The Hardest Random SAT Problems

Ian P. Gent
Department of Artificial Intelligence,
University of Edinburgh
80 South Bridge,
Edinburgh EH1 1HN, United Kingdom
I.P.Gent@edinburgh.ac.uk

Toby Walsh
IRST
Location Panté di Povo
I38100 Trento
Italy
walsh@irst.it

March 1994.

Keywords: Satisfiability, phase transitions, hard instances

Abstract:

We describe a detailed experimental investigation of the phase transition for several different classes of satisfiability problems including random k-SAT, the constant probability model, and encodings of k-colourability and the independent set problem. We show that the conventional picture of easy-hard-easy behaviour is inadequate. In each of the problem classes, although median problem difficulty shows an easy-hard-easy pattern, there is also a region of very variable problem difficulty. Within this region, we have found problems orders of magnitude harder than those in the middle of the phase transition. These extraordinary problems can easily dominate the mean problem difficulty. We report experimental evidence which strongly suggests that this behaviour is due to a "constraint gap", a region where the number of constraints on variables is minimal while simultaneously the depth of search required to solve problems is maximal. We also report results suggesting that better algorithms will be unable to eliminate this constraint gap and hence will continue to find very difficult problems in this region. Finally, we report an interesting correlation between these variable regions and a peak in the number of prime implicates. We predict that these extraordinarily hard problems will be of considerable use in analysing and comparing the performance of satisfiability algorithms.

1 Introduction

Many randomly generated NP-hard problems display a phase transition as some order parameter is varied, and as the problems go from being almost always soluble to being almost always insoluble [2]. This phase transition is often associated with problems which are *typically* hard to solve. In this paper, we show that with several different classes of satisfiability problems including random 3-SAT, the phase transition is indeed associated with problems which are typically hard *but* there are also regions in which problems are usually easy but sometimes extraordinarily hard. We postulate that this behaviour occurs when problems are "critically constrained". That is, when search must proceed to great depths because of the absence of easily observable constraints. We confirm this experimentally by demonstrating the existence of a "constraint gap" for the Davis Putnam procedure, the best known complete procedure for satisfiability. The constraint gap occurs in regions where most problems are satisfiable, and the ratio of constraint propagations to search branching reaches a sharp minimum, while the depth of search reaches a corresponding maximum. We predict that similar regions of very variable problem difficulty will be found with many other NP-hard problems besides satisfiability. The extraordinarily hard problems found in these regions may be of considerable use in analysing and comparing the performance of algorithms for NP-hard problems.

2 Satisfiability

Satisfiability (or SAT) is the problem of deciding if there is an assignment for the variables in a propositional formula that makes the formula true. We will consider SAT problems in conjunctive normal form (CNF); a formula, Σ is in CNF iff it is a conjunction of clauses, where a clause is a disjunction of literals, and a literal is a negated or un-negated variable. SAT is of considerable practical interest as many AI tasks like constraint satisfaction, diagnosis and planning can be encoded quite naturally in SAT. It is also of considerable theoretical interest as it is the archetypical NP-hard problem.

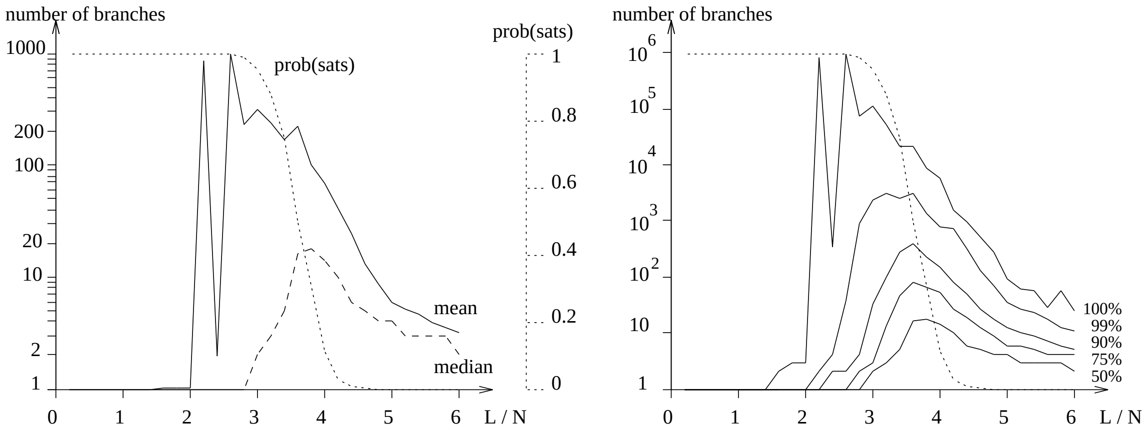
Figure 1). A standard procedure for determining satisfiability is due to Davis and Putnam [4] (see To simplify a set of clauses with respect to a partial truth assignment, we delete each clause that is satisfied by the partial truth assignment, and in every other clause delete any literals that contradict the partial truth assignment. Note that the Davis-Putnam procedure is non-deterministic since the literal used by the split rule is unspecified. As in previous studies (*eg.* [12, 6]), we will split upon the first literal in the first clause. We call this variant of the Davis-Putnam procedure "DP". Despite its simplicity, with efficient implementation and good heuristics for choosing literals to split on, the Davis-Putnam procedure is still the best complete procedure for satisfiability [5].

3 Constant Probability Model

In the constant probability model, given N variables, each of the 2N possible literals is included in a clause with probability p. Our experiments use a variant of the constant

procedure DP(Σ)
  if Σ empty **then return** satisfiable
  if Σ contains an empty clause **then return** unsatisfiable
  (*Tautology*) **if** Σ contains a tautologous clause c **then return** DP($\Sigma - \{c\}$)
  (*Unit propagation*) **if** Σ contains a unit clause c **then**
    return DP(Σ simplified by assigning truth value which satisfies c)
  (*Pure literal deletion*) **if** Σ contains a literal l but not the negation of l **then**
    return DP(Σ simplified by assigning truth value which satisfies l)
  (*Split*) **if** DP(Σ simplified by assigning a variable arbitrarily) is satisfiable
    then return satisfiable
    else return DP(Σ simplified by assigning variable opposite value)

Figure 1: The Davis-Putnam Procedure



(a) mean and median branches

(b) varying percentile branches

Figure 2: random CP problems tested using DP, N= 150. Note log scales

probability model proposed in [9] and since used in other experimental studies [12, 6]. In this problem class, empty and unit clauses are discarded and replaced by longer clauses since the inclusion of empty or unit clauses typically makes problems easier. We call this the "CP" model. In all our experiments, as in [12, 7], we choose p so that $2\mathrm{N}p = 3$ and the mean clause length remains approximately constant as N varies. In [7], Gent and Walsh show that the position of the phase transition occurs at fixed L/N when $2\mathrm{N}p$ is kept constant.

In Figure 2 (a) we plot the mean and median number of branches used by DP for the CP model at N = 150.[1] The number of branches is the number of leaf nodes in the search tree, and is an indication of problem difficulty. The dotted line indicates the observed probability that problems were satisfiable. Despite the log scale, there

[1] From L/N = 0.2 to 6 in intervals of 0.2 we tested 1000 problems at each point.

is a very considerable difference between mean and median performance. The worst case mean of 1,009 branches occurs at L/N = 2.6 in a mostly satisfiable region, whilst the worst case median of just 18 branches occurs at L/N = 3.8 in the middle of the phase transition. Problem difficulty in the mostly satisfiable region was very variable. Figure 2 (b) gives a breakdown in percentiles for the number of branches used from 50% (median) up to 100% (worst case). The worst case was 981,018 branches at L/N = 2.6, while at L/N = 3.8, the point of worst median performance, the worst case was just 8,982 branches, two orders of magnitude smaller. Comparison of Figure 2 (a) and (b) clearly shows that worst case behaviour is responsible for almost all the features seen in the mean in mostly satisfiable region.

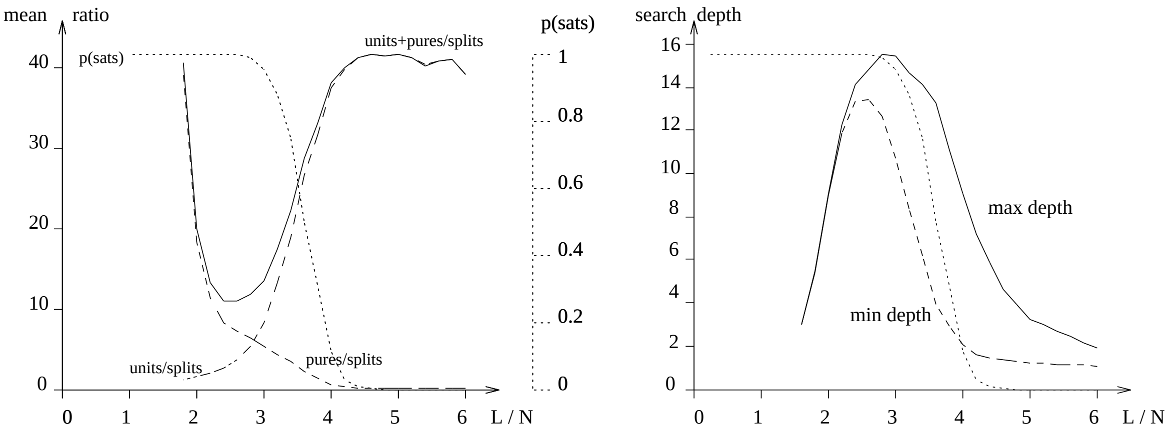
In [6], Gent and Walsh have shown that similar behaviour for CP is observed with better splitting heuristics, though variable and difficult behaviour is not apparent till larger N. In §5 we show that non-heuristic refinements to the Davis-Putnam procedure also appear unable to eliminate this behaviour. This suggests that the occurrence of extraordinarily hard problems in highly satisfiable regions is of great importance to the understanding of the hardness of satisfiability problems.

4 Critically Constrained Problems

One explanation for the extraordinary difficulty of some problems in the mostly satisfiable region is that these problems are *just* unsatisfiable or are satisfiable but give rise to subproblems which are *just* unsatisfiable. Such problems are "critically constrained".[2] That is, there are just enough constraints to make the problems unsatisfiable but no more than that. The Davis-Putnam procedure therefore has very little information about the best variable for splitting, and the best truth value to assign it. In addition, there is very little information for simplifying the resulting clauses. The problems are on a knife-edge between being satisfiable and unsatisfiable. As there is little information to suggest which it is, we must search through a large number of truth assignments to determine satisfiability.

The split rule is the only rule which gives rise to exponential behaviour. The other rules simplify the problem and do not branch the search. For instance, the unit and pure rules take advantage of constraints to commit to particular truth assignments. The poor performance of Davis-Putnam thus arises due to a large number of splits compared to unit propagations and pure literal deletions. We conjecture therefore that both the unit and pure rules will be of less importance in the mostly satisfiable region. In Figure 3 (a) we plot the mean ratio of pure literal deletions to splits, of unit propagations to splits and of the sum of pure literal deletions and unit propagations to splits for CP at N=150. Since the split rule is merely formalised guessing, the last of these ratios indicates the number of variable assignments that can be deduced for each guess during search. To avoid division by zero, we exclude the trivial problems which tend to occur at small L/N which are solved with no splits. Such problems can be solved in polynomial time using a simple preprocessing step which exhaustively applies the unit and pure rules. The

[2] We use the term critically constrained in a slightly different sense to that of [2].



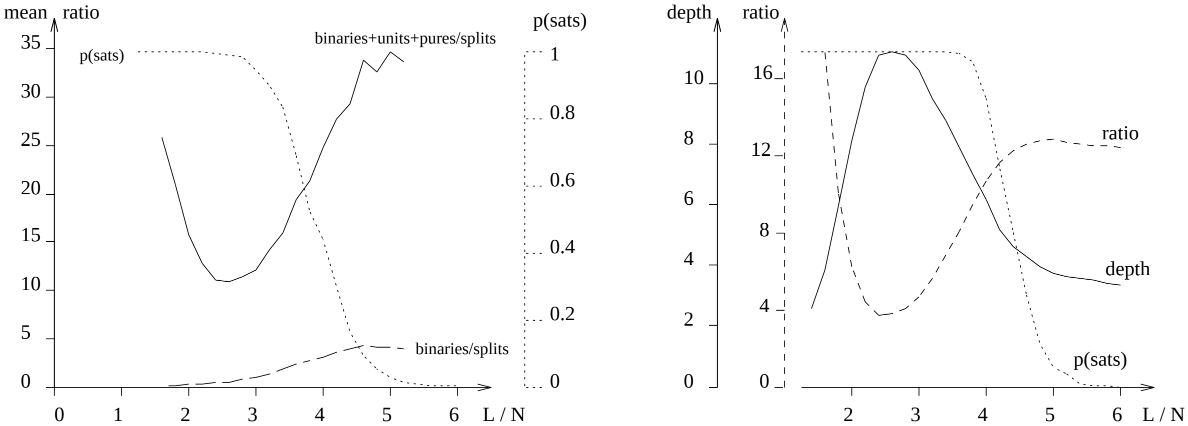
(a) Ratios of propagations to splits (b) Mean depth of search

Figure 3: random CP problems tested using DP, N= 150.

minimum in the mean ratio of the sum of units and pures to splits is 11.0 and occurs at L/N = 2.4, close to the region of most variable problem difficulty.

These graphs confirm that the unit and pure rules are not effective in the region of very variable problem difficulty. There appears to be a "constraint gap" in this region. That is, the unit and pure rules are often unable to identify any constraint on the truth assignments. We are thus forced to use the split rule extensively. This would suggest that the depth of search (*i.e.* the depth of nesting of split rule applications) would also peak in this region. In Figure 3 (b), we plot the mean minimum, and mean maximum depth of search. The peak of the minimum depth is 13.4 at L/N = 2.6, while the peak of maximum depth is 15.6 at L/N = 2.8. This coincides closely with the minimum in the ratio of the sum of units and pures to splits, and with the position of the variable region. For unsatisfiable problems, a peak in minimum search depth corresponds to an exponentially larger peak in problem difficulty, as all branches must be searched to at least the minimum depth of the tree. We confirmed this be plotting the logarithm of problem difficulty for unsatisfiable problems alone. This was approximately proportional to mean minimum search depth.

A very interesting question is whether the constraint gap occurs with incomplete procedures which can only solve *satisfiable* problems. One such procedure is GSAT [13]. Although we have investigated this point experimentally, we have as yet failed to find any strong evidence for variable behaviour or for a constraint gap in experiments, for instance, on CP at N = 150. Procedures like GSAT do not necessarily explore the whole search space, and so may avoid the exponential growth in search discussed above. Variable behaviour may, however, exist for such procedures but only at larger N.



(a) Constraint propagations, CP

(b) Constraint propagations, depth, 3-SAT

Figure 4: CP using DP+(Binary), and random 3-SAT using DP for N= 50

5 Binary Rule

There are other constraints which might be expected to narrow or even remove this constraint gap. For instance, one of the major features of the CP model which distinguish it from other problem classes like random k-SAT (see next section) is the variable length of clauses, and, in particular, the large numbers of binary clauses. Since there exists a linear time algorithm for the satisfiability of binary clauses [1], we have augmented DP with the following rule:

(Binary) if the binary clauses of (Σ simplified with the literal l set to $True$) are unsatisfiable then set l to $False$.

This rule has a non-deterministic choice of literal; this may affect the number of pure, unit or binary rules applied but *not* the number of splits.

In Figure 4 (a) we plot the mean ratios of the number of applications of the binary rule to splits and of the sum of the unit, pure and binary rules to splits for CP at N=50. As in §3 we fix $2\mathrm{N}p = 3$. [3] The binary rule allows a significant number of unsatisfiable problems to be solved without search, and as before these are omitted from the figure, accounting for some noise at large L/N. The ratios of unit and pure rule propagations to splits are similar to those in Figure 3 (a). Note that there are comparatively few applications of the binary rule compared to splits[4] and, like the unit rule, the utility of the binary rule increases with L/N. Although the binary rule reduces search significantly (the peak mean number of branches goes down from approximately 6 to 1.31), it does not appear to be very effective in the variable region. This suggests that the binary rule will not eliminate the constraint gap nor variable behaviour. Indeed, we found tentative

[3] From L/N = 0.2 to 6 in intervals of 0.2 we tested 1000 problems at each point.

[4] This may be affected by our implementation, in which the binary rule is only applied if the computationally cheaper pure and unit rules fail.

evidence of variability. At L/N = 2.6, one problem needed 75 branches, almost 3 times more than the next worse case over the whole experiment. The minimum value for the ratio of all propagations to splits also occurred at L/N =2.6, and was 10.8. Given the change in problem size and procedure, it is surprising that this value is so close to the value of 11.0 observed in §4.

We also implemented a restricted version of the binary rule which just determines the satisfiability of the binary clauses and does not simplify on any of the literals. Although this restricted rule is less expensive, it appears to be of little use in reducing search; for CP at N=100, $2\mathrm{N}p = 3$, it closed at most 20% of branches at large L/N but less than 3% of branches in the region of the constraint gap. It had little affect on mean behaviour. It remains to be seen if other constraints (*eg.* those on Horn or near Horn clauses) can be used to overcome the constraint gap, but we see no reason to expect this to be possible.

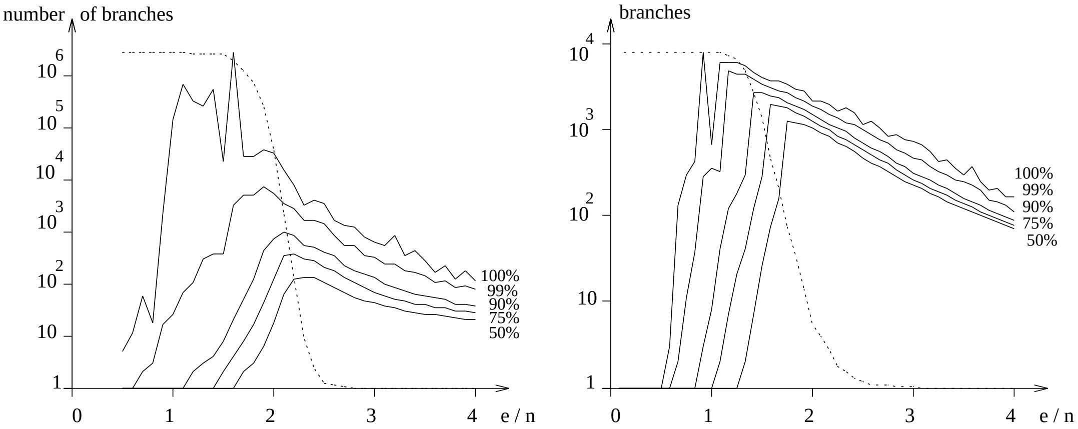
6 Random k-SAT

Although the constant probability model has been the focus of much theoretical study (see for example [10]), most recent experimental work has been on the phase transition of random 3-SAT [12, 3, 11]. A problem in random k-SAT consists of L clauses, each of which has exactly k literals chosen uniformly from the N possible variables, each literal being positive or negative with probability $\frac{1}{2}$.

We have observed a very similar constraint gap in 3-SAT to that seen for CP. In Figure 4 (b) we plot the mean ratio of propagations to splits and the mean minimum search depth for 3-SAT problems at N=50.[5] Again we omit problems solved by constraint propagation alone. The ratio of propagations to splits is very similar to Figure 3 (a), and reaches a minimum of 3.7 at L/N = 2.4. The graph of minimum search depth is very similar to Figure 3 (b), and reaches a maximum of 11.1 at L/N = 2.6. The graphs clearly illustrate the inverse relationship between search depth and constraint propagation, and the existence of a constraint gap away from the phase transition.

To date, we do not have conclusive evidence that the very variable behaviour described in §3 is found with random k-SAT. However, given the existence of a constraint gap, it is likely that the behaviour is present, but is just more difficult to observe than in CP. Crawford and Auton, for instance, have observed some problems in an otherwise easy region of random 3-SAT for L/N $\approx$ 2 that were as hard as the hard problems from the phase transition where L/N $\approx$ 4.3 [3]. To investigate this further, we compared 100,000 problems from 3-SAT at N = 50 and L/N set to 2 and 4.3 with a simplified version of the DP used in previous studies [12, 6] in which the pure rule is omitted. Remarkably, while problems at L/N=2 were typically very easy, one problem was nearly 60 times harder than the worst case at L/N=4.3. Yet this one problem was the only problem which needed more than 1000 branches to solve at L/N = 2, compared to 21 such problems at L/N = 4.3. Further details of the number of branches searched are given below.

[5] From L/N = 0.2 to 6 in intervals of 0.2 we tested 1000 problems at each point.



(a) percentile branches for 3COL (b) percentile branches for ISET

Figure 5: Encodings of two NP-hard problems into SAT

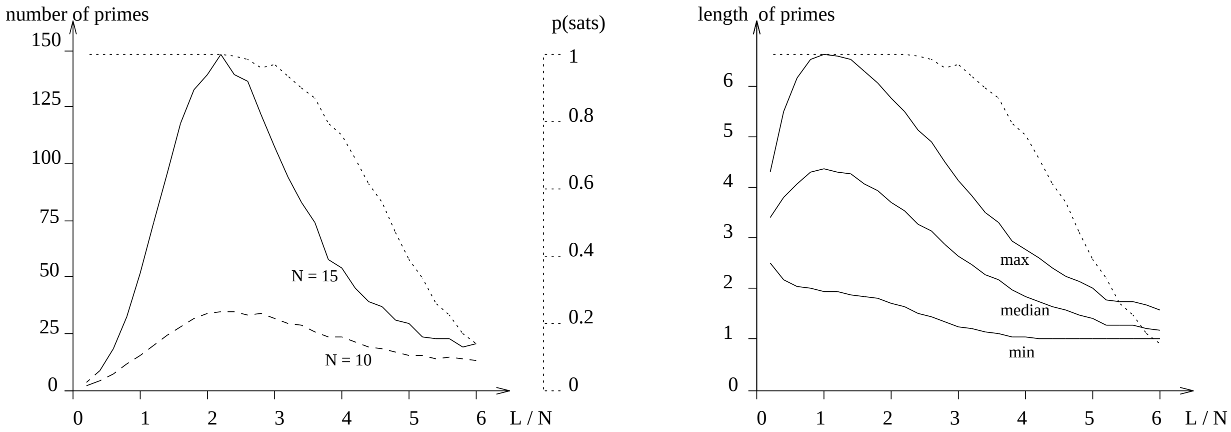
L/N	Prob(sats)	median	mean	s.d.	worst case
2.0	1	1	2.34	313	98,996
4.3	0.566	125	146	124	1,670

Although tentative, the evidence presented here of variable behaviour at L/N $\approx$ 2 is certainly consistent with the constraint gap observed in Figure 4 (b).

7 k-Colourability

Another way of randomly generating SAT problems is to map random problems from some other NP-hard problem into SAT. For example, the k-colourability (kCOL) of random graphs can be easily mapped into SAT. Given a graph, G the k-colourability problem is to assign one of k labels to each vertex of G so that adjacent vertices carry different labels. For a graph with n vertices and e edges, our encoding of kCOL into SAT uses $n.k$ variables. We generate random graphs to encode into SAT by choosing e edges from the $n.(n-1)/2$ possible uniformly at random. We use $\chi(n,e)$ to denote graphs drawn from this class.

In Figure 5 (a) we plot the breakdown in percentiles for the number of branches used by DP for encodings of 3-colourability for 1000 problems taken from $\chi(n,e)$ with $n=40$ and $e/n=0.5$ to 4 in steps of 0.1. The worst case was 2,905,011 branches at $e/n=1.6$, while at $e/n=2.4$, the point of worst median performance, the worst case was just 4,139 branches, 3 orders of magnitude smaller. As with the other random problem classes, median problem difficulty shows a simple easy-hard-easy pattern through the phase transition. Very similar behaviour for k-colourability was observed by Hogg and Williams using two special purpose colouring algorithms [8]. We again observed a constraint gap for this problem class, closely correlated with the variable region. For example, the maximum depth of search reached a peak at $e/n=1$.



(a) Mean number of primes N = 10, 15 (b) mean of min/median/max lengths, N = 15

Figure 6: Numbers and lengths of Prime Implicates in CP

8 Independent Set

Our final problem class is constructed by mapping the independent set problem (ISET) into SAT. Given an integer k and a graph (V, E), the independent set problem is to find a subset $S \subseteq V$ of size k such that all vertices of S are independent (not connected to each other). This is closely related to the clique problem since (V, E) has an independent set of size k iff $(V, \overline{E})$ has a clique of size k where $\overline{E}$ is the complement of E. For a graph with n edges, our encoding into SAT uses $n.k$ variables. As before, we use random graphs from $\chi(n, e)$.

In Figure 5 (a) we plot the breakdown in percentiles for the number of branches used by DP for encodings of the independent set problem for 1000 problems taken from $\chi(n, e)$ with $k = 6$, $n = 12$ and $e/n = \frac{1}{12}$ to 4 in steps of $\frac{1}{12}$. As before, the worst case performance is found in the region of typically underconstrained and satisfiable problems. The worst problem required 8,209 branches at 11 edges. By comparison, median problem difficulty shows a simple easy-hard-easy pattern through the phase transition. The peak median is 1,241 branches at 21 edges, where the worst case was 3,246 branches. We do not yet have conclusive evidence that the constraint gap occurs with this problem class as the ratio of all propagations to splits is within 10.6 ± 0.5 from 2 to 26 edges. Although the variable behaviour is not quite as dramatic as in our previous experiments, it does fit well the pattern identified in this paper. We conjecture therefore that variable behaviour will become more obvious with increasing n.

9 Prime Implicates

In an empirical study of the phase transition for random 3-SAT, Crawford and Auton found a secondary peak in problem difficulty in a region of high satisfiability [3]. Subsequently during a talk at AAAI-93, Crawford and Schrag observed that the number of prime implicates for random 3-SAT appears to peak in the same region, and suggested that the two phenomenon might be related.

A clause D is an *implicate* of a set of clauses C iff C implies D. D is a *prime* implicate iff it is an implicate and there is no other implicate E of C such that E implies D. Since an unsatisfiable set of clauses has a single prime implicate, the empty clause, the number and length of the prime implicates is not of help in understanding the difficulty of unsatisfiable problems. In Figure 6 (a) and (b) we therefore plot the mean number of prime implicates and their mean minimum, median and maximum length for satisfiable problems generated by the CP model for N=10 and 15. As before, we fix $2\mathrm{N}p = 3$. Computational limits prevented us from using larger N.

The peak mean number of prime implicates occurs at L/N = 2.2 for N= 15, and at 2.4 at N= 10. This corresponds closely to the region of very variable problem difficulty seen in §3, the constraint gap identified in §4, and the maximum in the depth of search. We expected the length of the prime implicates to be related to the difficulty of SAT problems since a branch closes if and only if it contains the negations of all the literals of one of the prime implicates. The solution depth and problem difficulty should therefore depend on the length of the prime implicates. Figure 6 (b) suggests, however, that if there is a correlation between the length of prime implicates and problem difficulty then it is not as direct as that with the number of prime implicates.

10 Related Work

Hogg and Williams have observed extremely variable problem difficulty for graph colouring using both a backtracking algorithm based on the Berlaz heuristic and a heuristic repair algorithm [8]. They found that the hardest graph colouring problems were in an otherwise easy region of graphs of low connectivity. The median search cost, by comparison, shows the usual easy-hard-easy pattern through the phase transition.

In an empirical study of the phase transition for random 3-SAT and the CP model using the Davis-Putnam procedure, Mitchell *et al.* noted that the mean is influenced by a very small number of very large values [12]. Their study therefore concentrated solely on the median as they felt that "it appears to be a more informative statistic". Our results suggest that the *distribution* of values is, in fact, of considerable importance in understanding problem difficulty, and that the median alone provides a somewhat incomplete picture.

In another empirical study of random 3-SAT using a tableau based procedure, Crawford and Auton observed a secondary peak in mean problem difficulty in a region of high satisfiability [3]. However, they noted that this peak did not seem to occur with the Davis-Putnam procedure and speculated that it was probably an artifact of the branching heuristics used by their procedure. Subsequently, as mentioned before, Crawford and Schrag have suggested that this peak might be related to the number of prime implicates. Our results suggest that this secondary peak also occurs with the Davis-Putnam procedure, but that it requires larger problems and larger sample sizes to be demonstrated convincingly.

11 Conclusions

We have performed a detailed experimental investigation of the phase transition for four different classes of randomly generated satisfiability problems. With each problem class, the median problem difficulty displays an easy-hard-easy pattern with the hardest problems being associated with the phase transition. We have shown, however, that the "conventional" picture of easy-hard-easy behaviour is inadequate since the distribution of problem difficulties has several other important features. In particular, all the problem classes have a region of very variable problem difficulty where problems are typically underconstrained and satisfiable. Within this region, we have found problems orders of magnitude harder than problems in the middle of the phase transition.

We have presented evidence that these very hard problems arise because of a "constraint gap"; that is, in this region, there are few constraints on the assignment of truth values to variables, requiring us occasionally to search through exponentially many possible truth assignments. As a consequence, the depth of the search tree also peaks in this region. We have also shown that this gap cannot be eliminated even if we take advantage of extra constraints (*eg.* those on the binary clauses). Finally, we have suggested that the appearance of these very hard problems is related to the number of prime implicates which peaks in this region. Given the wide range of problem classes that exhibit this very variable and sometimes extraordinary hard problem difficulty, this behaviour should be of considerable importance for the analysis of algorithms for satisfiability. In addition, our connection of this variable behaviour with a constraint gap should help researchers identify the hardest regions of other randomly generated problems for SAT and other NP-hard problems.

Acknowledgements

This first author was supported by a SERC Postdoctoral Fellowship and the second by a HCM Postdoctoral fellowship. We thank Alan Bundy, Pierre Lescanne, Bob Constable, and the members of the Mathematical Reasoning Group at Edinburgh (supported by SERC grant GR/H 23610), the Eureca group at INRIA-Lorraine, and the Department of Computer Science at Cornell University, for their constructive comments and for rather a lot of CPU cycles.

References

[1] B. Aspvall, M.F. Plass, and R.E. Tarjan. A linear-time algorithm for testing the truth of certain quantified Boolean formulas. *Information Processing Letters*, 8:121–123, 1979.

[2] P. Cheeseman, B. Kanefsky, and W.M. Taylor. Where the really hard problems are. In *Proceedings of the 12th IJCAI*, pages 163–169. International Joint Conference on Artificial Intelligence, 1991.

[3] J.M. Crawford and L.D. Auton. Experimental results on the crossover point in satisfiability problems. In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, pages 21–27. AAAI Press/The MIT Press, 1993.

[4] M. Davis and H. Putnam. A computing procedure for quantification theory. *J. Association for Computing Machinery*, 7:201–215, 1960.

[5] O. Dubois, P. Andre, Y. Boufkhad, and J. Carlier. SAT versus UNSAT. In *Proceedings of the Second DIMACS Challenge*, 1993.

[6] Ian P. Gent and Toby Walsh. Easy problems are sometimes hard. Research Paper 642, Dept. of Artificial Intelligence, Edinburgh, June 27 1993.

[7] I.P. Gent and T. Walsh. The SAT phase transition. To be presented at ECAI-94.

[8] T. Hogg and C. Williams. The Hardest Constraint Problems: A Double Phase Transition. Technical report, Dynamics of Computation Group, Xerox Palo Alto Research Center, 1993. PARC Preprint.

[9] J. N. Hooker and C. Fedjki. Branch-and-cut solution of inference problems in propositional logic. *Annals of Mathematics and Artificial Intelligence*, 1:123–139, 1990.

[10] Paul Walton Purdom Jr. and Cynthia A. Brown. The pure literal rule and polynomial average time. *SIAM Journal of Computing*, 14(4):943–953, November 1985.

[11] T. Larrabee and Y. Tsuji. Evidence for a Satisfiability Threshold for Random 3CNF Formulas. Technical Report UCSC-CRL-92-42, Baskin Center for Computer Engineering and Information Sciences, University of California, Santa Cruz, 1992.

[12] David Mitchell, Bart Selman, and Hector Levesque. Hard and easy distributions of SAT problems. In *Proceedings, 10th National Conference on Artificial Intelligence.* AAAI Press/The MIT Press, July 12-16 1992.

[13] Bart Selman, Hector Levesque, and David Mitchell. A new method for solving hard satisfiability problems. In *Proceedings, 10th National Conference on Artificial Intelligence.* AAAI Press/The MIT Press, 1992.