilenchik, D.: Complete convergence of message passing algorithms for some satisfiability problems. Theor. Comput. **9**, 617–651 (2013)
20. Feldman, V., Perkins, W., Vempala, S.S.: On the complexity of random satisfiability problems with planted solutions. SIAM J. Comput. **47**(4), 1294–1338 (2018)
21. Flaxman, A.: A spectral technique for random satisfiable 3CNF formulas. Random Struct. Algorithms **32**(4), 519–534 (2008)
22. Flaxman, A.D.: Average-case analysis for combinatorial problems. Ph.D. thesis, Carnegie Mellon University (May 2006)

23. Franco, J., Paull, M.C.: Probabilistic analysis of the davis putnam procedure for solving the satisfiability problem. Discrete Appl. Math. **5**(1), 77–87 (1983)
24. Friedgut, E.: Sharp thresholds of graph properties, and the k-SAT problem. J. Amer. Math. Soc. **12**(4), 1017–1054 (1999)
25. Friedrich, T., Krohmer, A., Rothenberger, R., Sauerwald, T., Sutton, A.M.: Bounds on the satisfiability threshold for power law distributed random SAT. In: Pruhs, K., Sohler, C. (eds.) 25th European Symposium on Algorithms (ESA). LIPIcs, vol. 87, pp. 37:1–37:15. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik (2017). https://doi.org/10.4230/LIPIcs.ESA.2017.37
26. Friedrich, T., Rothenberger, R.: Sharpness of the satisfiability threshold for non-uniform random k-SAT. In: Kraus, S. (ed.) 28th International Joint Conference Artificial Intelligence (IJCAI), pp. 6151–6155. ijcai.org (2019). https://doi.org/10.24963/ijcai.2019/853
27. Giráldez-Cru, J., Levy, J.: Generating SAT instances with community structure. Artif. Intell. **238**, 119–134 (2016)
28. Giráldez-Cru, J., Levy, J.: Locality in random SAT instances. In: Sierra, C. (ed.) 26th International Joint Conference Artificial Intelligence (IJCAI), pp. 638–644. ijcai.org (2017). https://doi.org/10.24963/ijcai.2017/89
29. Hu, Y., Luo, W., Wang, J.: Community-based 3-sat formulas with a predefined solution. CoRR **abs/1902.09706**, http://arxiv.org/abs/1902.09706 (2019)
30. Koutsoupias, E., Papadimitriou, C.H.: On the greedy algorithm for satisfiability. Inf. Process. Lett. **43**(1), 53–55 (1992)
31. Krivelevich, M., Vilenchik, D.: Solving random satisfiable 3CNF formulas in expected polynomial time. In: 17th Symposium Discrete Algorithms (SODA), pp. 454–463. ACM Press (2006). http://dl.acm.org/citation.cfm?id=1109557.1109608
32. Mitchell, D.G., Selman, B., Levesque, H.J.: Hard and easy distributions of SAT problems. In: Swartout, W.R. (ed.) 10th Conference Artificial Intelligence (AAAI), pp. 459–465. AAAI Press/The MIT Press (1992). http://www.aaai.org/Library/AAAI/1992/aaai92-071.php
33. Omelchenko, O., Bulatov, A.A.: Satisfiability threshold for power law random 2-SAT in configuration model. In: Janota, M., Lynce, I. (eds.) SAT 2019. LNCS, vol. 11628, pp. 53–70. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-24258-9_4
34. Rathi, V., Aurell, E., Rasmussen, L., Skoglund, M.: Bounds on threshold of regular random k-SAT. In: Strichman, O., Szeider, S. (eds.) SAT 2010. LNCS, vol. 6175, pp. 264–277. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-14186-7_22
35. Selman, B., Kirkpatrick, S.: Critical behavior in the computational cost of satisfiability testing. Artif. Intell. **81**(1–2), 273–295 (1996)