Random Instances of W[2]-complete Problems: thresholds, complexity, and algorithms

Yong Gao⋆

Department of Computer Science
Irving K. Barber School of Arts and Sciences
University of British Columbia Okanagan
Kelowna, Canada V1V 1V7
yong.gao@ubc.ca

Abstract. The study of random instances of NP complete and coNP complete problems has had much impact on our understanding of the nature of hard problems as well as the strength and weakness of well-founded heuristics. This work is part of our effort to extend this line of research to intractable parameterized problems. We consider instances of the threshold dominating clique problem and the weighted satisfiability under some natural instance distribution. We study the threshold behavior of the solution probability and analyze some simple (polynomial-time) algorithms for satisfiable random instances. The behavior of these simple algorithms may help shed light on the observation that small-sized backdoor sets can be effectively exploited by some randomized DPLL-style solvers. We establish lower bounds for a parameterized version of the ordered DPLL resolution proof procedure for unsatisfiable random instances.

1 Introduction

The theory of parameterized complexity and fixed-parameter algorithms is becoming an active research area in recent years [1, 2]. Parameterized complexity provides a new perspective on hard algorithmic problems, while fixed-parameter algorithms have found applications in a variety of areas such as computational biology, cognitive modelling, and graph theory. Parameterized algorithmic problems also arise in many areas of artificial intelligence and satisfiability search. See, for example, the survey of Gottlob and Szeider [3].

Recently, some problems related to detecting *backdoor sets* for instances of the propositional satisfiability problem (SAT) have been studied from the perspective of parameterized complexity [4–6]. In particularly, the issue of the worst-case intractability versus the practical hardness of the backdoor detection problem has been raised: while the backdoor detection problem is NP-complete and/or fixed-parameter intractable for many types of backdoors, SAT solvers such as SATZ can exploit the existence of small-sized backdoors quite effectively [4, 5, 7].

⋆ Supported in part by NSERC Discovery Grant RGPIN 327587-06

The study of the parameterized proof complexity of the satisfiability problem has been initiated in [8] where lower bounds on the parameterized resolution proof are established for CNF formulas that encode some first-order combinatorial principle.

The study of random instances of NP and coNP complete problems such as SAT has had much impact on our understanding of the nature of hard problems, the strength of resolution proof systems, and the strength and weakness of algorithms and well-founded heuristics [9–15].

This work is part of our effort to extend this line of research to intractable parameterized problems [16]. We discuss random instances of problems whose parameterized version is W[2]-complete, including instances of the dominating clique problem from the Erdös-Renyi random graph and instances of the weighted CNF satisfiability problem from a carefully-designed random distribution.

We establish lower and upper bounds on the threshold of the phase transition of the solution probability, and show that in some region of the instance space, satisfiable instances can be solved by simple algorithms in polynomial/fixed-parameter time with high probability. Since finding a solution to a satisfiable instance of the parameterized problems under our consideration can be viewed as the task of detecting backdoor sets with respect to an (extremely) naive sub-solver that simply checks whether the all-zero assignment is a satisfying assignment, the behavior of these simple algorithms may help shed light on the observation that small-sized backdoor sets can be effectively exploited by some randomized DPLL-style solvers (See our discussion at the end of Section 3.2).

For random instances in the unsatisfiable region, we establish a lower bound on the search tree size of the parameterized version of a basic resolution proof procedure — the ordered DPLL algorithm.

In the next Section, we define necessary terminologies and notation. In Section 3, we discuss the random models and the main results. Sections 4 through 6 contain the proofs of the results.

2 Preliminaries

An instance of a *parameterized decision problem* is a pair (I, k) where I is a problem instance and k is an input parameter. A standard example is the parameterized vertex cover problem where an instance (I, k) consists of a graph I and a positive integer k, and the question is to decide whether the graph has a vertex cover of at most k vertices. A parameterized problem is fixed-parameter tractable (FPT) if any instance (I, k) can be solved in $f(k)|I|^{O(1)}$ time.

Parameterized problems are inter-related by parameterized reductions, resulting in a classification of parameterized problems into a hierarchy of complexity classes $FPT \subset W[1] \subset W[2] \cdots \subset XP$. At the lowest level is the class of FPT problems. The top level XP contains all the problems that can be solved in time $f(k)n^{g(k)}$. It is widely believed that the inclusions are strict and the notion of completeness can be naturally defined via parameterized reductions.

Domination-style problems such as the dominating set problem are representative $W[2]$-complete problems. However, the behavior of random instances of the dominating set problem is not interesting. For sparse random graphs $G(n,p)$ with $p \in o(1)$, the size of the minimum dominating set is larger than $\log n$. For dense random graphs $G(n,p)$ with $0 < p < 1$ a fixed constant, any vertex subset of size $\log_{1/(1-p)} n$ is a dominating set.

We consider the dominating clique problem in the Erdos-Renyi random graph $G(n,p)$, hoping that due to the clique constraint, random instances will have a much richer structure. The CNF formulas encoding the instances of the dominating clique problem are interesting since their structure bears similarities to that of the CNF formulas that encode instances of practical problems such as planning and model-checking, making them potential good benchmarks for the empirical study of satisfiability search algorithms [17]. Generalizing this observation, we further propose and study a random distribution defined by combining a W[1]-complete problem and a W[2]-complete problem: the weighted 2-CNF satisfiability and the general weighted CNF satisfiability.

2.1 The Threshold Dominating Clique Problem

Given a graph $G(V,E)$, we use $N(v)$ to denote the set of neighbors of a vertex $v \in V$ and use $N(U)$ to denote the open neighbor of a subset of vertices U, i.e.,

$$N(U) = \{v \in V \setminus U : N(v) \cap U \neq \phi\}.$$

The cardinality of a vertex set U is denoted by $|U|$. A clique is a subset of vertices that induces a complete subgraph. A dominating set is a subset V_D of vertices such that $N(v) \cap V_D \neq \phi, \;$ for all $v \in V \setminus V_D$.

We use $G(n,p)$ to denote the Erdös-Renyi random graph where n is the number of vertices and p is the edge probability. In $G(n,p)$, each of the possible $\binom{n}{2}$ edges appears independently with probability p. Throughout the paper when we say "with high probability", we mean that the probability of the event under consideration is $1 - o(1)$.

Definition 1. *Let $G(V,E)$ be a graph. An α-threshold dominating clique of G is a subset of vertices $V_D \subset V$ that induces a clique such that for all $v \in V \setminus V_D$, $|N(v) \cap V_D| \geq \alpha$. A 1-threshold dominating clique is simply called a dominating clique.*

The dominating clique problem is NP-complete and W[2]-complete when parameterized by the size of the clique [1]. The α-threshold dominating clique can also be shown to be W[2]-complete by a reduction from the threshold dominating set problem (see the Appendix of [1] for the definition of the threshold dominating set problem).

2.2 The Weighted CNF Satisfiability Problem

As in the theory of NP-completeness, the propositional satisfiability problem also plays an important role in the theory of parameterized complexity. A CNF formula over a set of Boolean variables is a conjunction of disjunctions of literals. A d-clause is a disjunction of d-literals. An assignment to a set of n Boolean variables is a vector in $\{0,1\}^n$. The **weight** of an assignment is the number of the variables that are set to 1 (true) by the assignment. A representative $W[1]$-complete problem is the following weighted d-CNF satisfiability problem (weighted d-SAT):

Problem 1. **Weighted d-SAT**

Instance: A CNF formula consisting of d-clauses and a positive integer k.

Question: Is there a satisfying assignment of weight k?

Unlike the traditional satisfiability problem, the weighted 2-SAT is already W[1]-complete. The anti-monotone weighted d-SAT problem (the problem where each clause contains negative literatures only) is also W[1]-complete. The *weighted satisfiability problem* (weighted SAT) is similar to the weighted d-SAT except that there is no restriction on the length of a clause in the formula. The weighted SAT is a generic W[2] complete problem.

A formal definition of a parameterized tree-like resolution proof system for weighted SAT is given in [8]. Basically, a parameterized resolution system can be regarded as a classical resolution system that has access (for free) to all clauses with more than k negated variables, where k is the parameter of the weighted SAT.

The most widely-used algorithms for the traditional satisfiability problem are variants of the Davis-Putnam-Logemann-Loveland (DPLL) procedure [18]. We consider the parameterized version of the DPLL algorithm for weighted SAT. It proceeds in the same way as the standard DPLL algorithm with the exception that a node in the search tree fails if

1. either a clause has been falsified by the partial assignment, or
2. the number of variables assigned to true in the partial assignment has exceeded k.

This way of parameterizing a proof procedure was proposed in [8].

We will provide a lower bound on a weaker version of the parameterized variants of the DPLL procedure — the parametric ordered DPLL. In the ordered DPLL [10], the variables are given a fixed order (before the algorithm starts). Except for the unit-propagation reduction steps, the variable selected to branch on is always the first one in the order that has not been assigned a value.

3 Main Results

3.1 Random Instances of Dominating Clique Problem

We use $\mathrm{DOMC}^{n,p}_{\alpha,k}$ to denote a random instance of the α-threshold dominating clique problem parameterized by the clique size k on the random graph $G(n,p)$. The exact threshold of the phase transition of the α-threshold dominating clique problem can be established for all α by extending the proof given in [17]. The threshold for any constant α (or α up to $\epsilon \log n$ with sufficiently small $\epsilon > 0$) turns out to be the same.

Theorem 1. *Consider the random graph $G(n,p)$. For any constant $\alpha \geq 1$,*

$$\lim_n Pr\{G(n,p) \text{ has an } \alpha\text{-threshold dominating clique }\}$$
$$= \begin{cases} 0, \text{ if } p < \frac{3-\sqrt{5}}{2} \\ 1, \text{ if } p > \frac{3-\sqrt{5}}{2}. \end{cases} \tag{1}$$

The size of an α-threshold dominating clique in $G(n,p)$, if exists, turns out to be in $\Omega(\log_{1/p} n)$. As a consequence, random instances of the parameterized α-threshold dominating clique problem with a fixed parameter is with high probability unsatisfiable for any $\mathrm{DOMC}_{\alpha,k}^{n,p}$ with $p < 1$. Due to this reason, the discussions in this subsection, especially those for the satisfiable instances, are in fact for the "LOGNP"-behavior of the problem. For future studies, one may want to consider parameterized problems that ask for a dominating clique of size $k \log_{1/\epsilon} n$ for some small constant ϵ. This difficulty largely motivates the random weighted SAT distribution to be discussed in the next subsection.

The Unsatisfiable Instances

Two exact algorithms for the dominating clique problem have been proposed [19, 17]. The one proposed in [19] is shown to have a time complexity $O(1.339^n)$ while the one studied in [17] empirically works well on random graphs (In fact by adding a few simple cases, which never happen in random graphs, the algorithm studied in [17] can be shown to have a time complexity $O(1.383^n)$ by a simple analysis). In the following, we lower bound the search tree size of the ordered DPLL algorithm which is weaker than the above two branch-and-reduce algorithms, but is of interest in the study of proof complexity and logic inferences.

The parameterized dominating clique problem can be encoded as a weighted SAT problem as follows. Given a graph $G(V,E)$, we associate with each vertex with a Boolean variable. Let $\{x_1, \cdots, x_n\}$ be the set of variables corresponding to the set of vertices $V = \{v_1, \cdots, v_n\}$. The CNF formula consists of two types of clauses:

1. Anti-monotone 2-clauses. For each pair of vertices v_i and v_j such that $(v_i, v_j) \notin E$, there is a 2-clause $\overline{x}_i \vee \overline{x}_j$. This set of clauses enforces the clique constraint.
2. Monotone long clauses. For each vertex v_i, there is a clause

$$x_i \vee x_{i_1} \vee \cdots \vee x_{i_l}$$

where $\{x_{i_1}, \cdots, x_{i_l}\}$ are the neighbors of v_i. This set of clauses enforces the domination requirement.

The following theorem provides a lower bound on the size of the search tree of the parametric ordered DPLL resolution proof. Note that the result is more general than needed — we allow k to be as large as $\epsilon \log n$.

Theorem 2. *For any parameter $0 < k < \epsilon \log n$ where $\epsilon > 0$ is a small constant and any $0 < p < 1$, the size of the search tree of the parametric ordered DPLL algorithm for $DOMC^{n,p}_{k,\alpha}$ is $n^{\Omega(k)}$ with high probability.*

The Satisfiable Instances

On the positive side, we show that for any $p > \frac{1}{2}$, an α-threshold dominating clique of size $\Omega(\log n)$ in $G(n,p)$ can be found in $O(n^2)$ time with high probability.

Theorem 3. *There is an $O(n^2)$-time algorithm that with high probability, finds an α-threshold dominating clique of size $\Omega(\log n)$ in $G(n,p)$ with $p > \frac{1}{2}$.*

We consider the following greedy algorithm, G-DOMC. Except for the first α-steps, at any moment, the vertices of the graph are in one of the following groups:

1. V_C: the clique obtained so far;
2. V_W: vertices that are adjacent to every vertex in V_C;
3. $V_i, 0 \leq i \leq \alpha - 1$: a vertex v is in V_i if it is adjacent to exactly i vertices in V_C.
4. V_α: vertices that have been dominated by at least α vertices in V_C, but are not in V_W.

Vertices in V_α are those that have been α-threshold dominated but cannot be used to expand the current clique. Hence they play no role in the algorithm. It is easy to see that after the first α steps, vertices in V_W have been dominated by more than α vertices in V_C so that we do not need to worry about their domination. Since α is a fixed constant, it can be shown that with high probability, the algorithm will not terminate within the first α steps.

The algorithm G-DOMC repeatedly picks a random vertex in V_W to expand the current clique and updates the vertex sets V_W and V_i's accordingly, as shown in the following pseudo-code:

1. Initialization: $V_C = \phi$, $V_W = V$, and $V_i = \phi, 0 \leq i \leq \alpha$;
2. Repeat until either $V_i = \phi, \forall i \leq \alpha$ or $V_W = \phi$
   (a) randomly pick a vertex v in V_W
   (b) $V_C = V_C \cup \{v\}$; $V_W = V_W \cap N(v)$;
   (c) For each $0 \leq i \leq \alpha - 1$,

$$V_i = (V_i \setminus (N(v) \cup \{v\})) \cup (V_{i-1} \cap N(v))$$

Let $X(t)$ be the size of V_W after the t-th iteration and $Y_i(t)$ be the size of V_i after the t-th iteration. Intuitively since $p > \frac{1}{2}$, each vertex is adjacent to more than half of the vertices. If we construct the clique by greedily picking one of the potential vertices, then the number of potential vertices that can be used to expand the current clique decreases at a slower rate than the number of vertices that still need to be dominated. Consequently, all the vertices will be dominated before there is no way to expand the current clique.

Formally, we will prove that

$$\mathbb{P}\left\{X(t) > n^{\delta_1} \text{ and } Y_i(t) = 0,\ \forall\, 0 \leq i \leq \alpha - 1\right\} > 1 - O(\frac{1}{n^{\delta}}) \tag{2}$$

where $\delta > 0$ and $\delta_1 > 0$ are properly determined small constants and

$$t = -\frac{1+\delta}{\log(1-p)} \log n,$$

which guarantees that the algorithm finds an α-threshold dominating clique at step t with high probability. For the formal proof, see Section 5.

3.2 A Random Model for Weighted SAT

To have a random distribution that generates interesting instances for fixed parameters of some dominating-style problem, we propose the following model $\mathcal{M}_{n,k}^{p_1,p_2,m}$ for weighted SAT.

Definition 2. *An instances of* $\mathcal{M}_{n,k}^{p_1,p_2,m}$ *consists of*

1. *a collection of anti-monotone 2-clauses. Each of the potential* $\binom{n}{2}$ *anti-monotone clauses is included independently with probability* p_1*, and*
2. m *monotone clauses obtained independently in the following way: for each clause, each of the* n *variables appear with probability* p_2.

The number of variables is n *and the input parameter is* k.

It is a concern that the monotone clauses generated in the above may be trivial, either being empty or containing all the variables. This is not the case — it can be shown that for the range of m we are considering, all the clauses contain $p_2 n + o(n)$ variables with high probability.

We have the following result on the threshold of the phase transition of the solution probability.

Theorem 4. *Assume that* $0 < p_1, p_2 < 1$ *are fixed constants. Let* $b = (1 - (1 - p_2)^k)$*, and* $m = c \log n$*. The probability that a random instance of* $\mathcal{M}_{n,k}^{p_1,p_2,m}$ *has a solution is*

$$\lim_n \mathbb{P}\left\{\mathcal{M}_{n,k}^{p_1,p_2,m} \textit{ has a solution }\right\} = \begin{cases} 1, \textit{ if } c < -\frac{1}{\log b} \\ 0, \textit{ if } c > -\frac{k}{\log b}. \end{cases}$$

For the case of $c \log b > -1$, the proof of the above theorem actually indicates that the fraction of the satisfying assignments is in a "fixed-parameter" form. As a consequence, by simply sampling the assignments of weight k, we can find a satisfying assignment of weight k. The average number of samples needed is in a "fixed-parameter" for a typical instance from $\mathcal{M}_{n,k}^{p_1,p_2,m}$. (Note however that the average is taken with respect to the sampling process only.)

Corollary 1. *Let $m = c \log n$ such that $c \log b > -1$. There is a randomized algorithm that solves the satisfiable instances of $\mathcal{M}_{n,k}^{p_1,p_2,m}$ in $2^{O(k^2)} n^{O(1)}$ time.*

Proof. Consider the algorithm that repeatedly and randomly picks an assignment of weight k until a satisfying assignment is found.

Let $a = (1-p_1)$. From the proof of Theorem 4, we see that with high probability, an instance of $\mathcal{M}_{n,k}^{p_1,p_2,m}$ has more than $a^{\binom{k}{2}} \binom{n}{k} n^{c \log b}$ satisfying assignments.

For a typical (but fixed) instance from $\mathcal{M}_{n,k}^{p_1,p_2,m}$, the probability that a randomly-picked assignment is satisfying is

$$\frac{a^{\binom{k}{2}} \binom{n}{k} n^{c \log b}}{\binom{n}{k}} = a^{\binom{k}{2}} n^{c \log b}.$$

Thus, the average number of samples (with respect to the sampling process) required before a satisfying assignment is found is $2^{O(k^2)} n^{-c \log b}$.

To relate Corollary 1 to the backdoor set detection problem, consider the (extremely) naive sub-solver that simply checks whether the all-zero assignment is a satisfying assignment. Corollary 1 says that such a backdoor set can be found by sampling the $\binom{n}{k}$ possibilities $2^{O(k^2)} n^{-c \log b}$ times, and thus provides a theoretical support to the observation that the existence of samll-sized backdoor sets can be effectively exploited by randomized DPLL-style solvers with random restarts such as SATZ [7].

Similar to the case of the threshold dominating clique problem, a lower bound on the parametric ordered DPLL algorithm for unsatisfiable instances can be established.

Theorem 5. *Let $m = c \log n$ such that $c \log b < -k$. Then with high probability, the size of the search tree of the parametric ordered DPLL for instances of $\mathcal{M}_{n,k}^{p_1,p_2,m}$ is $n^{\Omega(k)}$.*

4 Proof of Theorem 2

Proof. We focus on the case of 1-threshold dominating clique. Let $V = \{v_1, \cdots, v_n\}$ be an ordering of the vertices and assume without loss of generality that this is also the order used by the order DPLL algorithm. Let $i = \beta n$ where $\beta > 0$ is a constant, $V_0 = \{v_1, \cdots, v_i\}$, and $U = V \setminus V_0$.

Let $\mathcal{D}$ be the collection of subsets of vertices in V_0 of size $\frac{k}{2}$ and denote by $\mathcal{N}(D)$ the set of vertices in U that are adjacent to every vertex in the vertex set $D \in \mathcal{D}$, i.e.,

$$\mathcal{N}(D) = \{u \in U \mid \forall w \in D, (u, w) \text{ is an edge}\}.$$

We say that a vertex set D in $\mathcal{D}$ is *promising* if

1. D induces a clique in $G(n, p)$, and
2. $N(v) \cap \mathcal{N}(D) \neq \phi$ for any vertex $v \in V \setminus (D \cup \mathcal{N}(D))$.

We claim that the size of the DPLL search tree is lower bounded by the number of promising vertex sets in $\mathcal{D}$. To see this, consider a subset of vertices

$$D = \{v_{i_1}, \cdots, x_{i_{\frac{k}{2}}}\} \subset V_0$$

and a path of length βn in the ordered DPLL search tree along which variables in D are assigned to true and the other variables on the path are assigned to false. Since D induces a clique, no anti-monotone clause has been falsified by the partial assignment. Since the variables in $\mathcal{N}(D)$ are those that have not been forced by the assignment to the variables in D, the fact that $N(v) \cap \mathcal{N}(D) \neq \phi$ implies that the monotone long clause enforcing the domination of vertex v is not empty yet. Therefore, this path will be explored by the ordered DPLL algorithm. This proves the claim.

To proceed, we first show that the size of $\mathcal{N}(D)$ is large with high probability. Since a vertex u is in $\mathcal{N}(D)$ if and only if it is adjacent to every vertex in D, the expected size of $\mathcal{N}(D)$ is

$$\mathbb{E}\left[|\mathcal{N}(D)|\right] = (1-\beta)np^{\frac{k}{2}}.$$

Let $I_D(u)$ be the indicator function of the event that u is in $\mathcal{N}(D)$. Due to the independence of the edges in $G(n,p)$, the variables $\{I_D(u), u \in U\}$ are independent Bernoulli variables with mean $p^{\frac{k}{2}}$. By the Chernoff bound (see, for example, [20]), we have

$$\mathbb{P}\left\{|\mathcal{N}(D)| > \frac{1}{2}(1-\beta)np^{\frac{k}{2}}\right\} \geq 1 - 2e^{-\frac{1}{2}(1-\beta)np^{\frac{k}{2}}}. \tag{3}$$

To complete the proof of the theorem, we show that with high probability, there are $n^{\Omega(k)}$ promising vertex sets. From Equation (3), we have for a fixed vertex set $D \in \mathcal{D}$,

$$\begin{aligned}&\mathbb{P}\left\{D \text{ is promising } | D \text{ induces a clique}\right\}\\&\quad\geq O(1)\left(1-(1-p)^{\frac{1}{2}(1-\beta)np^{\frac{k}{2}}}\right)^n \geq O(1)\end{aligned} \tag{4}$$

since p is a fixed constant.

Let X be the number of vertex subsets in $\mathcal{D}$ that are promising. The expectation of X satisfies

$$O(1)\binom{\beta n}{k/2}p^{\binom{k/2}{2}} \leq \mathbb{E}\left[X\right] \leq \binom{\beta n}{k/2}.$$

Therefore $\mathbb{E}[X]$ is in $n^{\Omega(k)}$ as long as $k < \epsilon \log n$ for some $\epsilon = \epsilon(p) > 0$. To complete the proof, we apply Chebyshev's inequality

$$\mathbb{P}\left\{|X - \mathbb{E}\left[X\right]| > \frac{1}{2}\mathbb{E}\left[X\right]\right\} \leq \frac{4\mathbb{E}\left[(X-\mathbb{E}\left[X\right])^2\right]}{(\mathbb{E}\left[X\right])^2}. \tag{5}$$

and show that

$$\mathbb{E}\left[(X-\mathbb{E}[X])^2\right]=\mathbb{E}\left[X^2\right]-(\mathbb{E}[X])^2=o(\mathbb{E}[X])^2,$$

which can be established by estimating the probability that two overlapping vertex sets D_1, D_2 in $\mathcal{D}$ are both promising. We omit the lengthy detail due to space limit.

5 Proof of Theorem 3

We prove the theorem by showing that with high probability, G-DOMC terminates with an α-threshold dominating clique of size $\Omega(\log n)$. Recall that in the algorithm G-DOMC, after the first α steps, the vertices of the graph are in one of the following groups: V_C (the current clique), V_W (vertices adjacent to every vertex in V_C), V_i (vertices dominated by exactly i vertices in V_C), and V_α (the finished vertices).

Let X_t be the size of V_W after the t-th iteration and Y_t^i be the size of V_i be the size of V_i after the t-th iteration. First, we have the following lemma

Lemma 1. *The number of vertices in V_W after the first α steps satisfies*

$$\mathbb{P}\left\{X_\alpha>\frac{1}{2}p^\alpha n\right\}\geq 1-O(e^{-n}).$$

Due to the above lemma, we will assume that $X_\alpha > \frac{1}{2}p^\alpha n$, which further implies that $Y_t^i \leq (1-\frac{1}{2}p^\alpha)n$ for any $0 \leq t \leq \alpha$.

Due to the assumption $p > \frac{1}{2}$, there exist small constants $\delta > 0$ and $\epsilon > 0$ such that

$$(1-\epsilon)p > (1-p)^{\frac{1}{1+\delta}}.$$

Let

$$t = -\frac{1+\delta}{\log(1-p)}\log n, \text{ and}$$
$$\delta_1 = 1-\frac{1+\delta}{\log(1-p)}\log(1-\epsilon)p > 0.$$

We will show that

$$\mathbb{P}\left\{X_t > n^{\delta_1} \text{ and } Y_t^i = 0,\ \forall\ 0\leq i\leq \alpha-1\right\} > 1-O(\frac{1}{n^\delta}) \tag{6}$$

which indicates that with high probability, the algorithm G-DOMC terminates in t steps and finds an α-threshold dominating clique. We first consider the probability of the event $\{X_t > n^{\delta_1}\}$.

Lemma 2.

$$\mathbb{P}\left\{X_t > n^{\delta_1}\right\} \geq (1-\frac{c}{n})^t.$$

for some fixed constant $c > 0$.

Proof. Recall that $X_t = |V_W|$ is the number of white vertices after step t. After the first α steps, the set of white vertices "evolve" on its own — no vertex in V_i's can become a white vertex and vertices in V_W have been α-dominated.

In step t, a vertex v in V_W is randomly selected, and the new V_W is formed as $V_W = V_W \cap N(v)$. Since in $G(n,p)$, the edges appear in the graph independently, by the "deferred decision" argument, we see that $\{X_t, t \geq \alpha\}$ is a Markovian chain.

Write $a_t = (1-\epsilon)^t p^t n = n^{\delta_1}$. We have

$$\begin{aligned}
&\mathbb{P}\left\{X_t > n^{\delta_1}\right\} = \mathbb{P}\left\{X_t > a_t\right\} \\
&\geq \mathbb{P}\left\{X_s > a_s \text{ for all } \alpha \leq s \leq t\right\} \\
&= \prod_{s=\alpha}^{t} \mathbb{P}\left\{X_s > a_s \mid X_{s-1} > a_{s-1}\right\}
\end{aligned}$$

We claim that

$$\mathbb{P}\left\{X_s > a_s \mid X_{s-1} > a_{s-1}\right\} \geq \mathbb{P}\left\{Bin(p, a_{s-1}) > a_s\right\}.$$

where $Bin(p, a_{s-1})$ is a random variable that has a binomial distribution with parameters p and a_{s-1}, i.e., $Bin(p, a_{s-1})$ is the sum of a_{s-1} Bernoulli random variables with mean p. To see this, recall that X_{s-1} is the number of vertices in V_W after step $s-1$ of G-DOMC. In step s, each of the white vertices survives with probability p, and the events that white vertices survive are mutually independent.

By the Chernoff bound on the tail probability of Bernoulli variables, we have

$$\begin{aligned}
&\mathbb{P}\left\{X_s > a_s \mid X_{s-1} > a_{s-1}\right\} \\
&\quad = \mathbb{P}\left\{X_s > (1-\epsilon)pa_{s-1} \mid X_{s-1} > a_{s-1}\right\} \\
&\quad \geq 1 - e^{-\frac{\epsilon^2 p^2}{2} a(s-1)}
\end{aligned}$$

Since $a_t = (1-\epsilon)^t p^t n$ and by the choice of t, ϵ, we see that $e^{-\frac{\epsilon^2 p^2}{2} a_{s-1}} \leq e^{-n_1^{\delta}} \in O(\frac{1}{n})$. The Lemma follows.

We now bound the conditional probability that $Y_t^i > 0$ given that $X_t > n^{\delta_1}$.

Lemma 3. *Given that $X_t > n^{\delta_1}$ (i.e., the algorithm does not terminate due to the lack of vertices to expand the clique), we have*

$$\mathbb{P}\left\{Y_t^i > 0 \textit{ for some } 0 \leq i \leq \alpha - 1\right\} \leq O(\frac{\alpha}{n^{\delta}}).$$

Proof. Recall that after the first α steps, there will be no vertex-exchange between V_W and the V_i's. Therefore, the probabilistic behavior of the system of vertex sets $\{V_i, 0 \leq i \leq \alpha - 1\}$ is independent of the specific choice of the vertex in V_W (given that $|V_W| = X_t > 0$ so that there is always a vertex to pick).

In step t, some vertices in V_i move to V_{i+1} because they are connected to the vertex just added to the clique. Due to the same reason, there are also vertices moving from V_{i-1} to V_i. Therefore, the expectation of $Y_i(t)$ is

$$\mathbb{E}\left[Y_t^0\right] = (1-p)\mathbb{E}\left[Y_{t-1}^0\right],$$
$$\mathbb{E}\left[Y_t^i\right] = (1-p)\mathbb{E}\left[Y_{t-1}^i\right] + p\mathbb{E}\left[Y_{t-1}^{i-1}\right].$$

Write $y_t^i = \mathbb{E}\left[Y_t^i\right]$. By induction, we have

$$y_t^i \leq Ct^i(1-p)^t n$$

for some constant $C > 0$. Consequently by Markov's inequality, we have

$$\mathbb{P}\left\{Y_t^i > 0\right\} \leq y_t^i \leq \frac{1}{n^\delta}.$$

Therefore,

$$\mathbb{P}\left\{Y_t^i > 0 \text{ for some } 0 \leq i \leq \alpha - 1\right\} \leq \frac{\alpha}{n^\delta}.$$

The lemma follows.

Combining Lemma 2 and Lemma 3, we see that Equation (6) holds. This proves Theorem 3.

6 Proof of Theorem 4

Let S be the set of assignments of weight k. For each $s \in S$, let I_s be the indicator function of the event that s satisfies $\mathcal{M}_{n,k}^{p_1,p_2,m}$. Consider the random variable $X = \sum_{s\in S} I_s$, the number of assignments in S that satisfy $\mathcal{M}_{n,k}^{p_1,p_2,m}$. Write $a = 1 - p_1$ and $b = 1-(1-p_2)^k$, and recall that $m = c\log n$. We have

$$\mathbb{E}[X] = \binom{n}{k} a^{\binom{k}{2}} n^{c\log b}.$$

The case of $c > -\frac{k}{\log b}$ follows from Markov's inequality. For the case of $c < -\frac{1}{\log b}$, we consider the variance $\mathrm{V}(X)$ of $X = \sum_{s\in S} I_s$. We say that two assignments in S have i overlaps if there are exactly i variables that are set to true by both of the two assignments. Let $S(i)$ be the set of (ordered) pairs of assignments in S that have i overlaps. $\mathrm{V}(X)$ can be written as

$$\begin{aligned}\mathrm{V}(X) &= \mathbb{E}\left[(X - \mathbb{E}[X])^2\right] \\ &\leq \mathbb{E}[X] + \sum_{i=0}^{k} \sum_{(s_1,s_2)\in S(i)} (\mathbb{E}[I_{s_1}I_{s_2}] - \mathbb{E}[I_{s_1}]\mathbb{E}[I_{s_2}]). \qquad (7)\end{aligned}$$

For any $(s_1, s_2) \in S(0)$, it is easy to see that

$$\mathbb{E}[I_{s_1}I_{s_2}] - \mathbb{E}[I_{s_1}]\mathbb{E}[I_{s_2}] = 0.$$

Consider $(s_1, s_2) \in S(i)$ with $i > 0$. We have

$$\mathbb{E}[I_{s_1} I_{s_2}] - \mathbb{E}[I_{s_1}]\,\mathbb{E}[I_{s_2}] = (1-p_1)^{\binom{2k-i}{2}}(1 - 2(1-p_2)^k + (1-p_2)^{2k-i})^m \\ -(1-p_1)^{2\binom{k}{2}}(1-(1-p_2)^k)^{2m}$$

For sufficiently large n, the above can be upper bounded by

$$\begin{aligned} &C(1-p_1)^{\binom{2k-i}{2}}\left(1 - 2(1-p_2)^k + (1-p_2)^{2k-i}\right)^m \\ \leq\ &C(1-p_1)^{\binom{2k-i}{2}}\left(1-(1-p_2)^k\right)^m. \end{aligned}$$

where $C > 0$ is a fixed constant. Therefore, we have

$$\begin{aligned} &\sum_{i=1}^{k} \sum_{(s_1,s_2)\in S(i)} (\mathbb{E}[I_{s_1} I_{s_2}] - \mathbb{E}[I_{s_1}]\,\mathbb{E}[I_{s_2}]) \\ &\leq Ck\binom{n}{2k-1}(1-p_1)^{\binom{2k-1}{2}}\left(1-(1-p_2)^k\right)^m. \end{aligned}$$

It follows that

$$\begin{aligned} \frac{\mathrm{V}(X)}{(\mathbb{E}[X])^2} &\leq \frac{\mathbb{E}[X] + Ck\binom{n}{2k-1}\delta(1-p_1)^{\binom{2k-1}{2}}\left(1-(1-p_2)^k\right)^m}{n^{2k}(1-p_1)^{2\binom{k}{2}}\left(1-(1-p_2)^k\right)^{2m}} \\ &\in O(\frac{1}{n^{1+c\log b}}). \end{aligned}$$

For the case $c < -\frac{1}{\log b}$, write $0 < \epsilon = -c\log b < 1$. We have by Chebyshev's inequality

$$\mathbb{P}\left\{|X - \mathbb{E}[X]| > n^{\frac{1-\epsilon}{4}} \frac{1}{n^{\frac{1-\epsilon}{2}}}\mathbb{E}[X]\right\} \leq \frac{1}{n^{(1-\epsilon)/2}}.$$

Recall that $\mathbb{E}[X] = \binom{n}{k}a^{\binom{k}{2}}(1-(1-p_2)^k)^m$. It follows that with probability $1 - O(\frac{1}{n^{\epsilon/2}})$, we have

$$X \geq a^{\binom{k}{2}} n^{k-\epsilon} - o(n^{k-\epsilon}). \tag{8}$$

This completes the proof.

7 Conclusions

In this paper, we have studied the behavior of random instances of two W[2]-complete problems. The threshold behavior of the solution probability under the proposed random models is studied. Lower and upper bounds on the complexity of satisfiable and unsatisfiable instances are established.

It is interesting to see if the dominating clique problem on random graphs with $\frac{3-\sqrt{5}}{2} < p < \frac{1}{2}$ can be solved in polynomial time with high probability. Establishing lower bounds on the proof complexity of more general parameterized resolution proof system is a challenging future task.

There is a gap between the lower and upper bounds on the threshold of the solution probability of the random weighted SAT model. Closing the gap is interesting. Identifying more scenarios that lead to fixed-parameter tractable class of instances is perhaps even more interesting.

References

1. R. Downey, M. Fellows, Parameterized Complexity, Springer, 1999.
2. R. Neidermeier, Invitation to Fixed-Parameter Algorithms, Oxford University Press, 2006.
3. G. Gottlob, S. Szeider, Fixed-parameter algorithms for artificial intelligence, constraint satisfaction, and database problems, The Computer Journal (to appear).
4. B. Dilkina, C. Gomes, A. Sabharwal, Tradeoffs in the complexity of backdoor detection, in: Proceedings of CP'07, 2007, pp. 256–270.
5. S. Szeider, Backdoor sets for DLL subsolvers, Journal of Automated Reasoning (1 - 3) (2005) 73–88.
6. N. Nishimura, P. Ragde, S. Szeider, Detecting backdoor sets with respect to horn and binary clauses, in: Proceedings of SAT'04, 2004.
7. R. Williams, G. Gomes, B. Selman, On the connections between heanvy-tails, backdoors, and restarts in combinatorial search, in: Proceedings of SAT'03, 2003, pp. 222–230.
8. S. Dantchev, B. Martin, S. Szeider, Parameterized proof complexity, in: Proceedings of FOCS'07, IEEE Press, 2007, pp. 150–160.
9. D. Achlioptas, P. Beame, M. Molloy, A sharp threshold in proof complexity, in: Proceedings of STOC'01, 2001, pp. 337–346.
10. P. Beame, R. Karp, T. Pitassi, M. Saks, The efficiency of resolution and Davis-Putnam procedures, SIAM Journal on Computing 31 (4) (2002) 1048–1075.
11. P. Cheeseman, B. Kanefsky, W. Taylor, Where the *really* hard problems are, in: Proceedings of IJCAI'91, Morgan Kaufmann, 1991, pp. 331–337.
12. S. Cook, D. Mitchell, Finding hard instances of the satisfiability problem: A survey, in: Du, Gu, Pardalos (Eds.), Satisfiability Problem: Theory and Applications, Vol. 35 of DIMACS Series in Discrete Mathematics and Theoretical Computer Science, American Mathematical Society, 1997.
13. I. Gent, T. Walsh, Analysis of heuristics for number partitioning, Computational Intelligence 14 (3) (1998) 430–451.
14. C. Gomes, C. Fernandez, B. Selman, C. Bessiere, Statistical regimes across constrainedness regions, Constraints 10 (4) (2005) 313–337.
15. Y. Gao, J. Culberson, Consistency and random constraint satisfaction models, Journal of Artificial Intelligence Research 28 (2007) 517–557.
16. Y. Gao, Random instances of parameterized complete problems: phase transitions and complexity, Tech. rep., Computer Science, University of British Columbia Okanagan (2007).
17. J. Culberson, Y. Gao, C. Anton, Phase transitions of dominating clique problem and their implications to heuristics in satisfiability search, in: Proceedings of IJCAI'05, 2005, pp. 78–83.
18. M. Davis, G. Logemann, D. Loveland, A machine program for theorem proving, Communications of the ACM 5 (1962) 394–397.
19. D. Kratsch, M. Liedloff, An exact algorithm for the minimum dominating clique problem, Theoretical Computer Science 385 (1-3) (2007) 226–240.
20. N. Alon, J. H. Spencer, The Probabilistic Method, Wiley, 2000.