Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence (2008)

Phase Transitions and Complexity of Weighted Satisfiability and Other Intractable Parameterized Problems

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. In this work, we initiate an effort to extend this line of research to random instances of intractable parameterized problems.

We propose random models for a representative intractable parameterized problem, the weighted d-CNF satisfiability, and its generalization to the constraint satisfaction problem. The exact threshold for the phase transition of the proposed models is determined. Lower bounds on the time complexity of variants of the DPLL algorithm for these parameterized problems are also established. In particularly, we show that random instances of the weighted 2-CNF satisfiability, already an intractable parameterized problem, are typically easy in both of the satisfiable and unsatisfiable regions by exploiting an interesting connection between the unsatisfiability of a weighted 2-CNF formula and the existence of a Hamiltonian-cycle-like global structure.

1. Introduction

The theory of parameterized complexity and fixed-parameter algorithms is becoming an active research area in recent years (Downey and Fellows 1999; Neidermeier 2006). 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 modeling, graph theory, and various optimization problems.

Parameterized algorithmic problems also arise in many areas of artificial intelligence research. See, for example, the survey of Gottlob and Szeider (Gottlob and Szeider 2007). Recently, some problems related to detecting *backdoor sets* for instances of the propositional satisfiability problem (SAT) have been studied from the perspective of parameter complexity (Szeider 2005; Nishimura, Ragde, and Szeider 2004). A backdoor (to instances of a hard problem) is a subset of variables that, once fixed, will result in one or more polynomial time sovable subproblems. The existence of small sized backdoors naturally leads to fixed-parameter algorithms. For example, in Bayesian networks and constraint networks, a feedback vertex set of the (directed) network is such as backdoor. 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 effectively (Dilkina, Gomes, and Sabharwal 2007; Szeider 2005).

The study of parameterized proof complexity of the propositional satisfiability problem has been initiated in (Dantchev, Martin, and Szeider 2007), 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-complete problems and coNP-complete problems such as SAT has had much impact on our understanding of the nature of hard problems (Achlioptas, Beame, and Molloy 2001; Beame et al. 2002; Cheeseman, Kanefsky, and Taylor 1991) as well as the strength and weakness of well-founded algorithms and heuristics (Gent and Walsh 1998; Gomes et al. 2005; Gao and Culberson 2007).

In this work, we initiate an effort to extend this line of research to intractable parameterized problem in the parameterized complexity hierarchy. We propose random models for a representative parameterized intractable problem, the weighted d-CNF satisfiability, and its variants. We establish the exact threshold of the phase transition of the proposed model. This is in contrast to the threshold behavior of most of the NP complete problems, such as the satisfiability and graph coloring, where the exact threshold is extremely hard to nail down.

For the weighted 2-CNF satisfiability problem, which is already W[1]-complete, we show that random instances are typically easy in both of the satisfiable and unsatisfiable regions. We prove that random (unsatisfiable) instances of the weighted 2-CNF satisfiability problem have a "fixed-parameter-sized" resolution proof by exploiting an interesting link between the unsatisfiability of a weighted 2-CNF formula and the existence of a Hamiltonian cycle in a properly defined directed graph. This is somewhat surprising and indicates that the typical-case behavior of random instances of parameterized problems are quite different from that of NP-complete problems. First, it is the existence of a global structure (a directed Hamiltonian cycle) that leads to easy instances, as compared to the case of NP-complete problems where small structures are usually the signature of easy instances. Secondly, unlike the case of traditional satisfiability, the weighted 2-CNF satisfiability problem is already fixed-parameter intractable. As a comparison, for weighted d-CNF satisfiability problem with $d > 2$, we establish a non-fixed parameter tractable lower bound on the search tree size of the parameterized version of the ordered DPLL algorithm.

We observe that there are certain relations between finding a solution to an instance of the parameterized problem under consideration the tasks of backdoor detection with respect to an (extremely) naive sub-solver that simply checks whether the all-zero assignment is satisfying, and hope that further investigation along this line may help shed further light on the (typical-case) hardness of the backdoor detection problem and the empirical observation on the effectiveness of practical satisfiability solvers in exploiting small-sized backdoors.

The results reported in the current paper are all of theoretical nature and the objective is to lay the necessary foundation for a study of the phase transitions and typical-case hardness of fixed-parameter intractable problems. We believe that in the future study, empirical studies are indispensable, especially for situations where theoretical analyses are hard to come by. See, for example, the work in (Gent and Walsh 1998; Gomes et al. 2005; Gao and Culberson 2007) and the references therein for the many empirical studies on the phase transitions of NP-hard problems.

2. Preliminaries

Parameterized complexity

An instance of a *parameterized decision problem* is a pair (I, k) where I is a problem instance and k is the problem parameter. Usually, the parameter k either specifies the "size" of the solution or is related to some structural property of the underlying problem, such as the treewidth of a graph. For example, in an instance (I, k) of the parameterized vertex cover (VC) problem, I is a graph and k is the size of the vertex cover. The question is to decide whether I has a vertex cover of size k.

A parameterized problem is fixed-parameter tractable (FPT) if any instance (I, k) of the problem can be solved in $f(k)|I|^{O(1)}$ time, where $f(k)$ is a computable function of k only. It is known that the parameterized VC problem can be solved in $O(1.28^k + kn)$ time where n is the number of vertices of the input graph.

Parameterized problems are inter-related by parameterized reductions, resulting in a classification of parameterized problems into a hierarchy of complexity classes

$$FPT \subseteq W[1] \subseteq W[2] \cdots \subseteq XP.$$

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.

As in the case of the theory of NP-completeness, the satisfiability problem plays an important role in the theory of parameterized complexity (Downey and Fellows 1999). 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.

Definition 1. *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, i.e., the number of 1's in the assignment.*

A representative $W[1]$-complete problem is the following weighted d-CNF satisfiability problem (weighted d-SAT) (Downey and Fellows 1999):

Problem 1. ***Weighted d-SAT***

Input: *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, 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 weighted d-SAT except that there is no restriction on the length of a clause in the formula. Weighted SAT is a representative $W[2]$-complete problem.

Parameterized Proof Systems and Parameterized DPLL Algorithms

A formal definition of a parameterized tree-like resolution proof system for the weighted d-SAT problem is given in (Dantchev, Martin, and Szeider 2007). Basically, a parameterized resolution can be regarded as a classical resolution that have access (for free) to all clauses with more than k negated variables, where k is the parameter of the weighted SAT problem.

Accordingly, one can 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,
2. or the number of variables assigned to true in the partial assignment has exceeded k.

In addition to the parameterized version of the general DPLL algorithm, we may also consider the parameterized version of *ordered*-DPLL and the *oblivious*-DPLL algorithms studied in (Beame et al. 2002). In the ordered DPLL, an ordering of the variables is fixed and the next variable selected to branch on is always the first one that has not been assigned a value. In the oblivious-DPLL algorithm, the variable orderings for different branches of the DPLL search tree may be different, but these orderings are pre-specified before the algorithm starts and are independent of the formula.

3. Main Results

We study the behavior of random instances of weighted d-SAT and its variants. In the study of the phase transitions of traditional satisfiability problem, a widely-used instance distribution is the random d-CNF formula that consists of a set of m clauses selected uniformly at random with (or without) replacement from all the $2^d\binom{n}{d}$ possible d-clauses.

However, it is problematic to use the above instance distribution for parameterized problems such as weighted d-SAT. For example, under this distribution, random instances of the anti-monotone weighted 2-SAT (a W[1]-complete problem), are always satisfiable and a solution can be found by the simple algorithm that greedily assigns true to variables. For the case of weighted 2-SAT, random instances can be solved by simply counting the number of "independent" positive clauses. (For weighted d-SAT with $d > 2$, the situation doesn't become any better.)

In search for a more interesting model, we notice that the all-zero assignment $\{0, \cdots, 0\}$ always satisfies an anti-monotone CNF formula (in the traditional sense). Motivated by this observation and the fact that the weighted anti-monotone 2-SAT is already fixed-parameter intractable (W[1]-complete), we consider a random model of weighted d-SAT in which clauses that contain positive literals only are forbidden. Random instances from such a model are always satisfied by the all-zero assignment in the traditional sense, and thus bear a (superficial) similarity to the model of standard satisfiability with hidden solutions studied in the literature. See, for example, (Jia, Moore, and Strain 2007). The difference is that in our case, there is no point to really "hide" the planted assignment — what we are looking for are assignments of weight k, i.e., assignments whose Hamming distance to the hidden solution is k. The problem is fixed-parameter intractable regardless of whether or not one knows the hidden solution.

We note that forbidding clauses that contain positive literals only does not make the weighted d-SAT problem trivial in the parameterized sense. In fact, based on Marx's result (Marx 2005) on a parameterized analog of Schaefer's Dichotomy theorem for parameterized Boolean constraint satisfaction problems, we have

Lemma 1. *Weighted d-SAT with a hidden solution is W[1]-complete for any $d \geq 2$ even if clauses containing positive literals only are forbidden.*

A formal definition of the random model for weighted d-SAT is in the following

Definition 2. *A random instance of the weighted d-SAT with a hidden solution is denoted by $\mathcal{F}_{k,d}^{n,p}$ where k is the parameter, n is the number of variables, and $p = p(n)$ defines the probability that a clause will be included in the CNF formula.*

The clauses of $\mathcal{F}_{k,d}^{n,p}$ are selected in the following way: each subset $\{x_{i_1}, \cdots, x_{i_d}\}$ of d variables contributes one clause to $\mathcal{F}_{k,d}^{n,p}$ with probability $p(n)$. There are $2^d - 1$ possible clauses on $\{x_{i_1}, \cdots, x_{i_d}\}$ (excluding the clause that contains positive literals only), and we pick one of them uniformly at random.

Note that the average number of clauses in $\mathcal{F}_{k,d}^{n,p}$ is $\binom{n}{d}p(n)$. In the literature, there are several slightly different random models for CNF formulas, including the one that picks a collection of clauses uniformly at random with (or without) replacements. For most of the properties studied previously, these models can be regarded as equivalent up to a "small" number of clauses so that one is free to choose a model that is easy to analyze.

For fixed-parameter problems, the equivalence of these models are not obvious, especially for the weighted 2-SAT. This is the chief reason we use the above definition.

We establish the exact threshold of the phase transition of $\mathcal{F}_{k,d}^{n,p}$ for any $d \geq 2$:

Theorem 1. *Let $p = \frac{c \log n}{n^{d-1}}$ with $c > 0$ being a constant and let $c^* = (2^d - 1)(d-1)!$. We have*

$$\lim_n \mathbb{P}\left[\mathcal{F}_{k,d}^{n,p} \text{ is satisfiable}\right] = \begin{cases} 0, & \text{if } c > c^*; \\ 1, & \text{if } c < c^*. \end{cases}$$

For the case of $d = 2$ and $d = 3$, the thresholds are respectively $c^* = 3$ and $c^* = 14$.

Proof. See Section 4. ■

For weighted 2-SAT which is already W[1]-complete, we provide another proof by exploring an interesting connection between the unsatisfiability of $\mathcal{F}_{k,2}^{n,p}$ and the existence of a Hamiltonian cycle in a properly-defined directed graph. The proof actually shows that with high probability, random instances of $\mathcal{F}_{k,2}^{n,p}$ in the unsatisfiable regime have a parametric resolution refutation of size $O(k)n^{O(1)}$ which can also be constructed in time $O(k)n^{O(1)}$. In other words, typical unsatisfiable instances from $\mathcal{F}_{k,2}^{n,p}$ are fixed-parameter tractable.

Theorem 2. *Consider a random instance from $\mathcal{F}_{k,2}^{n,p}$ where k is a fixed constant. Let $p = \frac{c(\log n + c_1 \log\log n)}{n}$. We have for any $c \geq 3$ and $c_1 > k - 1$, with high probability a parametric resolution proof of size $O(k)n^{O(1)}$ can be constructed in $O(k)n^{O(1)}$ time*

Proof. See Section 5. ■

The above result still holds even if the parameter k is in $\Theta(\log n)$. By way of contrast, a lower bound $n^{\Omega(k)}$ on the the size of a DPLL resolution refutation for random instances of $\mathcal{F}_{k,d}^{n,p}$ with $d \geq 3$ can be easily established, as shown in the following theorem 3. The difference between the resolution complexity of weighted 2-SAT $\mathcal{F}_{k,2}^{m,n}$ and weighted 3-SAT $\mathcal{F}_{k,3}^{m,n}$ is interesting. Note that unlike the situation in the standard satisfiability problem, weighted 2-SAT and weighted 3-SAT are both $W[1]$-complete.

Theorem 3. *For $p > c\frac{\log n}{n^2}$ with $c > 14$ and any fixed parameter $k > 0$, the size of the search tree of an oblivious DPLL algorithm for an instance of the random instances of $\mathcal{F}_{k,3}^{n,p}$ is $n^{\Omega(k)}$ with high probability.*

Proof. See Section 6. ■

Generalizations

As a generalization to weighted SAT, we can consider the following *weighted binary constraint satisfaction* problem (weighted-CSP) with domain size $d > 2$. Let D be the domain and $d_0 \in D$ be a fixed domain value. An instance of weighted CSP is a pair (C, k) where $k > 0$ is an integer and C is a CSP instance such that for any pair of variables (x_i, x_j), the tuple (d_0, d_0) is always compatible. Note that $\{d_0, \cdots, d_0\}$ is always a satisfying assignment. The question is to decide whether C has a satisfying solution that differs from the solution $(d_0, \cdots, d_0)$ by k variables. We are not aware of any study on this type of CSPs. But in terms of the modelling aspect of CSPs, this may be useful in modelling tasks where one has a configuration with a set of default settings and wants to see if k of the default settings can be changed without violating any constraint.

Lemma 2. *Weighted CSP is W[1]-complete if the domain size is a fixed constant.*

A random model similar to $\mathcal{F}_{k,d}^{n,p}$ for weighted d-SAT can be considered: with probability $p = p(n)$, there is a constraint between a pair of variables. For each constraint, t pairs of domain values are selected as the no-goods from $D^2 \setminus \{(d_0, d_0)\}$, where $t > 0$ is the constraint tightness.

The expected number of satisfying solutions to the random weighted binary CSP is

$$\binom{n}{k}(d-1)^k(1 - p(n)\frac{t}{d^2-1})^{k(n-k)}$$

which tends to 0 if $p(n) = c\frac{\log n}{n}$ such that

$$c > \frac{t}{d^2-1},$$

giving an immediate upper bound on the threshold of the phase transition.

In the study of the phase transitions of random CSPs, much effort has been paid to designing random models to avoid flawed variables and constraints. See, e.g., (Gent et al. 2001; Gao and Culberson 2007). We note that in the context of fixed-parameter complexity, the flawed variable is not a big issue for the weighted CSP problem we have defined in the above. This is because the solutions we are looking for are assignments that differ from the "hidden" solution $(d_0, \cdots, d_0)$ by k variables, while in the tradition CSPs a single flawed variable will result in the unsatisfiability.

Another way to generalize the weighted SAT problem is to consider the morphing of instances from weighted 2-SAT and the general weighted SAT. The former is W[1]-complete while the latter is W[2]-complete. The model is interesting since the structure of its instances bears many similarities to the structure of the CNF formulas that encode practical problems such as planning and model-checking. Initial results on the phase transitions and complexity of this model turn out to be interesting. Details can be found in (Gao 2008).

Definition 3. *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 problem parameter of the parameterized problem is k.

Relations to Backdoor Detections

There are certain relations between the weighted satisfiability problems discussed in this paper and the backdoor detection problem for the traditional satisfiability of the corresponding CNF formula. Consider the naive subsolver $\mathcal{A}$ that simply checks if the all-zero assignment is satisfying.

For the CNF formula defined in Definition 3, it is clear that a weight-k satisfying assignment for the parameterized problem specifies a weak-backdoor with respect to $\mathcal{A}$ for the standard satisfiability of the CNF formula.

For the weighted d-SAT instance where clauses containing only positive literals are forbidden, the relation is subtle — if an instance has no weight-k satisfying assignment but the formula is satisfiable in the traditional sense, then the formula has no size-k strong backdoor with respect to $\mathcal{A}$.

While more arguments are needed for these observations to materialize, we hope that studies along this line may help shed further light on the nature of the complexity of backdoor detection problems.

4. Proof of Theorem 1

Proof. Let $\mathcal{T}$ be the collection of subsets of d variables and let s be an assignment to the variables. We say that a subset of variables $T = \{x_1, \cdots, x_d\} \in \mathcal{T}$ is **s-good** if either

1. T doesn't contribute a clause to $\mathcal{F}_{d,k}^{n,p}$, or
2. the d-clause in $\mathcal{F}_{d,k}^{n,p}$ contributed by T is satisfied by the assignment s.

Let S be the set of assignments of weight k. Recall that the weight of an assignment is the number of variables that are assigned to true by the assignment. Consider an assignment $s \in S$ where the k variables $\{y_{i_1}, \cdots, y_{i_k}\}$ are assigned to true. From the definition of $\mathcal{F}_{d,k}^{n,p}$, the probability for a subset T of d variables to be s-good is

$$\mathbb{P}\left[T \text{ is s-good}\right] = \begin{cases} 1, \text{if } T \cap \{y_{i_1}, \cdots, y_{i_k}\} = \phi; \\ 1 - p(n)\frac{1}{2^d-1}, \text{otherwise}. \end{cases} \quad (1)$$

Let $\mathcal{T}_s \subset \mathcal{T}$ be the collection of subsets of d variables that have a nonempty intersection with $\{y_{i_1}, \cdots, y_{i_k}\}$. We have

$$|\mathcal{T}_s| = \sum_{j=1}^{d}\binom{n-k}{d-j}\binom{k}{j}.$$

Let X be the number of the assignments in S that satisfy $\mathcal{F}_{d,k}^{n,p}$. Then, we have

$$\begin{aligned} \mathbb{E}\left[X\right] &= \sum_{s\in S}\prod_{T\in\mathcal{T}}\mathbb{P}\left[T \text{ is s-good}\right] \\ &= \sum_{s\in S}\prod_{T\in\mathcal{T}_s}\mathbb{P}\left[T \text{ is s-good}\right] \\ &= \binom{n}{k}\left(1 - p(n)\frac{1}{2^d-1}\right)^{\sum_{j=1}^{k}\binom{n-k}{d-j}\binom{k}{j}}, \quad (2) \end{aligned}$$

which is asymptotically equivalent to

$$n^k e^{-\frac{kc\log n}{(2^d-1)(d-1)!}}$$

(Recall that $p(n) = c\frac{\log n}{n^{d-1}}$).

The upper bound on the threshold c^* follows from Markov's inequality. To lower bound the threshold c^*, we use Chebyshev's inequality

$$\mathbb{P}[X=0] \leq \frac{\mathbb{E}\left[X^2\right]}{(\mathbb{E}[X])^2} - 1.$$

Let D_i be the number of pairs of satisfying assignments of weight k that have i overlaps. We have

$$\mathbb{E}[D_i] = \sum_{s_1,s_2} \prod_{T:T\cap Y\neq\phi} \mathbb{P}[T \text{ is both } s_1\text{-good and } s_2\text{-good}],$$

where the sum is over all (ordered) pairs of weight-k assignments with i-overlaps. Write $\mathbb{E}\left[X^2\right]$ as

$$\mathbb{E}\left[X^2\right] = \sum_{i=1}^{k} \mathbb{E}[D_i]. \tag{3}$$

Let $\epsilon > 0$ be any number. We claim that for and $c = 1-\epsilon < \frac{1}{2^d-1}d-1!$, We claim that

$$\begin{cases} \mathbb{E}[D_i] \in o(n^{2k\epsilon - i\epsilon}) \text{ for } i > 1, \text{ and} \\ \lim\limits_{n} D_0 = (\mathbb{E}[X])^2 \end{cases} \tag{4}$$

Let s_1 and s_2 be two assignments with i overlaps. Without loss of generality, assume that the set of variables assigned to true by s_1 is $\{y_1, \cdots, y_{k-i}, y_{k-i+1} \cdots, y_k\}$ and the set of variables assigned to true by s_2 is $\{y_{k-i+1}, \cdots, y_k, y_{k+1}, \cdots, y_{2k-i}\}$. Let $Y = \{y_1, \cdots, y_k, \cdots, y_{2k-i}\}$. Consider a subset T of d variables. The probability that T is both s_1-good and s_2-good can be estimated as follows:

1. If $T \cap Y = \phi$, then $\mathbb{P}[T \text{ is good for } s_1 \text{ and} s_2] = 1$.
2. If $T \cap Y \neq \phi$, then $\mathbb{P}[T \text{ is good for } s_1 \text{ and } s_2]$ is at most
$$\left(1 - p(n)\frac{1}{2^d-1}\right).$$
To see this, note that in this case T has a non-empty intersection with either $\{y_1, \cdots, y_k\}$, or $\{y_{k-i+1}, \cdots, y_{2k-i}\}$, or both. Therefore, either
$$\mathbb{P}[T \text{ is } s_1\text{-good}] = \left(1 - p(n)\frac{1}{2^d-1}\right), \text{or}$$
$$\mathbb{P}[T \text{ is } s_2\text{-good}] = \left(1 - p(n)\frac{1}{2^d-1}\right).$$

Note that the total number of $T \in \mathcal{T}$ such that $T \cap Y \neq \phi$ is

$$\sum_{j=1}^{d} \binom{n-(2k-i)}{d-j}\binom{2k-i}{j}.$$

It follows that for $i > 1$,

$$\begin{aligned} \mathbb{E}[D_i] &= \binom{n}{k}\binom{n-k}{k-i} \prod_{T:T\cap Y\neq\phi} \left(1 - p(n)\frac{1}{2^d-1}\right) \\ &\sim n^{2k-i}\left(1 - p(n)\frac{1}{2^d-1}\right)^{\sum_{j=1}^{d}\binom{n-(2k-i)}{d-j}\binom{2k-i}{j}} \\ &\sim n^{2k-i} e^{-\frac{(2k-i)c\log n}{(2^d-1)(d-1)!}} \\ &\sim n^{2k\epsilon - i\epsilon}. \end{aligned} \tag{5}$$

For the case of $i = 0$, it can be shown that $\lim\limits_{n} D_0 = (\mathbb{E}[X])^2$. This proves the claim. The theorem follows from equations (2), (3), and (5). ∎

5. Proof of Theorem 2

Theorem 2 is established by exploiting an interesting connection between the parameterized resolution complexity of weighted 2-SAT and the existence of a Hamiltonian cycle in a properly defined directed graph. The following lemma establishes the connection.

Lemma 3. *Consider an instance $\mathcal{F}$ of the weighted 2-SAT problem with parameter k and consider an arbitrary ordering $\{x_1, \cdots, x_n\}$ of the variables. If $\mathcal{F}$ contains the following cycle of "forcing" clauses,*

$$\overline{x}_1 \vee x_2,\ \overline{x}_2 \vee x_3,\ \cdots, \overline{x}_{n-1} \vee x_n, \overline{x}_n \vee x_1,$$

then there is a parameterized resolution proof of size $O(kn)$ for $\mathcal{F}$. Furthermore, the parameterized version of the DPLL algorithm constructs such a resolution proof.

Proof. Recall that the parameterized resolution proof system has access to all the clauses that contain more than k negated variables or more than $n-k$ positive variables. For each $i \geq 1$, resolving

$$\overline{x}_i \vee x_{i+1},\ \cdots, \overline{x}_{i+k} \vee x_{i+k+1}$$

and $\overline{x}_{i+1} \vee \cdots \vee \overline{x}_{i+k+1}$ results in $\overline{x_i}$. These $\overline{x}_i$'s together with $x_1 \vee \cdots \vee x_{n-k+1}$ result in a contradiction. ∎

We emphasize that a cycle of forcing clauses, if exist, can be automatically exploited by DPLL-style algorithms. For the random model $\mathcal{F}_{2,k}^{n,p}$ of weighted 2-SAT, the existence of a cycle of "forcing" clauses can be shown due to a result of McDiarmid on the relation between the existence of a Hamiltonian cycle (or a long path) in a random graph and the existence of a directed Hamiltonian cycle (or a directed long path) in a "directed random graph".

Lemma 4. *(McDiarmid 1981) Let $0 < p < 1/2$ and $D(n,p)$ be a random directed graph obtained from the undirected graph $G(n,2p)$ by randomly directing the edges. Then,*

$$\begin{aligned} &\mathbb{P}[D(n,p) \text{ has a directed Hamiltonian cycle}] \\ &\geq \mathbb{P}[G(n,p) \text{ has a Hamiltonian cycle}]. \end{aligned}$$

We remark that the above inequality also holds for the existence of simple paths of certain length, while in (McDiarmid 1981) it is shown that for simple paths without length constraints, the above inequality becomes equality.

From Lemma 3, it is sufficient to show that there is a directed Hamiltonian cycle in the following directed graph $D_{\mathcal{F}}(n,p')$: Each vertex corresponds to a variable. There is an directed edge from x_i to x_j if and only if the 2-clause $\overline{x_i} \vee x_j$ is in the random formula $\mathcal{F}_{d,k}^{n,p}$. From the definition of $\mathcal{F}_{d,k}^{n,p}$, $D_{\mathcal{F}}(n,p')$ is the random directed graph $D(n,p')$ discussed in Lemma 4 with the edge probability $p' = \frac{1}{3}\frac{c\log n}{n}$. Since $c > 3$, the result follows from Lemma 4 and the threshold of the existence of a Hamiltonian cycle in the random graph $G(n,p)$ (see, for example, (Bollobas 2001)).

6. Proof of Theorem 3

Proof. We lower bound the number of nodes that an oblivious DPLL algorithm has to explore at depth $n^{\frac{1}{2}-\epsilon}$ where $\epsilon > 0$ is small constant. Associated with each node is a partial assignment to the variables on the path from the root to the node. Let v be a node and let $x_1, \cdots, x_l$ be the variables on the path to v in the search tree. We say v is *promising* if

1. exactly $\frac{k}{2}$ variables in $\{x_1, \cdots, x_l\}$ have been set to true;
2. no clause that contains at least two literals from $\{x_1, \cdots, x_l\}$ is in $\mathcal{F}_{k,3}^{n,p}$.

A promising node will be explored by an oblivious DPLL algorithm since the partial assignment to the variables on the path to the promising node neither falsify any clause nor produce a unit clause.

Let V be the set of nodes at depth $n^{\frac{1}{2}-\epsilon}$ such that for each $v \in V$, $\frac{k}{2}$ variables on the path from the root to v have been set to true. The total number of such nodes is

$$|V| = \binom{n^{\frac{1}{2}-\epsilon-1}}{\frac{k}{2}-1} + \binom{n^{\frac{1}{2}-\epsilon-1}}{\frac{k}{2}} = \binom{n^{\frac{1}{2}-\epsilon}}{\frac{k}{2}}$$

The probability that a node $v \in V$ is a promising node is at least

$$\left(1 - \frac{c \log n}{n^2}\right)^{\frac{1}{2}n^{1-2\epsilon}(n-n^{\frac{1}{2}-\epsilon})+n^{\frac{3}{2}\ 3\epsilon}}.$$

The probability that a node in V is not promising is thus asymptotically less that $n^{-2\epsilon}$. Using Markov's inequality, the probability that more than $n^{\frac{1-2\epsilon}{4}k-\epsilon}$ nodes in V are not promising is less than $\frac{1}{n^{\epsilon}}$. Therefore with high probability, the number of promising nodes is in $\Omega(n^{\frac{1-2\epsilon}{4}k})$. ■

7. Conclusions

We initiate an effort to extend the study of phase transitions and typical-case complexity to intractable parameterized problems which abound in many areas of AI research. While the focus of the current work is on the random models of representative parameterized hard problems and theoretical results on the threshold of the phase transitions and hardness of typical instances under these distributions, much work needs to be done to develop practical algorithms and solvers for intractable parameterized problems, to conduct empirical studies in proper context, and to further study the relation between the backdoor detection problem and the hardness of the parameterized version of the the same problem instance.

Acknowledgments

Work supported in part by NSERC Discovery Grant RGPIN 327587-06 and a UBCO startup grant.

References

Achlioptas, D.; Beame, P.; and Molloy, M. 2001. A sharp threshold in proof complexity. In *ACM Symposium on Theory of Computing*, 337–346.

Beame, P.; Karp, R.; Pitassi, T.; and Saks, M. 2002. The efficiency of resolution and Davis-Putnam procedures. *SIAM Journal on Computing* 31(4):1048–1075.

Bollobas, B. 2001. *Random Graphs*. Cambridge University Press.

Cheeseman, P.; Kanefsky, B.; and Taylor, W. 1991. Where the *really* hard problems are. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, 331–337. Morgan Kaufmann.

Dantchev, S.; Martin, B.; and Szeider, S. 2007. Parameterized proof complexity. In *Proceedings of The 48th Annual Symposium on Foundations of Computer Science (FOCS'07)*, 150–160. IEEE Press.

Dilkina, B.; Gomes, C.; and Sabharwal, A. 2007. Tradeoffs in the complexity of backdoor detection. In *Proceedings of the 13th International Conference on Principles and Practice of Constraint Programming (CP'07)*, 256–270.

Downey, R., and Fellows, M. 1999. *Parameterized Complexity*. Springer.

Gao, Y., and Culberson, J. 2007. Consistency and random constraint satisfaction models. *Journal of Artificial Intelligence Research* 28:517–557.

Gao, Y. 2008. Random instances of w[2]-complete problems: thresholds, complexity, and algorithms. In *Proceedings of 11th International Conference on Theory and Applications of Satisfiability Testing (SAT'08), to appear*.

Gent, I., and Walsh, T. 1998. Analysis of heuristics for number partitioning. *Computational Intelligence* 14(3):430–451.

Gent, I.; MacIntyre, E.; Prosser, P.; Smith, B.; and Walsh, T. 2001. Random constraint satisfaction: Flaws and structure. *Constraints* 6(4):345–372.

Gomes, C.; Fernandez, C.; Selman, B.; and Bessiere, C. 2005. Statistical regimes across constrainedness regions. *Constraints* 10(4):313–337.

Gottlob, G., and Szeider, S. 2007. Fixed-parameter algorithms for artificial intelligence, constraint satisfaction, and database problems. *The Computer Journal (to appear)*.

Jia, H.; Moore, C.; and Strain, D. 2007. Generating hard satisfiable formulas by hiding solutions deceptively. *Journal of Artificial Intelligence Research* 107–118.

Marx, D. 2005. Parameterized complexity of constraint satisfaction problems. *Computational Complexity* (2):153–183.

McDiarmid, C. 1981. General percolation and random graphs. *Adv. Appl. Prob* 13:40–60.

Neidermeier, R. 2006. *Invitation to Fixed-Parameter Algorithms*. Oxford University Press.

Nishimura, N.; Ragde, P.; and Szeider, S. 2004. Detecting backdoor sets with respect to horn and binary clauses. In *Proceedings of the Seventh International Conference on Theory and Applications of Satisfiability Testing (SAT'04)*.

Szeider, S. 2005. Backdoor sets for DLL subsolvers. *Journal of Automated Reasoning* (1 - 3):73–88.