Phase Transition In Random SAT Problems

Milena Vujošević–Janičić & Jelena Tomašević
Matematički fakultet, Univerzitet u Beogradu
Studentski trg 16, 11000 Beograd, Srbija
www.matf.bg.ac.yu/~milena, www.matf.bg.ac.yu/~jtomasevic

Abstract

In this paper, we give a brief overview of the phase transition in NP-complete problems, with a special attention given to SAT problem and its variants. We introduce a new, k-GD-SAT problem and provide experimental evidence for its phase transition. In addition, our experiments suggest that for each k, there is a linear relationship between phase transition points.

U ovom radu dajemo kratak pregled karakteristika fazne promene u NP-kompletnim problemima sa akcentom na SAT problem i njegove varijante. Prikazaćemo novi, k-GD-SAT model i eksperimentalne rezultate koji ukazuju na faznu promenu u ovom problemu. Dodatno, eksperimentalni rezultati za ovaj problem, za svaku vrednost k, ukazuju i na linearnu zavisnost izmedju tačaka fazne promene.

1 Introduction

For each NP-complete problem not all of its instances are equally hard. It is important to know which instances are the hardest ones. This can be used to predict behavior of an algorithm that solves the problem. Also, this can be used to make a relevant comparison of different algorithms for the same problem.

During the last decade of the twentieth century, it was discovered that in many NP-complete problems the hardest instances can be characterized by some simple syntactic rule [9]. First results of this sort addressed SAT (satisfiability) problem [16]. For instance, it was experimentally shown that the hardest instances for 3-SAT problem are the instances where the ratio of number of clauses and number of variables is close to 4.25 [8]. Similar results for other types of SAT problems and other NP-complete problems were gained in the following years, leading to a better understanding of the nature of NP-complete problems and their relationship to the problems from the class P.

The rest of the paper is organized as follows: we give the notion of the NP-complete problems (Section 2), and we discuss phase transition behavior in some of them (Section 3). We comment on SAT problem (Section 4) and about phase transition in different variants of SAT problem (Section 5). We introduce a new, k-GD-SAT problem and provide experimental results about its phase transition (Section 6). In Section 7 we draw final conclusions and discuss future work.

2 NP-Complete Problems

A decision problem is a problem that as an output requires an answer that is either YES or NO. If there is an algorithm that is able to produce the correct answer for any input of length n in at most $c \cdot n^k$ steps, where k and c are some constants independent of the input, then we say that the problem can be solved in polynomial time and we say that it belongs to the class P. Intuitively, the problems in the class P are those that can be solved efficiently.

Another important class of problems is NP, the class of non-deterministic polynomial problems. A problem is in this class if there is some algorithm that can guess the solution (while there are exponentially many potential solutions in the size of the input) and then can verify whether or not the guess is correct in polynomial time.

NP-complete problems make an important subset of the class NP. A decision problem C is NP-complete if it is in NP and it is NP-hard, i.e., every other problem in NP is *reducible* to it. Reducible here means that for every problem L from NP, there is a polynomial-time reduction, a deterministic algorithm which transforms instances l from L into instances c from C, such that the answer to c is YES if and only if the answer to l is YES. To prove that a NP problem A is NP-complete, it is sufficient to show that a problem already known to be NP-complete reduces to A. Note that the above definition and the claim do not guarantee there are any NP-complete problems. The first problem proved (by Cook, [4]) to be NP-complete is SAT problem (for details see Section 4). Since Cook's proof, thousands of other problems have been shown to be NP-complete.

The question whether or not the classes P and NP are equal is one of the most important open problems in theoretical computer science and mathematics. If one finds an efficient (polynomial-bounded) solution to any NP-complete problem, then every problem in NP can be solved efficiently and therefore must be in P, hence proving P=NP. Most theoretical computer scientists currently believe that this is not the case.

3 Phase Transition in NP-Complete Problems

A phase transition is a point where a system undergoes a sudden change of state. In many NP-complete problems, we are interested in the point where problem instances transit from the state of "most instances have solutions" to "most instances have no solution". In most cases there is a problem size independent constraint parameter which indicates how hard are the corresponding instances.

Phase transitions are typically shown by plotting the proportion of solvable problems against the constraint parameter. A characteristic of phase transitions is that as the problem size increases, the curves become sharper; that is, when plotted against a constraint parameter, the transition occurs in a shorter interval of this parameter.

When exploring phase transition, usually randomly generated instances of a problem are considered. These random instances are generated with respect to certain syntactical and probabilistic constraints.

Phase transition is a phenomenon observed in many NP-complete problems. Some of them are:

Number Partitioning Problem [9, 14] In this problem, n numbers from a given range are to be partitioned into l bags with the same sum. The special sort of this problem is partitioning n randomly chosen integers between 1 and 2^m into two subsets with sums S_1 and S_2 [3]. A partition is called *perfect* if $S_1 - S_2 = 0$ when the sum of all n integers in the original set is even, or $S_1 - S_2 = 1$ when the sum is odd. The decision problem is to determine whether a perfect partition exists. It was proved that when parameterizing the random problem in terms of $k = m/n$, the problem has a phase transition around $k = 1$.

Vertex Covering Problem This problem is one of the basic NP-complete problems [10]. Having an undirected graph $G = (V, E)$ with n vertices and e edges, a *vertex cover* is defined as a subset V_{vc} of vertices such that for all edges $(i, j) \in E$ there is at least one of its end points in V_{vc}. The decision problem is to determine whether a vertex cover of fixed cardinality X exists or not. It was proved that for random variant of this problem there is a phase transition in parameter X/n [19]. The position of phase transition depends on average degree of random generated graphs. If the probability of adding an edge to a graph is equal to c/n, then the expected average degree is equal to $c \cdot (n - 1)/2$. For example, if $c = 2$ then the problem has a phase transition around the value 0.4.

Graph Coloring Problem In this problem, a graph with n vertices and e edges is to be colored with m colors in such a way that connected vertices have different colors. The decision problem is to determine if a coloring for the fixed m exists. The phase transition was observed for the average degree used as the parameter. For example, in random 3-coloring graph problem [9] the phase transition has been observed around average degree of 4.6.

The Travelling Salesman Problem We consider a random TSP where n cities are uniformly distributed over a square with the side equal to a. The decision problem is to determine whether there is a tour of length less or equal to d which visits all n cities. For this problem, it is experimentally shown in [7] that the phase transition occurs at the point 0.75 for the parameter $d/(a\sqrt{n})$.

4 SAT Problem

Propositional satisfiability is the problem of determining, for a formula of the propositional calculus, if there is an assignment of truth values to its variables for which that formula evaluates to true. By SAT we mean the problem of propositional satisfiability for formulae in conjunctive normal form.

It was shown by Cook that SAT is NP-complete [4]. This was the first problem to be shown NP-complete, and still holds a central position in the study of computational complexity as the canonical NP-complete problem. The importance of the SAT problem is also grounded in practical applications, since many hard real-world problems (or their components) in areas such as AI planning, circuit satisfiability and software verification can be efficiently reformulated as instances of SAT. Also, many decision problems, such as graph coloring problems, planning problems, and scheduling problems can be rather easily encoded into SAT. Therefore, good SAT solvers are of great importance and significant research effort has been devoted in trying to find efficient SAT algorithms.

Due to a general belief that a polynomial time algorithm for SAT is not likely to be found, for now the only way to grade a solver is by its performance on the average, or in the worst case. In this context a phase transition in SAT problem is very important since experimental results show that most difficult problems for all SAT solvers are formulae in the region of phase transition.

There are two classes of algorithms for solving SAT: complete (they guarantee a correct answer for each input) and incomplete algorithms (they are usually much more efficient). Most of the modern complete algorithms are variants of DLL [5] algorithms such as Chaff. The second group includes stochastic local search algorithms, such as WalkSAT, genetic algorithms, survey propagation algorithms, etc.

5 Phase Transition For Different Random SAT Problems

Experimental results suggest that there is a phase transition in SAT problems between satisfiability and unsatisfiability as the ratio of the number of clauses and the number of variables is increased [16]. It is conjectured that, for different types of problem sets, there are values c_0 of L/N, which we call *phase transition points* such that:

$$\lim_{N\to\infty} s(N,[cN]) = \begin{cases} 1, & \text{for } c < c_0 \\ 0, & \text{for } c > c_0 \end{cases},$$

where s is a *satisfiability function* that maps sets of propositional formulae (of L clauses over N variables) into the segment $[0,1]$ and corresponds to a percentage of satisfiable formulae. Experimental results also suggest that in all SAT problems there is a typical easy-hard-easy pattern as the ratio L/N is increased, while the most difficult SAT formulae for all decision procedures are those in the crossover region.

A definition of random SAT problem includes information on the distribution of clause lengths and on distribution of literals within one clause. In *random k-SAT model*, for given values N and L, an instance of random k-SAT formula is produced by randomly generating L clauses of length k. Each clause is produced by randomly choosing k distinct variables from the set of N available variables, and negating each with probability 0.5 [16]. In *random mixed* SAT [8], each clause is generated as in random k-SAT except that the length of clauses is chosen randomly according to a finite probability distribution ϕ on integers. In $2+p$-SAT model [1], a formula with L clauses has (approximately) $(1-p)L$ clauses of the length 2 and pL clauses of the length 3 ($0 \le p \le 1$). Hence, this model smoothly interpolates between 2-SAT and 3-SAT model. In *constant probability model* [12], given N variables and L clauses, each clause is generated so that it contains each of $2N$ different literals with probability p.

For random k-SAT, estimates for phase transition points are $c_2 = 1$, $c_3 \approx 4.267$, $c_4 \approx 9.931$, $c_5 \approx 21.117$, $c_6 \approx 43.37$, $c_7 \approx 87.79$ [15] (c_k denotes a crossover point for k-SAT). In [2], there are rigorous bounds for c_k given: $2^k \ln 2 - k \le c_k \le 2^k \ln 2$

6 Random k-GD-SAT Model

We consider a family of random SAT problems that we introduced in [18], based on geometric distribution of clause lengths, denoted by k-GD-SAT. This model is a generalization of gd-SAT model [13]. In this model, generating of clauses over the set of N variables, for the probability parameter p ($0 < p \leq 1$), is specified by the stochastic context-free grammar[1] given in Table 1. Clauses are generated independently of each other.

Rule	Prob.
$\langle clause \rangle := \langle clause \rangle \vee \langle literal \rangle$	$1-p$
$\langle clause \rangle := \underbrace{\langle literal \rangle \vee \ldots \vee \langle literal \rangle}_{k}$	p
$\langle literal \rangle := \langle variable \rangle \mid \neg \langle variable \rangle$	0.5
$\langle variable \rangle := v_1 \mid v_2 \mid \ \ldots \ \mid v_N$	$1/N$

Table 1: Stochastic grammar for generating k-GD-SAT clauses

By the given stochastic grammar, for k-GD-SAT, only clauses of length equal or greater than k can be generated. Lengths of clauses in k-GD-SAT model have a geometric distribution; the probability of a clause of length l is $p(1-p)^{l-k}$, for $l \geq k$, and is equal to 0, for $l < k$. According to the properties of geometric distribution, the most probable clause length in k-GD-SAT model is k (with the probability p), while the expected clause length can be shown to be equal to $k - 1 + 1/p$. For $p = 1$, k-GD-SAT model becomes random k-SAT model. For $p = 1$, 2-GD-SAT model becomes 2-SAT model and, hence, it belongs to the class P. For any fixed p such that $p < 1$, k-GD-SAT is NP-complete. As p decreases, 2-GD-SAT problems smoothly interpolate between 2-SAT and NP-complete 2-GD-SAT problems. This makes k-GD-SAT model convenient for exploring a computational cost for directly linked P and NP-complete problems.

Our experiments show there is a phase transition between satisfiability and unsatisfiability in k-GD-SAT model, for a range of values of k and of the probability parameter p. All experimental results[2] presented in this paper were obtained by using zChaff SAT solver [17]. Figure 1 shows satisfiability function and computational cost for random k-GD-SAT, for $k = 2$, $p = 1/2$, and for $N = 25, 50, 75, 100$. Fraction of satisfiable formulae is shown against parameter L/N; for each N, in each L/N point, there were 1000 formulae randomly generated. These results confirm the general experience with other NP-complete problems that the hardest problem instances are in region of the phase transition.

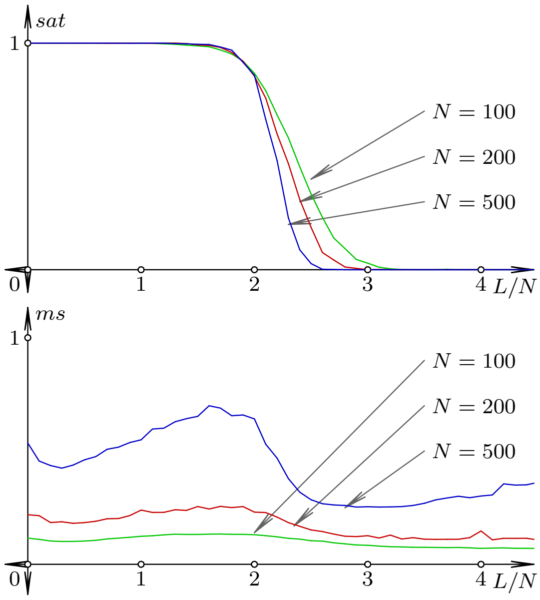

Figure 1: Satisfiability function (top) and computational cost (bottom) for random k-GD-SAT, for $k = 2$, for $p = 1/2$, and for $N = 100, 200, 500$.

We are also interested in a relationship between crossover points for different values of p (for a fixed k). For that purpose, for each p, we consider 50% satisfiability points, i.e., L/N values for which there are 50% satisfiable formulae. For different types of k-GD-SAT problem (for different values of k, p and N), we experimentally approximate these 50% satisfiability points. Since the phase transition region narrows as N grows, these 50% satisfiability points converge to the crossover point.

We performed a series of experiments and obtained experimental approximation for 50% satisfiability points for different values of k, p, and N.

[1] A stochastic context-free grammar is a context-free grammar with a stochastic component that attaches a probability to each of the production rules and controls its use.

[2] All experimental data and used programs are available upon request from the first author.

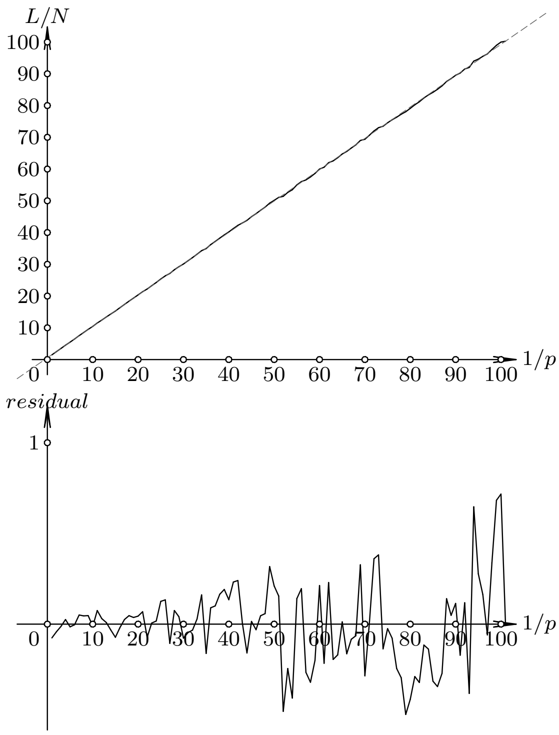

Figure 2: 50% satisfiability points for k-GD-SAT, for $N = 50$, $k = 2$, $1/p$ ranging from 1 to 101 and the line $y = 0.987281x + 0.558788$ that is the least square fit (top); Residuals from the best square fit line (bottom).

Figure 2 (top) shows 50% satisfiability points for k-GD-SAT, for $k = 2$, $N = 50$, $1/p$ ranging from 1 to 101 by step 1. For each point on the curve 1000 random formulae were generated. We determined a line that is the least square fit (i.e., a line for which the sum of squares of residuals is the least possible) and we measured residuals for all points and for the fit given by this line. The residuals are shown in Figure 2 (bottom). Although there is a noise in these results due to a relatively small sets of formulae used in experiments, all residuals are much less then 1. Similar results were obtained for other values of k and N (see Figure 3). These results provide evidence that crossover curves for k-GD-SAT model are (asymptotically) linear in parameter $1/p$.

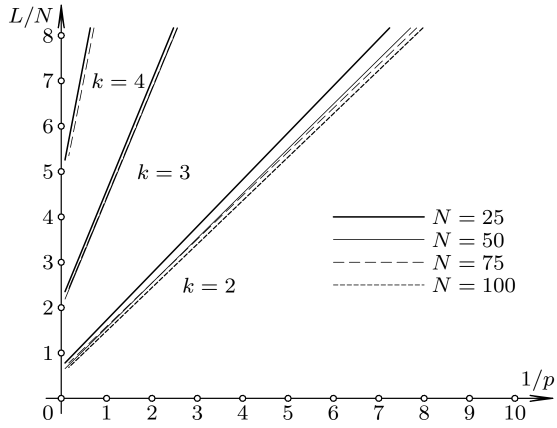

Figure 3: Experimentally approximated 50% satisfiability points for $N = 25, 50, 75, 100$, based on values for $1/p = 10$ and $1/p = 50$, and for $k = 2, 3, 4$.

7 Conclusions and Further Work

In this paper, we gave a brief overview of the phase transition in NP-complete problems, with a focus on SAT problem. We introduced a new random SAT model — k-GD-SAT, based on probability parameter p, that controls geometrical distribution on clause lengths. We provided experimental evidence about the phase transition for this model. Our experiments also gave a surprising outcome, suggesting that for each k, there is a linear relationship between crossover points (in parameter $1/p$).

In our further work, we are planning to look for theoretical confirmation and explanation of our experimental results. We hope that these results could shed some new light on understanding the nature of NP-complete problems.

References

[1] D. Achlioptas, L. M. Kirousis, E. Kranakis, and D. Krizanc. Rigorous results for random $(2+p)$-SAT. In Proceedings of RALCOM' 97, 1997.

[2] Dimitris Achlioptas, Assaf Naor, and Yuval Peres. Rigorous location of phase transitions in hard optimization problems. *Nature*, 435:759–764, 2005.

[3] Christian Borgs, Jennifer Chayes and Boris Pittel. Phase transition and finite-size scaling for the integer paritioning problem. Technical Report MSR-TR-2000-68

[4] Stephen A. Cook. The complexity of theorem-proving procedures. In STOC 71: Proceedings of the third annual ACM symposium on Theory of computing, pages 151158. ACM Press, 1971.

[5] Martin Davis, George Logemann, and Donald Loveland. A machine program for theorem-proving. *Communications of the ACM*, 5(7):394–397, 1962.

[6] E. Friedgut. Sharp threshold for graph properties and the k-sat problem. *Journal of the American Mathematical Society*, 12:1017–1054, 1999.

[7] Gent I.P., Walsh T. The TSP phase transition *Artificial Intelligence*, Volume 88(1), pp. 349-358(10), 1996.

[8] Ian P. Gent and Toby Walsh. The SAT phase transition. In *Proceedings of ECAI-94*, pages 105–109, 1994.

[9] I. P. Gent, E. Macintyre., P. Prosser, and T. Walsh. The constraidness of search. In *Proceedings of AAAI-96*, pages 246–252, Menlo Park, AAAI Press/MIT Press., 1996.

[10] M. R. Garey and D. S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness (W. H. Freeman, New York, 1997).

[11] A. Goerdt. A treshold for unsatisfiability. *17th International Symposium on Mathematical Foundations of Computer Science*, 1992.

[12] J. N. Hooker and C. Fedjki. Branch-and-cut soultion of inference problems in propositional logic. *Ann. Math. Artif. Intell.*, 1:123–139, 1990.

[13] Predrag Janičić. GD-sat model and crossover line. Journal of Experimental and Theoretical Artificial Intelligence, 13(3):181198, 2001.

[14] Stephan Mertens. Phase Transition in the Number Partitioning Problem. Physical review letters, volume 81 number 20, 1998.

[15] Stephan Mertens, Marc Mézard, and Riccardo Zecchina. Threshold values of random k-sat from the cavity method. Technical Report CoRR cs.CC/0309020, 2003.

[16] G. David Mitchell, Bart Selman, and J. Hector Levesque. Hard and easy distributions of sat problems. In *Proceedings AAAI-92*, pages 459–465, San Jose, CA, 1992. AAAI Press/The MIT Press.

[17] Matthew W. Moskewicz, Conor F. Madigan, Ying Zhao, Lintao Zhang, and Sharad Malik. Chaff: engineering an efficient sat solver. In *DAC '01: Proceedings of the 38th conference on Design automation*, pages 530–535. ACM Press, 2001.

[18] Milena Vujošević-Janičić, Jelena Tomašević, and Predrag Janičić. Random k-gd-sat Model and its Phase Transition. Faculty of Mathematics, University of Belgrade (submitted).

[19] Martin Weigt and Alexander K. Hartmann. Number of Guards Needed by a Museum: A Phase Transition in Vertex Covering of Random Graphs. Physical review letters, volume 84, number 26, 2000.