In More is Different --- Fifty Years of Condensed Matter Physics. Essays in honor of Nobel laureate Philip W. Anderson, Princeton Series in Physics, edited by N. Ong and R. Bhatt, 2001, 331--339.

CONTENTS

CHAPTER 1

INSIGHTS FROM STATISTICAL PHYSICS INTO COMPUTATIONAL COMPLEXITY

SCOTT KIRKPATRICK[1] AND BART SELMAN[2]

[1]IBM, Thomas J. Watson Research Center, Yorktown Heights, NY 10598
[2]Computer Science Deptartment, Cornell University, Ithaca, NY 14850

ABSTRACT

"NP-complete" problems are at the core of many computational tasks of practical interest, which means that the cost of solution will grow exponentially with problem size in the worst case (provided P $\neq$ NP). Although heuristics can be quite effective on such problems in most cases, there is a growing appreciation that these problems contain phase transitions, and at the phase boundaries exponential complexity becomes the typical outcome, not just the worst case. Using methods from statistical physics, a much better understanding of such phase transition phenomena in computational problems has been obtained in recent years. We will review several key results in this area, thereby illustrating some of the deep connections between computer science and statistical physics. The seminal work by Fu and Anderson [13, 14] provided the initial impetus for much of this work.

1.1 INTRODUCTION

Many key computational tasks of practical interest have been shown to be computationally intractable. Such problems, found for example, in planning, scheduling, machine learning, hardware design, and computational biology, generally belong to the class of NP-complete problems. To solve an NP-complete problem, it is widely believed that the computational resource requirements grow exponentially with problem size, at least in the worst case. The typical case behavior of these

problems is often much more difficult to characterize, but is more relevant from a practical perspective.

Fu and Anderson [13, 14] first conjectured a deep connection between NP-complete problems and models studied in statistical physics. More recently, we have shown that NP-complete problems can exhibit phase transition phenomena, analogous to those in physical systems, with the hardest problem instances occurring at the phase boundary. However, the exact relationship between phase transition phenomena and computational properties has remained unclear. For example, phase transitions have also been observed in computationally "easy" problems (*i.e.*, ones that are not NP-complete).

In this paper, we will discuss recent results that give a precise characterization of the relationship between phase transition phenomena and typical case computational complexity. More specifically, we show that when the underlying computational task exhibits a continuous phase transition, resource requirements grow only polynomially with problem size, while a special type of discontinuous phase transition corresponds to an exponential growth in typical resource requirements, characteristic of truly hard problem instances.

These results illustrate the potential benefits of exploring the rich connections between computer science and statistical physics.

1.2 SATISFIABILITY AND HARD-PROBLEM INSTANCES

As our computational task, we will consider the Boolean satisfiability (SAT) problem, an archetypal NP-complete problem. In the SAT problem, one is given a formula in Boolean logic and the task is to determine whether the formula is satisfiable. We will consider Boolean formulae written in a special form, called conjunctive normal form. Each formula consists of a series of clauses conjoined together (logical "AND"), where each clause is a disjunction (logical "OR") of literals. A literal is a Boolean variable or its negation (logical "NOT"). In the k-SAT problem, each clause contains exactly k literals. An example of a formula consisting of three clauses and two Boolean variables, A and B, is ((A OR (NOT B)) AND ((NOT A) OR B) AND ((NOT A) OR (NOT B))). The formula is *satisfiable*, because, *e.g.*, the assignment A set to FALSE and B set to FALSE satisfies this formula. One method for checking whether a formula with N Boolean variables is satisfiable is to check for each of the 2^N truth assignments, whether there is one that satisfies all the clauses in the formula. If none is found, then the formula is *unsatisfiable*.

One might imagine that there are much more clever ways of determining whether a formula is satisfiable, for example, methods that don't search through the space of all possible truth assignments. However, one of the key results in computational complexity theory shows that it is highly unlikely that such a clever algorithm exists. More specifically, Cook [9] showed the k-SAT problem with $k \geq 3$ to be NP-complete. As a consequence, there does not exist any procedure

that does significantly better on all Boolean formulae than one that exhaustively checks all truth assignments (assuming P $\neq$ NP, a widely believed, but as of yet unproven, conjecture [7]). In other words, no matter how clever an algorithm one develops for Boolean satisfiability, there will be formulae on which the algorithm takes time exponential in the number of variables. Note that NP-competeness is a worst-case notion. In practice, one is often more interested in what happens in a "typical" case or "average" case scenario. We now turn our attention to the behavior of algorithms on such typical Boolean formulae.

Our initial interest in the satisfiability problem arose from early reports that many satisfiability problems are easily solvable. For example, Goldberg [19] describes a class of random SAT problems that are surprisingly easy for the Davis-Putnam satisfiability procedure, which is one of the most widely used complete algorithms for satisfiability testing [11]. Goldberg's work led to an extensive theoretical exploration of his particular random instance model. In Goldberg's model, each Boolean clause is generated by selecting literals with some fixed probability. This leads to clauses of varying length. A rigorous analysis, reported in a series of papers [15, 16, 33], has shown that in this model, the *average-case* complexity is polynomial for *almost all* choices of parameter settings. In other words, it is difficult to generate computationally hard problem instances. Note that this does not mean that hard instances do not exist; it is simply means that such instances are extremely rare, and, in fact, may never be observed in practice.

In [28, 25, 37], we show how by using a different model for generating random formulae, called the *fixed-clause-length* model, one can easily generate hard problem instances. Consider generating a random 3-SAT problem. Each clause is generated by randomly selecting three variables from among N variables; each of these variables is negated with probability 0.5. We generate a total of M clauses. We found that the key in generating computationally hard instances is the ratio between M and N. Fig. 1.1 shows the median cost of solving randomly generated instances at different ratios of variables to clauses. The data was obtained by running the Tableau method, which is a highly efficient implementation of the Davis-Putnam (DP) procedure [10]. We measure the solution cost in number of recursive calls to the DP procedure. This measure is proportional to the actual run time of the algorithm but is machine independent. We see that the cost peaks at around a ratio of 4.3 clauses per variable.[1] Our experimental data shows that at this point the cost of determining satisfiability grows exponentially with size of the formulae.

Figure 1.2 gives the fraction of formulae that are unsatisfiable as a function of the ratio of clauses to variables for randomly generated formulae with 50 variables. At low ratios, few clauses compared to the number of variables, almost all instances are satisfiable (*i.e.*, the unsatisfiable fraction is almost zero). At relatively high ratios of clauses to variables, in a sense too many constraints, almost all

[1] For large N, this ratio converges to around 4.25. [10, 25].

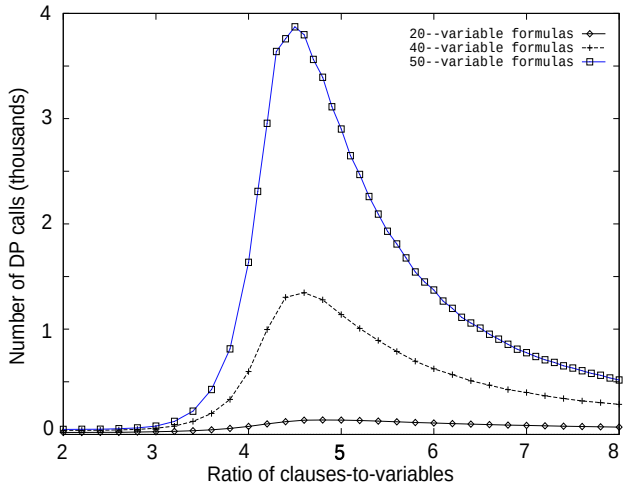

Figure 1.1: Solving 3SAT instances.

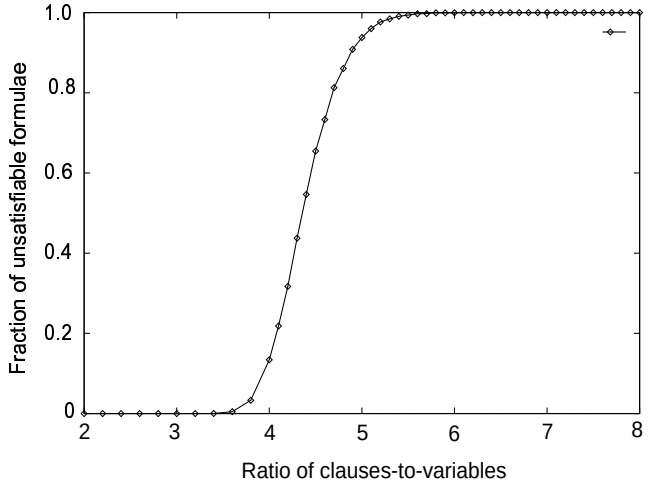

Figure 1.2: Fraction of unsatisfiable 3SAT problems.

randomly generated instances are unsatisfiable. A sudden change occurs around the critical ratio of 4.3. At this ratio, a "phase transition" from the mostly satisfiable phase to the mostly unsatisfiable phase takes place. From Figure 1.1, we see

that the phase-transition region coincides with the area with the hardest problem instances. In this region, the instances are again critically constrained. Below, we will take a closer look at what happens inside the phase transition region. A similar relation between exponential scaling and phase transition phenomena has been observed for a range of NP-complete problems. See for example, [6, 21].

The randomly generated critically constrained problem instances have been used extensively in the study and development of algorithms for graph coloring and satisfiability testing [36]. A key question is whether the results obtained for such instances are at all indicative of the behavior of the algorithms on more structured, real-world instances. The results of the DIMACS Challenge on Satisfiability Testing [38] suggest that the behavior of algorithms on hard random problems can indeed be representative of the behavior on more structured problems. The DIMACS Benchmark Problem Set contained several hard random instances and numerous more structured problems. The satisfiability algorithms fell in two categories: complete systematic procedures, and incomplete stochastic methods. These methods complement each other, in that there are problem classes where the stochastic methods are best, whereas on other problem classes the systematic methods are superior. However, within each category, algorithms that were fastest on the hard random instances usually also performed best on the more structured problems. Apparently, the hard random instances do exercise the various time critical parts of the algorithms. Therefore, the performance of algorithms on such hard random instances is a reasonably good indicator of the overall performance on a more diverse set of problem instances.

Aside from being useful as benchmark problems, there is also evidence that critically constrained problems may occur naturally in real-world applications. Nemhauser [32] studied a large airline scheduling problem, involving approximately 500 planes. The original schedule was obtained with a heuristic method and therefore only an approximation of the optimal solution. After a substantial computational effort, Nemhauser's group found a provably optimal solution. It was expected that such an optimal solution would lead to a savings of one or more planes over the heuristically obtained schedule. However, quite surprisingly, the optimal schedule did not save a single plane. The explanation appears to be that the problem had become critically constrained: Because of economic factors, the airline had assigned additional routes to planes that were idle during parts of the day in the original schedule. So, external factors can give rise to critically constrained real-world planning and scheduling problems. The ratio of constraints to variables will probably differ from the critical ratios found in hard random instances, because of the inherent internal structure of real-world problems. The reader is encouraged to consult any of the following additional references [21, 4, 8, 10, 18, 22, 27, 39, 41].

1.3 CONNECTIONS TO STATISTICAL PHYSICS

It has long been apparent that NP-complete problems have something in common with the models discussed in the literature of the statistical mechanics of disordered media. Loosely speaking, the optimization problems around which most NP-complete decision problems are formulated can be translated into the problem of finding ground states of appropriately constructed systems with many simple degrees of freedom. And the fact that "spin glasses," known to have very long relaxation times under Metropolis-style evolution at any low temperature, are found in many models of dynamical systems with quenched-in random interactions, has made establishing a connection between "glassiness" and the complexity defined by computational cost in computer science a natural objective. However, the problems defined in the two fields are not at all the same. The most significant difference is that complexity classes such as P and NP are defined in terms of the existence or nonexistence of at least one worst-case instance, while relaxation times and glassy behavior in statistical mechanics are characteristics of the most probable configurations of large systems, computed in practice as averages over partition functions (*i.e.*, over all initial conditions, or over all ways of defining the random system given some control parameters). This extra step of defining a probability measure over instances of a combinatoric problem in order to analyze average case complexity is one which until recently the computer science community was reluctant to take. Or, stated more concretely, relatively few of such average case compelxity results have been obtained in computer science. This is partly due to the difficulty of chosing the "right" underlying probability distribution for the ensemble.

Let's consider the simplest case in which to make the connection between a known NP-complete problem and a spin glass — partitioning a random graph with weighted edges into two equal sets of nodes, while minimizing the cost of the edges which cross the boundary between the two partitions. This is the example treated by Fu and Anderson [13]. Each node i can be in only on of two states, which we might call the "left" partition or the "right" partition. This can be represented by a spin S_i for each node, with $S_i = -1$ meaning that node i is in the "left" partition, and $S_i = 1$ meaning the "right" partition. If J_{ij} is the weight associated with each edge in the graph (many of these will be zero), the cost of a particular partition can be calculated as

$$H_1 = \sum_{i,j} J_{ij}(1 - S_i S_j)/2. \tag{1.1}$$

We need a second interaction term to enforce the constraint that the sizes of the two partitions are equal, or nearly so. Let

$$H_2 = \lambda \sum_{i,j} (S_i S_j), \tag{1.2}$$

and the Hamiltonian for the model equivalent to a particular graph bipartitioning problem is

$$H = H_1 + H_2. \tag{1.3}$$

For sufficiently large λ, the second term only allows configurations where the number of nodes in the "left" partition is equal to that in the "right" partition. The model which results is an infinite-range Ising antiferromagnet with random ferromagnetic interactions added (to keep edges connected by large weights in the same partitions). From the study of the SK models, the simplest spin glasses, this is known to give rise to a spin glass phase at low temperatures. We can define a similar Hamiltonian for the SAT problem.

We shall follow the notation used in the series of papers by Monasson and Zecchina. Since the variables in k-SAT are Boolean, we can use Ising spins to represent them. Let $S_i = +1$ if the variable x_i is set to true, and $S_i = -1$ if it is false. We will capture the set of M clauses in a formula in a single $M \times N$ matrix of random interactions, Δ, with $\Delta_{l,i} = +1$, if the l^{th} clause contains x_i, $\Delta_{l,i} = -1$, if the l^{th} clause contains $\neg x_i$ (here $\neg$ denotes logical negation), and otherwise $\Delta_{l,i} = 0$.

Then the number of clauses not satisfied provides a natural Hamiltonian for this problem, and is given by

$$E[\Delta, S] = \sum_{l=1,M} \delta \left[\sum_{i=1,N} \Delta_{l,i} S_i \; ; \; -k \right] \tag{1.4}$$

where $\delta[i;j]$ is the Kronecker symbol.

To fix the number of variables in each clause to be k we add the constraint

$$\sum_{i=1,N} \Delta_{l,i}^2 = k \quad \text{for all} \quad l = 1, ..., M \tag{1.5}$$

This can either be viewed as an Ising Hamiltonian with random external fields, or studied by expanding out the Kronecker delta, in which case we obtain interactions between up to k spins at a time. This gives a sense in which $k = 2$ (only pairwise interactions) is clearly distinct from $k \geq 3$.

1.4 A CLOSER LOOK AT THE PHASE TRANSITION

Figure 1.3 shows the phase transition for 3SAT for several different values of N (the number of variables). Note how the threshold function sharpens up for larger values of N. In [25], we show that the threshold has characteristics typical of phase transitions in the statistical mechanics of disordered materials.

"Finite-size scaling" analysis is a useful tool in the study of phase transition phenomena. This approach is based on rescaling the horizontal axis by a factor

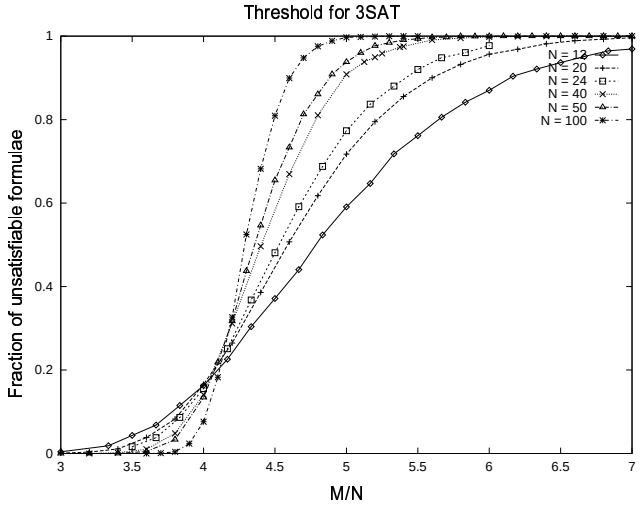

Figure 1.3: The 3SAT phase transition sharpens up.

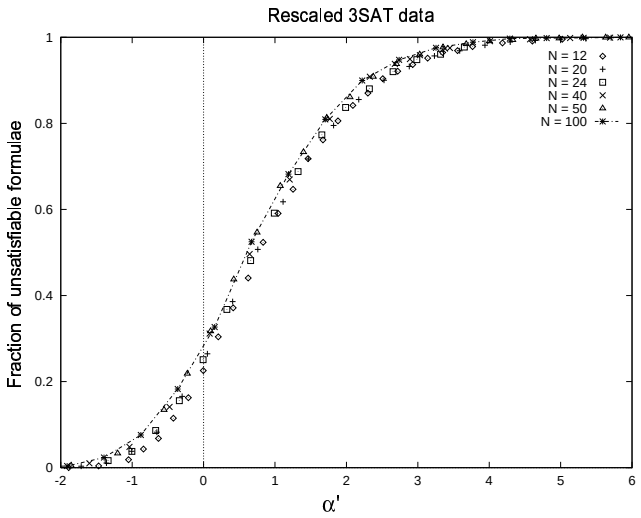

Figure 1.4: Rescaling the 3SAT phase transition.

that is a function of N. The function is such that the horizontal axis is stretched out for larger N. So, in effect, rescaling "opens up" the phase-transition for higher

values of N, and thus gives us a better look inside the transition. Figure 1.4 shows the result of rescaling the curves from Figure 1.3. Along the rescaled horizontal axis is $\alpha' = N^{1/\nu}(\alpha - \alpha_c)/\alpha_c)$, where $\alpha = M/N$, $\nu = 1.5$, $\alpha_c = 4.2$. We see that the original curves are rescaled into a single universal curve. Very recent theoretical work by Wilson [40] claims that the current experimental data has led to an underestimate of the critical exponent ν, since $\nu \to 1$ as k becomes large is predicted by "annealed" arguments as well as obtained in the replica-symmetric solution of Monasson and Zecchina, while Wilson's construction gives the critical regime a breadth always characterized by $\nu >= 2$. Wilson's results may require values of N that are not accessible experimentally. As a result, some questions remain open about the correct way to define large k, N limits in this problem.

Finite-size scaling can also be used to study properties other than satisfiability in the critical region. See [35] for a rescaling of computational cost curves, and [34] for a rescaling of the prime implicate function.

1.5 MIXTURES OF 2-SAT AND 3-SAT PROBLEMS

As noted earlier, the k-SAT problem is NP-complete for $k \geq 3$. For $k = 2$, there exists a linear time algorithm for determining the satisfiability of Boolean formulae [3]. This method cleverly avoids checking all possible truth assignments and works effectively (scaling linear in N) on all possible 2-SAT problems.

In order to understand what occurs between $k = 2$ and $k = 3$, we have studied [30, 2, 31] formulae containing mixtures of 2- and 3-clauses: consider a random formula with M clauses, of which $(1-p)M$ contain two literals and pM contain 3 literals, with $0 \leq p \leq 1$. This "$2+p$–SAT" model smoothly interpolates between 2–SAT ($p = 0$) and 3–SAT ($p = 1$). The problem is NP–complete, since any instance of the model for $p > 0$ contains a sub-formula of 3-clauses, but our interest here is in the complexity of "typical" problem instances.

We seek $\alpha_c(2+p)$, the threshold ratio M/N of the above model at fixed p. We know $\alpha_c(2) = 1$ and $\alpha_c(3) \simeq 4.25$. The formulae cannot be almost always satisfied if the number of 2–clauses (respectively 3–clauses) exceeds N (resp. $\alpha_c(3)N$). As a consequence, the critical ratio must be bounded by $\alpha_c(2+p) \leq \min\left(\frac{1}{1-p}, \frac{\alpha_c(3)}{p}\right)$.

The $2+p$–SAT model can be mapped onto a diluted spin glass model with N spins S_i: $S_i = 1$ if the Boolean variable x_i is true, $S_i = -1$ if x_i is false. And, again, with any configuration we associate an energy E, or cost-function, equal to the number of clauses violated. Random couplings between the spins are induced by the clauses. *The most important result of the replica approach [29] is the emergence, in the large N, M limit and at fixed p and α, of order parameters describing the statistics of optimal assignments, which minimize the number of violated clauses.* In this section, we give an overview of the results from statistical mechanics.

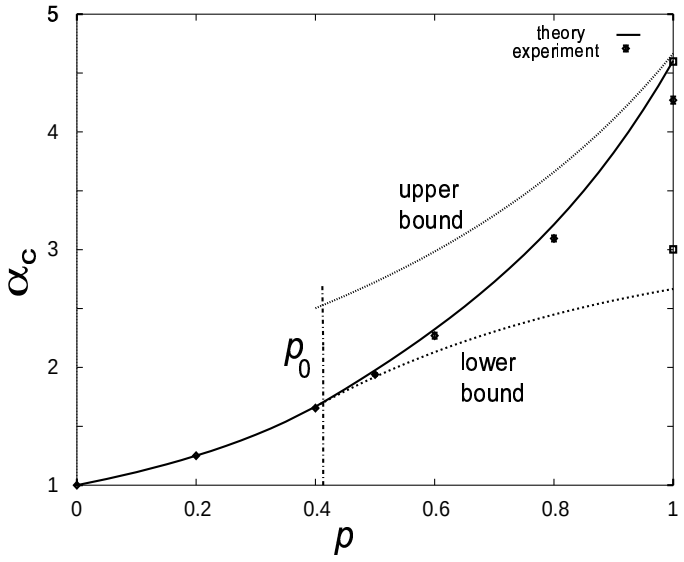

Figure 1.5: Theoretical and experimental results for the SAT/UNSAT transition in the 2+p-SAT model.

Consider an instance of the $2+p$-SAT problem. We use the $\mathcal{N}_{GS}$ ground state configurations to define

$$m_i = \frac{1}{\mathcal{N}_{GS}} \sum_{g=1}^{\mathcal{N}_{GS}} S_i^g \tag{1.6}$$

the average value of spin S_i over all optimal configurations. Clearly, m_i ranges from -1 to $+1$ and $m_i = -1$ (respectively $+1$) means that the corresponding Boolean variable x_i is always false (resp. true) in all ground states. The distribution $P(m)$ of all m_i gives the microscopic structure of the ground states. The accumulation of magnetizations m around ± 1 represents a "backbone" of almost completely constrained variables, whose logical values cannot vary from solution to solution, while the center of the distribution $P(m \simeq 0)$ describes weakly constrained variables. The threshold α_c will coincide with the appearance of an extensive backbone density of fully constrained variables x_i, with a finite probability weight at $m = \pm 1$. A simple argument shows that the backbone must vanish when $M/N = \alpha < \alpha_c$. Consider adding one clause to a SAT formula found below α_c. If there is a finite fraction of backbone spins, there will be a finite probability that the added clause creates an UNSAT formula, which cannot occur.

For $\alpha < \alpha_c$, the solution exhibits a simple symmetry property, usually referred to as Replica Symmetry (RS), which leads to an order parameter which

is precisely the magnetization distribution $P(m)$ defined above. An essential qualitative difference between 2-SAT and 3-SAT is the way the order parameter $P(m)$ changes at the threshold. This discrepancy can be seen in the fraction $f(k,\alpha)$ of Boolean variables which become fully constrained, at and above the threshold. As said above, $f(k,\alpha)$ is identically null below the threshold. For 2-SAT, $f(2,\alpha)$ becomes strictly positive above $\alpha_c = 1$ and is continuous at the transition : $f(2,1^-) = f(2,1^+) = 0$. On the contrary, $f(3,\alpha)$ displays a discontinuous change at the threshold : $f(3,\alpha_c^-) = 0$ and $f(3,\alpha_c^+) = f_c(3) > 0$.

While for the continuous transitions, the exact value of the threshold can be derived within the RS scheme, for the discontinuous case the RS prediction gives only upper bounds. The exact value of the threshold can be predicted only by a proper choice of the order parameter at the transition point, *i.e.*, by a more general symmetry breaking scheme, a problem which is still open. However, the predictions of the RS equations, such as the number of solutions, remain valid up to α_c, and the RS prediction for the nature of the threshold should be qualitatively correct.

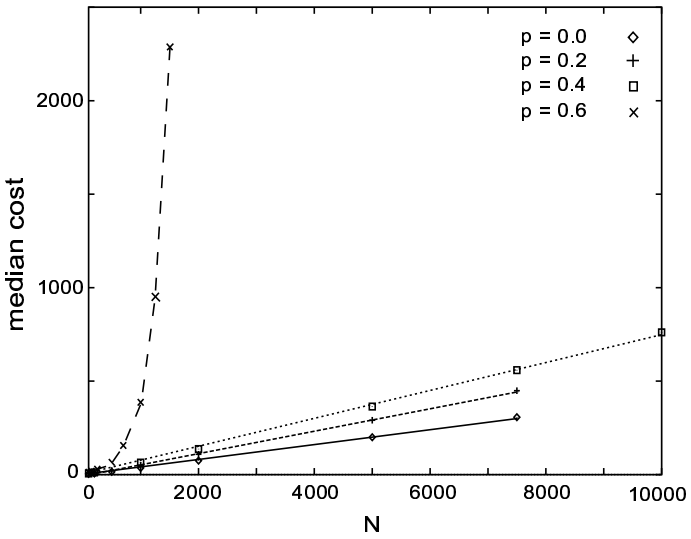

Figure 1.6: Median computational cost of proving a formula SAT or UNSAT using the Tableu procedure [10], for p ranging from 0 to 1.

For the mixed $2+p$–SAT model, the key issue is therefore to understand how a discontinuous 3–SAT-like transition may appear when increasing p from zero up to one and how it affects the computational cost of finding solutions near threshold. Applying the method of [29], we find for $p < p_0$ (according to various estimates, p_0 lies between 0.4 and 0.416), there is a **continuous** SAT/UNSAT

transition at $\alpha_c(2+p) = \frac{1}{1-p}$. This has recently been verified by rigorous analysis up to $p = 0.4$ [1]. The RS theory appears to be correct for $\alpha < \alpha_c(2+p)$, and thus gives both the critical ratio and the typical number of solutions, as in the $k = 2$ case. The SAT/UNSAT transition should coincide with a replica symmetry breaking transition, as discussed in [29]. So, for $p < p_0$, the model shares the characteristics of random 2–SAT.

For $p > p_0$, the transition becomes **discontinuous** and the RS transition gives an upper bound for the true $\alpha_c(2+p)$. The RS theory correctly predicts a discontinuous appearance of a finite fraction of fully constrained variables which jumps from 0 to f_c when crossing the threshold $\alpha_c(2+p)$. However, both values of $f_c(2+p)$ and α_c are slightly overestimated, *e.g.*, for $p = 1$, $\alpha_c^{RS}(3) \simeq 4.60$ and $f_c^{RS}(3) \simeq 0.6$ whereas experiments give $\alpha_c(3) \simeq 4.27$ and $f_c(3) \sim 0.5$. A replica symmetry breaking theory will be necessary to predict these quantities. For $p > p_0$, the random 2+p–SAT problem shares the characteristics of random 3–SAT.

Figure 1.5 shows our results for the location of the SAT/UNSAT transition for the 2+p-SAT model. The vertical line at p_0 separates the continuous from the discontinuous transition. The full line is the replica-symmetric theory's predicted transition, believed exact for $p < p_0$, and the diamond data points are results of computer experiment and finite-size scaling. The other two lines show upper and lower bounds obtained in [1], while the stronger upper bound due to [26], and the best known lower bound, due to [17], are indicated by square data points (at $p = 1.0$).

Figure 1.6 gives data on the scaling behavior of the Davis-Putnam style satisfiability procedure, Tableau [10], on the 2+P-SAT problem for a range of values of p. (For each value of p, we give the cost at the phase transition boundary.) The main observation to make is the linear scaling for $p < p_0$ and the exponential scaling for higher values of p. This is consistent with the analytical results that show that the behavior of the overall ensemble of clauses changes at a the critical ratio p_0, where it the system switches form the polynomial charateristics of 2-SAT to the exponential scaling for 3-SAT.

BIBLIOGRAPHY

[1] Achlioptas, D., Kirousis, L., Kranakis, E., and Krizanc, D. Rigorous results for random $(2+p)$–SAT. *Proc. RALCOM 97*, 1–10 (1997).

[2] Anderson, P.W. Solving problems in finite time. *Nature*, Vol. 400(8), 1999.

[3] Aspvall, B, Plass, M.F. and Tarjan, R.E. *Inf. Process. Lett.* 8, 121 (1979)

[4] Baker, A. Intelligent Backtracking On Constraint Satisfaction Problems: Experimental and Theoretical Results. Techn. Report. CIS-TR-95-08, Univ. of Oregon, Computer & Information Science, 1995.

[5] Buro, M. and Kleine-Büning, H. Report on a SAT competition. Techn. Rep. # 110, Dept. of Math. and Inform., University of Paderborn, Germany (1992).

[6] Cheeseman, P., Kanefsky, B., Taylor, and William M. Where the really hard problems are. *Proc. IJCAI-91* (1991) 331–336.

[7] Clay Mathematics Institute. Millennium Prize Problems. See www.ams.org/claymath/.

[8] Clearwater, S.H., Huberman, B.A., Hogg, T. Cooperative solution of constraint satisfaction problems. *Science*, 254 (1991).

[9] Cook, S.A. The complexity of theorem-proving procedures. *Proc. STOC-71* (1971) 151–158.

[10] Crawford, J.M. and Auton, L.D. Experimental results on the cross-over point in satisfiability problems. *AAAI-93* (1993) 21–27. (Ext. version in Artif. Intel.)

[11] Davis, M. and Putnam, H. A computing procedure for quantification theory. *J. of the ACM*, 7 (1960) 201–215.

[12] Dubois, O., and Boufkhad, Y. A general upper bound for the satisfiability threshold of random K–SAT formulas. *Journal of Algorithms 24*, 395-420 (1997)

[13] Fu, Y.-T., and Anderson, P. W. Applications of statistical mechanics to NP-complete problems in combinatorial optimization. *J. Phys.*, **A**19, 1605 (1986).

[14] Fu, Y.-T., and Anderson, P. W. In *Lectures in the Sciences of Complexity*, D. Stein (ed.), (Addison-Wesley, 1989), p. 815.

[15] Franco, J. and Paull, M. Probabilistic analysis of the Davis Putnam procedure for solving the satisfiability problem. *Discrete Applied Math.* 5 (1983) 77–87.

[16] Franco, J. Elimination of infrequent variables improves average case performance of satisfiability. *SIAM J. Comput.*, 20(6) (1991) 1119–1127.

[17] Frieze, A., and Suen, S. Analysis of two simple heuristics on a random instance of K–SAT. *Journal of Algorithms 20* (1996), 312-335.

[18] Gent, I.P. and Walsh, T. Computational phase transitions in real problems. Research Report, Dept. of AI, Edinburgh (1995).

[19] Goldberg, A. On the complexity of the satisfiability problem. Courant Comp. Sci. Rep., No. 16, New York University, NY, 1979.

[20] Hayes, B. Can't get no satisfaction. *American Scientist* Vol. 85(2), 108–112 (1996). http://www.amsci.org/amsci/issues/Comsci97/compsci9703.html

[21] Hogg, T., Huberman, B.A. & Williams, C. (eds.) Frontiers in problem solving: phase transitions and complexity. *Artificial Intelligence* Vol. 81 (I & II) (1996).

[22] Hogg, T. and Williams, C.P. Expected gains from parallelizing constraint solving for hard problems. *AAAI-94* (1994) 331–336.

[23] Kamath, A., Motwani, R., Palem, K., and Spirakis, P. Tail bounds for occupancy and the satisfiability threshold. *Random Structures and Algorithms* 7, 59 (1995)

[24] Kirkpatrick, S., Gelatt, C.D., and Vecchi, M.P. Optimization by simulated annealing. *Science*, 220 (1983) 671–680.

[25] Kirkpatrick, S. and Selman, B. Critical Behavior in the Satisfiability of Random Boolean Expressions. *Science*, 264 (May 1994) 1297–1301. Also p. 1249: "Mathematical Logic: Pinning Down a Treacherous Border in Logical Statements" by B. Cipra.

[26] Kirousis, L., Kranakis, E., and Krizanc, D. Approximating the unsatisfiability threshold of random formulas. *Proceedings of the 4th European Symposium on Algorithms*, (1992), 27-38.

[27] Larrabee, T. and Tsuji, Y. Evidence for a satisfiability threshold for random 3CNF formulas. *Proc. AAAI Symp. on AI and NP-hard problems*, Palto Alto, CA (1993).

[28] Mitchell, D., Selman, B., and Levesque, H.J. Hard and easy distributions of SAT problems. *Proc. AAAI-92*, San Jose, CA (1992) 459–465.

[29] Monasson, R., and Zecchina R. *Phys. Rev.* E 56, 1357 (1997).

[30] Monasson, R., and Zecchina R. Kirkpatrick, S., Selman, B., Troyansky, L. Phase Transition and Search Cost in the $2 + p$ SAT Problem. Proc. of *PhysComp96*, Toffoli, T., Biafore, M., Leão, J., eds., Boston (1996).

[31] Monasson, R., and Zecchina R., Kirkpatrick, S., Selman, B., Troyansky, L. Determining computational complexity from characteristic 'phase transitions'. *Nature*, Vol. 400(8), 1999, 133–137.

[32] Nemhauser, G., Barnhart, C., Hane, C., Johnson, E., Marsten, R., and Sigismondi, G. The fleet assignment problem. AI/OR Org. Workshop, Plymounth, VT (1994).

[33] Purdom, Jr. P.W., and Brown, C.A. Polynomial average-time satisfiability problems. *Inform. Sci.*, 41 (1987) 23–42.

[34] Schrag, R. and Crawford, J. Implicates and prime implicates in Random 3-SAT. *Artificial Intelligence*, 81 (1-2) 192–222 (1996).

[35] Selman, B. and Kirkpatrick, S. Critical behavior in the cost of K-satisfiability. *Artificial Intelligence* Vol. 81, 273–295 (1996).

[36] Selman, B., Levesque, H.J., and Mitchell, D. A new method for solving hard satisfiability problems. *Proc. AAAI-92*, San Jose, CA (1992) 440–446.

[37] Selman, B.; Mitchell, D.; and Levesque, H. Generating Hard Satisfiability Problems. *Artificial Intelligence*, Vol. 81, 1996, 17–29.

[38] Trick, M. and Johnson, D. (Eds.) *Proc. DIMACS Challenge on Satisfiability Testing.* Piscataway, NJ, 1993. (DIMACS Series on Discr. Math.)

[39] Williams, C.P. and Hogg, T. Using deep structure to locate hard problems. *Proc. AAAI-92*, San Jose, CA (1992) 472–277.

[40] Wilson, D.B. The empirical values of the critical k-SAT exponents are wrong. xxx.lanl.gov/abs/math.PR/0005136.

[41] Wolfram, S. Universality and complexity in cellular automata. *Physica D*, 10 (1984) 1–35.