An Empirical Analysis of Search in GSAT*

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
INRIA-Lorraine
615, rue du Jardin Botanique
54602 Villers-les-Nancy, France.
walsh@loria.fr

June 15, 1993

Abstract

We describe an extensive study of search in GSAT, an approximation procedure for propositional satisfiability. GSAT performs greedy hill-climbing on the number of satisfied clauses in a truth assignment. Our experiments provide a more complete picture of GSAT's search than previous informal accounts. We describe in detail the two phases of search: rapid hill-climbing followed by a long plateau search. We demonstrate that when applied to randomly generated 3-SAT problems, both the number of satisfied clauses and the branching rate scale linearly with N, the number of variables. Our results allow us to make detailed numerical conjectures about the length of the hill-climbing phase, the average gradient of this phase, and to conjecture that both the average score and average branching rate decay exponentially during plateau search. We end by showing how these results can be used to direct future theoretical analysis. This work provides a case study of how computer experiments can be used to improve understanding of the theoretical properties of algorithms.

*Submitted to *Journal of Artificial Intelligence Research.* Available as Research Paper 634 from the first author's department.

1 Introduction

Mathematicians are increasingly recognizing the usefulness of experiments with computers to help advance mathematical theory. Experiments can be used to refute conjectures. For example, Euler conjectured that no nth power could be written as the sum of fewer than n other nth powers. This was disproved by Lander and Parkin in 1957 using a computer search through 5th powers.[1] Experiments can also provide strong evidence to support conjectures. For instance, Sinisalo has used a computer to show that Goldbach's conjecture (that every even number greater than two is the sum of two primes) holds to at least 4×10^{11}. Experiments can also be used to suggest conjectures and generate new insights. Chaos theory and non-linear dynamics are two areas which have benefitted greatly from such experimentation.

It is surprising therefore that one area of mathematics which has benefitted little from empirical results is the theory of algorithms. Indeed, it is somewhat ironic as algorithms, the objects of this theory, are merely abstract descriptions of computer programs. Of course, we should in principle be able to reason about the properties of an algorithm entirely deductively. However, such theoretical analysis is often too complex for our current mathematical tools. Where theoretical analysis is practical, it is often limited to (unrealistically) simple cases. For example, results presented in [7] for the greedy algorithm for satisfiability do not apply to interesting and hard region of problems as described in §3. In addition, actual behaviour on real problems is sometimes quite different to worst and average case analyses. We therefore support the calls of McGeoch [9], Hooker [6] and others for the development of an empirical science of algorithms. In such a science, experiments as well as theory are used to advance our understanding of the properties of algorithms. One of the aims of this paper is to demonstrate the benefits of such an empirical approach. We will present some surprising experimental results and demonstrate how such results can direct future efforts for a theoretical analysis.

The algorithm studied in this paper is GSAT, a randomized hill-climbing procedure for propositional satisfiability (or SAT) [13, 12]. Propositional satisfiability is the problem of deciding if there is an assignment for the variables in a propositional formula that makes the formula true. Recently, there has been considerable interest in GSAT as it appears to be able to solve large and difficult satisfiability problems beyond the range of conventional procedures like Davis-Putnam [13]. We believe that the results we give here will actually apply to a larger family of procedures for satisfiability called GenSAT [3]. Understanding such procedures more fully is of considerable practical interest since SAT is, in many ways, the archetypical (and intractable) NP-hard problem. In addition, many AI problems can be encoded quite naturally in SAT (*eg.* constraint satisfaction, diagnosis and

[1] $27^5 + 84^5 + 110^5 + 133^5 = 144^5$ to be precise. More recently, Noam Elkies has used a Connection Machine to find a 4th power which is the sum of three other 4th powers. The reader is not advised to search for a 3rd power which is the sum of two other 3rd powers, but this footnote is too brief to explain why.

vision interpretation, refutational theorem proving, planning).

This paper is structured as follows. In §2 we introduce GSAT, the algorithm studied in the rest of the paper. In §3 we define and motivate the choice of problems used in our experiments. The experiments themselves are described in §4. These experiments provide a more complete picture of GSAT's search than previous informal accounts. The results of these experiments are analysed more closely in §5 using some powerful statistical tools. This analysis allow us to make various experimentally verifiable conjectures about GSAT's search. For example, we are able to conjecture: the length of GSAT's initial hill-climbing phase; the average gradient of this phase; the linear scaling of various important features like the score (on which hill-climbing is performed) and the branching rate. In §6 we show how such results can be used to direct future theoretical analysis. Finally, in §7 we describe related work and end with some brief conclusions in §8.

2 GSAT

GSAT is a random greedy hill-climbing procedure. GSAT deals with formulae 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. GSAT starts with a randomly generated truth assignment. and hill-climbs by flipping the variable assignment which gives the largest increase in the number of clauses satisfied (which we will call the "score" from now on). Given the choice between several equally good flips, GSAT picks one at random. If there exists no flip which increases the score, then a variable is flipped which does not change the score or (failing that) which decreases the score the least.

```
procedure GSAT($\Sigma$)
    for i := 1 to Max-tries
        T := random truth assignment
        for j := 1 to Max-flips
            if T satisfies $\Sigma$ then return T
            else Poss-flips := set of vars which increase satisfiability most
                 V := a random element of Poss-flips
                 T := T with V's truth assignment flipped
        end
    end
    return "no satisfying assignment found"
```

In [3] we describe a large number of experiments which suggest that neither greediness not randomness is important for the performance of this procedure. These experiments also suggest various other conjectures. For instance, for random 3-SAT problems (see §3) the log of the runtime appears to scale with a less than linear dependency on the problem size. Conjectures such as these could, as we noted in the introduction, be very profitably used to direct future efforts to analyse GSAT theoretically. Indeed, we believe that the experiments reported here suggest various conjectures which would be useful in a proof of the relationship between runtime and problem size (see §6 for more details)

3 Problem Space

To be able to perform experiments on an algorithm, you need a source of problems on which to run the algorithm. Ideally the problems should come from a probability distribution with some well-defined properties, contain a few simple parameters and be representative of problems which occur in real situations. Unfortunately, it is often difficult to meet all these criteria. In practice, one is usually forced to accept either problems from a well-defined distribution with a few simple parameters or a benchmark set of real problems, necessarily from some unknown distribution. In these experiments we adopt the former approach and use CNF formulae randomly generated according to the random k-SAT model.

Problems in random k-SAT with N variables and L clauses are generated as follows: a random subset of size k of the N variables is selected for each clause, and each variable is made positive or negative with probability $\frac{1}{2}$. For random 3-SAT, there is a phase transition from satisfiable to unsatisfiable when L is approximately 4.3N [11, 8, 2]. At lower L, most problems generated are under-constrained and are thus satisfiable; at higher L, most problems generated are over-constrained and are thus unsatisfiable. As with many NP-complete problems, problems in the phase transition are typically much more difficult to solve than problems away from the transition [1]. The region L=4.3N is thus generally considered to be a good source of hard SAT problems and has been the focus of much recent experimental effort.

4 GSAT's search

When GSAT was first introduced, it was noted that search in each try is divided into two phases. In the first phase of a try, each flip increases the score. However, this phase is relatively short and is followed by a second phase in which most flips do not increase the score, but are instead sideways moves which leave the same number of clauses satisfied. This phase is a search of a "plateau" for the occasional flip that can increase the score.[2] One of the aims of this paper is to improve upon such informal observations by making *quantitative* measurements of GSAT's search, and by using these measurements to make several experimentally testable predictions.

[2] Informal observations to this effect were made by Bart Selman during the presentation of [13] at AAAI-92. These observations were enlarged upon in [4].

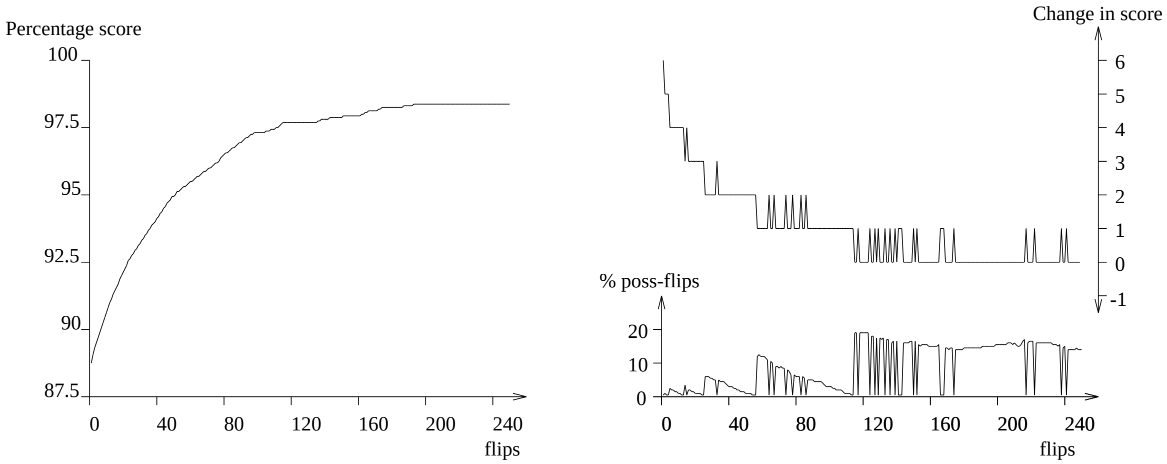
To achieve such aims, three points of methodology are essential. First, experiments should be performed with the largest problem size possible and as many times as possible. There may well be emergent properties at large problem sizes, whilst performing many experiments reduces variance. Second, a good view of the data must be sought. That is, we must look for features of performance which are meaningful and which are as predictable as possible: these features may not be the most immediately obvious to record. Third, data must be analysed, not simply measured. Suitable analysis of data may show features which are not clear from a simple (graphical) presentation. Invaluable discussion of all these research principles is contained in [9]. In the rest of this paper we show how these principles enable us to make very detailed numerical predictions about GSAT's search.

Many features of GSAT's search space can be graphically illustrated by plotting how they vary during a try. The most obvious feature to plot is the score, the number of satisfied clauses. In our quest for a good view of GSAT's search space, we also decided to plot "poss-flips" at each flip: that is, the number of equally good flips between which GSAT randomly picks. This is an interesting measure since it indicates the branching rate of GSAT's search space.

We begin with one try of GSAT on a 500 variable random 3-SAT problem in the difficult region of L/N = 4.3 (Figure 1a). Although there is considerable variation between tries, this graph illustrates features common to all tries. Both score (in Figure 1a) and poss-flips (in Figure 1b) are plotted as percentages of their maximal values, that is L and N respectively. The percentage score starts just above 87.5%, which might seem surprisingly high. Theoretically, however, we expect a random truth assignment in k-SAT to satisfy $\frac{2^k-1}{2^k}$ of all clauses (in this instance, $\frac{7}{8}$). As expected from the earlier informal description, the score climbs rapidly at first, and then flattens off as we mount the plateau. The graph is discrete since positive moves increase the score by a fixed amount, but some of this discreteness is lost due to the small scale. To illustrate the discreteness, in Figure 1b we plot the change in the number of satisfied clauses made by each flip (as its exact value, unscaled). Note that the x-axis for both plots in Figure 1b is the same.

The behaviour of poss-flips is considerably more complicated than that of the score. It is easiest first to consider poss-flips once on the plateau. The start of plateau search, after 115 flips, coincides with a very large increase in poss-flips, corresponding to a change from the region where a small number of flips can increase the score by 1 to a region where a large number of flips can be made which leave the score unchanged. Once on the plateau, there are several sharp dips in poss-flips. These correspond to flips where an increase by 1 in the score was effected, as can be seen from Figure 1b. It seems that if you can increase the score on the plateau, you only have a very small number of ways to do it. Also, the dominance of flips which make no change in score graphically illustrates the need for such "sideways" flips, a need that has been noted before [13, 3].

Perhaps the most fascinating feature is the initial behaviour of poss-flips. There



(a) Score

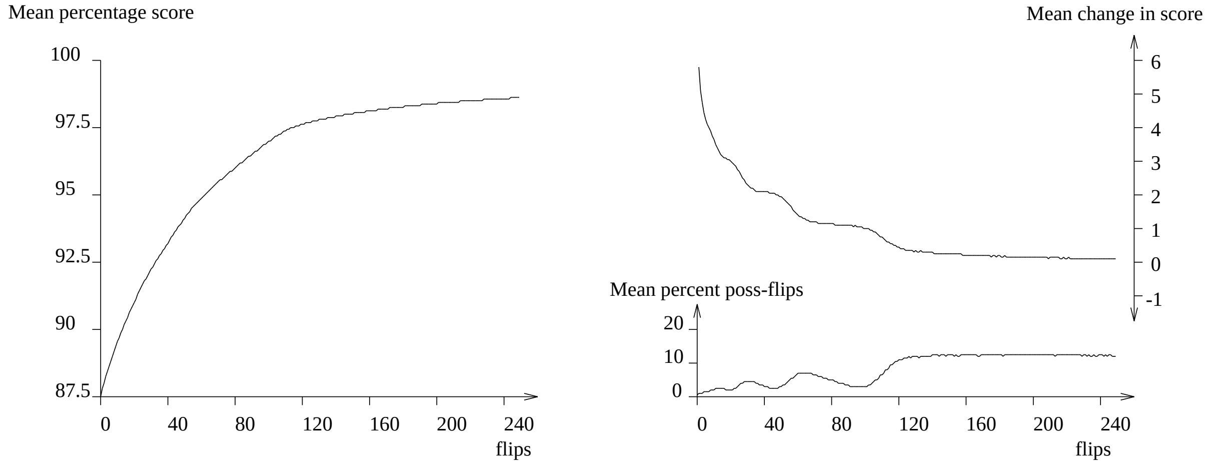
(b) Change in score, and poss-flips

Figure 1: GSAT's behaviour during one try, N = 500, L = 2150, first 250 flips

are four well defined wedges starting at 5, 16, 26, and 57 flips, with occasional sharp dips. These wedges demonstrate behaviour analogous to the that of poss-flips on the plateau. The plateau spans the region where flips typically do not change the score: we call this region H_0 since hill-climbing typically makes zero change to the score. The last wedge spans the region H_1 where hill-climbing typically increases the score by 1, as can be seen very clearly from Figure 1b. Again Figure 1b shows that the next three wedges (reading right to left) span regions H_2, H_3, and H_4. As with the transition onto the plateau, the transition between each region is marked by a sharp increase in poss-flips. Dips in the wedges represent unusual flips which increase the score by more than the characteristic value for that region, just as the dips in poss-flips on the plateau represent flips where an increase in score was possible. This exact correlation can be seen clearly in Figure 1b. Note that in no region H_j did a change in score of $j + 2$ occur, and that there was no change in score of -1 at all. In addition, each wedge in poss-flips appears to decay close to linearly. This is explained by the facts that once a variable is flipped it no longer appears in poss-flips (flipping it back would decrease score), that most of the variables in poss-flips can be flipped independently of each other, and that new variables are rarely added to poss-flips as a consequence of an earlier flip. On the plateau, however, when a variable is flipped which does not change the score, it remains in poss-flips since flipping it back also does not change the score.

To determine if this behaviour is typical, we generated 500 random 3-SAT problems with N=500 and L/N=4.3, and ran 10 tries of GSAT on each problem. Figure 2a shows the mean percentage score[3] while Figure 2b presents the mean percentage poss-flips together with the mean change in score at each flip. (The small discreteness in this figure is due to the discreteness of Postscript's plotting.)

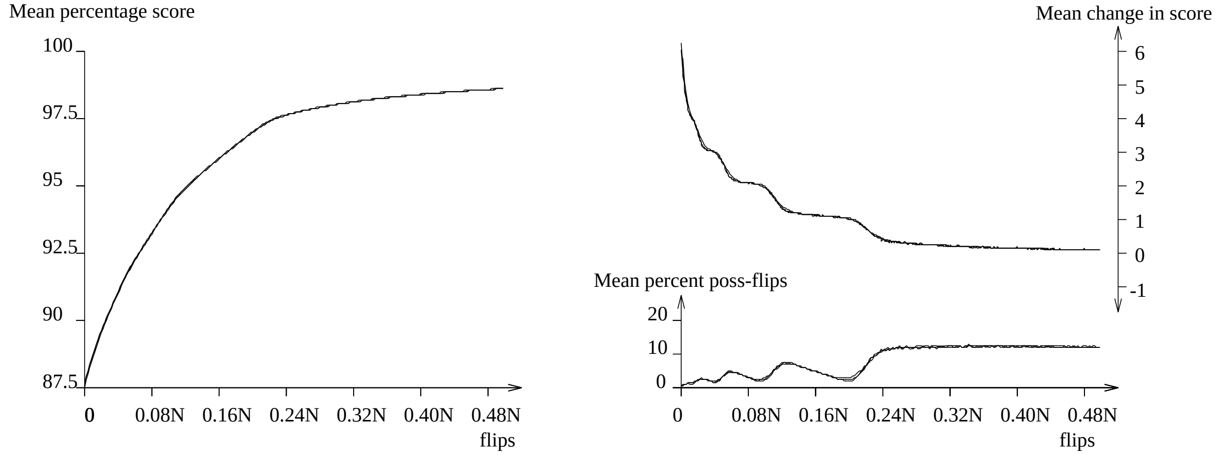
[3] In this paper we assign a score of 100% to flips which were not performed because a satisfying truth assignment had already been found.



(a) Mean score (b) Mean change in score, poss-flips

Figure 2: Mean GSAT behaviour, N = 500, L/N = 4.3, first 250 flips

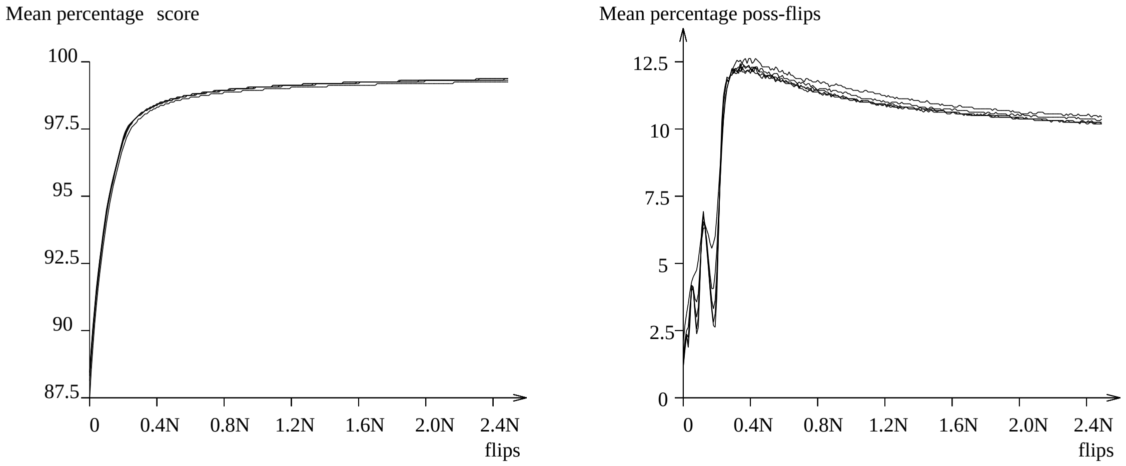
The average percentage score is very similar to the behaviour on the individual run of Figure 1, naturally being somewhat smoother. The graph of average poss-flips seems quite different, but it is to be expected that you will neither observe the sharply defined dips in poss-flips from Figure 1b, nor the very sharply defined start to the wedges, since these happen at varying times. It is remarkable that the wedges are consistent enough to be visible when averaged over 5,000 tries; the smoothing in the wedges and the start of the plateau is caused by the regions not starting at exactly the same time in each try.



(a) Mean score (b) Mean change in score, poss-flips

Figure 3: Scaling of mean GSAT behaviour, N = 500, 750, 1000, first 0.5N flips

Experiments with other values of N with the same ratio of clauses to variables demonstrated qualitatively similar behaviour. More careful analysis shows the remarkable fact that not only is the behaviour qualitatively similar, but quantitatively similar, with a simple linear dependency on N. If graphs similar to Figure 2



(a) mean score, L/N = 4.3

(b) mean poss-flips, L/N = 4.3

Figure 4: Scaling of mean GSAT behaviour, N = 100, 200, 300, 400, 500

are plotted for each N with the x-axis scaled by N, behaviour is almost *identical*. To illustrate this, Figure 3 shows the mean percentage score, percentage poss-flips, and change in score, for N = 500, 750, and 1000, for L = 4.3N and for the first 0.5N flips (250 flips at N = 500). Both Figure 3a and Figure 3b demonstrate the closeness of the scaling. Indeed, in each graph the scaling is so exact that the figures appear to contain just one, thick, line! However, in Figure 3b there is a slight tendency for the different regions of hill-climbing to become better defined with increasing N.

The figures we have presented clearly show scaling behaviour, but only reach a very early stage of plateau search. To investigate further along the plateau, we performed experiments with 100, 200, 300, 400, and 500 variables from 0 to 2.5N flips. In Figure 4a shows the mean percentage score in each case, while Figure 4b shows the mean percentage poss-flips, magnified on the y-axis for clarity. Both these figures demonstrate the closeness of the scaling on the plateau. Again, in Figure 4a the five average score graphs scale so well that the figure appears to contain only one line. In Figure 4b the graphs are not quite so close together. During hill-climbing the wedges become much better defined with increasing N. During plateau search, although separate lines are distinguishable, the difference is always considerably less than 1% of the total number of variables.

The problems used in these experiments (random 3-SAT with L/N=4.3) are believed to be unusually hard and are satisfiable with probability approximately $\frac{1}{2}$. Neither of these facts appears to be relevant to the scaling of GSAT's search. To check this we performed a similar range of experiments with a ratio of clauses to variables of 6. Although almost all such problems are unsatisfiable, we observed exactly the same scaling behaviour. The score does not reach such a high value as in Figure 4a, as is to be expected, but nevertheless shows the same linear scaling. On the plateau, the mean value of poss-flips is lower than before. We again observed this behaviour for L/N = 3, where almost all problems are satisfiable. Score approaches 100% faster than before, and a higher value of poss-flips is reached on the plateau, but the decay in the value of poss-flips seen in Figure 4b does not seem to be present.

To summarise, we have shown that GSAT's hill-climbing goes through several distinct phases, and that the average behaviour of certain important features scale linearly with N. These results provide a considerable advance on previous informal descriptions of GSAT's search.

5 Numerical Analysis

In this section, we will show that even more useful results can be obtained if the data presented graphically in §4 is analysed numerically. We divide our analysis into two parts: first we deal with the plateau search, where behaviour is relatively simple, then we analyse the hill-climbing search.

5.1 Plateau Search

On the plateau, both average score and poss-flips seem to scale linearly with N. We now examine this phenomenon more closely. Since both score and poss-flips appear to decay in an exponential fashion, we performed regression analysis on our experimental data using an exponential model. As neither score nor possible-flips decay to an obvious asymptote (for example, the average score does not seem to get arbitrarily close to 100%), it was not possible to use linear regression on logarithmic values of our data. We therefore performed non-linear regression using the CNLR option from SPSS [14]. Taking into account the prediction of linear scaling along both axes, and the direction from which the asymptote is approached, the models we are testing are:

$$\begin{aligned} S(x) &= \mathrm{N} \cdot (B - C \cdot e^{-\frac{x}{A \cdot \mathrm{N}}}) \qquad (1) \\ P(x) &= \mathrm{N} \cdot (E + F \cdot e^{-\frac{x}{D \cdot \mathrm{N}}}) \qquad (2) \end{aligned}$$

where x represents the number of flips, $S(x)$ the average score at flip x and $P(x)$ the average number of possible flips. To determine GSAT's behaviour just on the plateau, we analysed data starting from 0.4N flips, a time when plateau search always appears to have started (see §5.2). The data used was the same as that presented in §4, from 0.4N flips to 2.5N flips.

Table 1 shows the results of our analysis on average score. As expected, values of A and C are positive, indicating exponential decay upwards towards the asymptote $B \cdot \mathrm{N}$. This asymptote is always slightly less than L. As L/N increases it becomes more difficult to satisfy all the clauses, and the difference between $B \cdot \mathrm{N}$ and L increases. The fit of the data for each experiment is extremely close, suggesting that the model is good. Also, for each value of L/N, the predicted

L/N	N	A	B	C	R^2
3	100	0.481	2.996	0.0473	0.994
3	200	0.504	2.997	0.0431	0.995
3	300	0.510	2.997	0.0439	0.996
3	400	0.504	2.997	0.0438	0.996
3	500	0.511	2.997	0.0428	0.995
4.3	100	0.535	4.27	0.0815	0.995
4.3	200	0.549	4.27	0.0794	0.995
4.3	300	0.558	4.27	0.0783	0.994
4.3	400	0.557	4.27	0.0778	0.994
4.3	500	0.566	4.27	0.0772	0.995
6	100	0.462	5.89	0.117	0.994
6	200	0.488	5.89	0.114	0.994
6	300	0.496	5.89	0.112	0.994
6	400	0.492	5.89	0.114	0.993
6	500	0.492	5.89	0.112	0.993

Table 1: Regression results for average score of GSAT.[4]

parameters vary only slightly with N, providing further evidence for the scaling of GSAT's behaviour.

Some care is needed in interpreting these results. We cannot be sure of the distribution of average values of score. Such information is important to performing accurate regressions. We have also observed variation in the parameters A, B, and C if we perform regressions on larger numbers of flips than 2.5N. However, an excellent fit is still found whose predictions for the average score differ only very slightly. We suspect that the discreteness in $S(x)$, especially at large number of flips, may affect regression using a continuous model. All these points should be set against the fact that a remarkably good fit is found in each experiment, and that this is so despite the experiments being completely independent of each other.

Table 2 shows regression results for average poss-flips data based on the model (2), again with data taken from each experiment from 0.4N flips to 2.5N flips. For L/N = 4.3 and 6, we find that the data fits the model extremely well, and that the parameters D, E, and F are very consistent for varying N and fixed L/N. In particular, it seems that for L/N = 4.3 the asymptotic value of poss-flips is about 10% of N and that for 6 it is about 5% of N. We could not find, however, a good fit to the model at L/N = 3. It is likely that in this case GSAT performs too well

[4] The value of R^2 is a number in the interval $[0,1]$ indicating how well the variance in data is explained by the regression formula. $1 - R^2$ is the ratio between variance of the data from its predicted value, and the variance of the data from the mean of all the data. A value of R^2 close to 1 indicates that the regression formula fits the data very well.

L/N	N	D	E	F	R^2
4.3	100	0.993	0.101	0.0359	0.996
4.3	200	0.878	0.101	0.0348	0.998
4.3	300	0.888	0.100	0.0348	0.997
4.3	400	0.871	0.100	0.0338	0.997
4.3	500	0.838	0.100	0.0348	0.996
6	100	0.800	0.0553	0.0398	0.998
6	200	0.817	0.0513	0.0391	0.999
6	300	0.821	0.0504	0.0377	0.998
6	400	0.782	0.0504	0.0372	0.998
6	500	0.789	0.0502	0.0373	0.999

Table 2: Regression results on average poss-flips of GSAT.

to give a consistent story on the plateau and many tries solve problems in a small number of flips, and indeed that the mean value of poss-flips on the plateau may simply be a constant.

To summarise, we have performed a detailed analysis of GSAT's average score and poss-flips behaviour during plateau search. For L/N = 3, 4.3, 6, the average value of score can be predicted very well by a simple model of exponential decay towards an asymptotic value. For L/N = 3 this value is very close to 100% of clauses being satisfied, while for L/N = 4.3 it is approximately 99.3% of clauses and for L/N = 6 it is approximately 98.2% of clauses. The average value of poss-flips on the plateau can also be modelled very well by exponential decay for L/N = 4.3, 6 but not for L/N = 3. Our models predict that average behaviour varies linearly with N, with parameters that appear to be constant given fixed L/N.

5.2 Hill-climbing

We have also analysed GSAT's behaviour during its hill-climbing phase. In §4 we identified different phases of GSAT's hill-climbing: phases where most flips increase the score by 3, then by 2, then by 1. In Figures 1 & 2 each phase appears to last roughly twice the length of the previous one: that is, the phase where flips are on average size 2 is about twice as long as the phase where flips are on average size 3, etc. This motivates the following conjectures: GSAT moves through a sequence of regions H_j for $j = ..., 3, 2, 1$ in which the majority of flips increase the score by j, and where the length of each region H_j is proportional to 2^{-j} (except for the region H_0 which represents plateau search). In addition, the total length of these hill-climbing regions depends linearly on N.

To investigate this conjecture, we analysed 1000 tries (50 tries each on 20 different problems) for random 3-SAT problems at N=500 and L/N=4.3. Because we very rarely observe flips in H_j that increase the score by *less* than j, we defined

H_j as the region between the first flip which increases the score by exactly j and the first flip which increases the score by less than j (unless the latter actually appears before the former, in which case H_j is empty). One simple test of our conjecture is to compare the total time spent in H_j with the total time up to the end of H_j; we predict that this ratio will be $\frac{1}{2}$. For $j = 1$ to 4 the mean and standard deviations of this ratio and the length of each region were:[5]

Region	mean ratio	s.d.	mean length	s.d.
All climbing	—	—	112	7.59
H_1	0.486	0.0510	54.7	7.69
H_2	0.513	0.0672	29.5	5.12
H_3	0.564	0.0959	15.7	3.61
H_4	0.574	0.0161	7.00	2.48

This data supports our conjecture although as j increases each region is slightly longer than predicted. Note, however, that the region H_4 is only about 7 flips long and occurs at the start of the search when we would expect more variable behaviour (indeed in two tries the region H_4 was empty). It is thus difficult to draw conclusions from the deviation of the ratio from $\frac{1}{2}$: this could either be a genuine effect or due to noise. Examination of a frequency distribution of the lengths of each of these regions suggests they are normally distributed, but the fact that the standard deviation in the length of all climbing is less than that of H_1 suggests that the lengths of different regions within a given try are not independent. Note finally that the total length of hill-climbing at N=500 is 0.22N flips. At N=100 it is 0.23N. This is consistent with the scaling behaviour observed in §4.

Our conjecture has an appealing corollary. Namely, that if there are i non-empty hill-climbing regions, the average change in score per flip during hill-climbing is:

$$\frac{1}{2} \cdot 1 + \frac{1}{4} \cdot 2 + \frac{1}{8} \cdot 3 + \frac{1}{16} \cdot 4 + \cdots + \frac{1}{2^i} \cdot i \quad \approx \quad 2. \tag{3}$$

This approximation will improve as N and i increase. It follows from this that the average score at the end of hill-climbing is simply the initial score plus twice the length of hill-climbing (ignoring errors due to flips that do better or worse than expected). At N=500, we observed a mean ratio of change in score per flip during hill-climbing of 1.94 with a standard deviation of 0.1. At N=100, the ratio is 1.95 with a standard deviation of 0.2. Both these figures are slightly lower than predicted. This can be accounted for by the above approximation which slightly over-estimates the average change in score.

This simple model also ignores flips in H_j which increase the score by more than j. Such flips were seen in Figure 1b in regions H_3 to H_0, and are not uncommon. In the experiment reported on earlier in this section, 9.8% of flips in H_1 were of size 2, 6.3% of flips in H_2 were of size 3. However, flips of size $j + 2$ were very

[5] the data for "All climbing" is the length to the start of H_0.

rare, forming only about 0.02% of all flips in H_1 and H_2. By examining graphs of the average difference in score, we conjectured that an exponential decay similar to that in H_0 occurs in each H_j. To be precise, we conjecture that the average change in number of satisfied clauses from flip x to flip $x+1$ in H_j is given by:

$$j + E_j \cdot e^{-\frac{x}{D_j \cdot \mathrm{N}}} \tag{4}$$

In the next section we argue that this may correspond to a model of GSAT's search in which there are a certain number of flips of size $j+1$ in each region H_j, and the probability of making a $j+1$ flip is merely dependent on the number of such flips left. The rest of the time, GSAT is obliged to make a flip of size j. We tested this model using the non-linear regression option from SPSS. our data from 1000 tries fitted the model well, giving values of R^2 of 96.8% for H_1 and 97.5% for H_2. The regression gave estimates for the parameters of: $D_1 = 0.045$, $E_1 = 0.25$, $D_2 = 0.025$, $E_2 = 0.15$. Not surprisingly, since the region H_3 is very short, data was too noisy to obtain a better fit with the model (4) than with one of linear decay. These results do, however, support our conjecture, but more experiments on larger problems are needed to lengthen the region H_j for $j \geq 3$.

6 Theoretical Analysis

Empirical results like those given in §5 can be used to direct efforts to analyse algorithms theoretically. For example, consider the plateau region of GSAT's search. The results of §5.1 suggest that the average score at flip x is given by the model,

$$S(x) \quad = \quad \mathrm{N} \cdot (B - C \cdot e^{-\frac{x}{A \cdot \mathrm{N}}})$$

where A, B and C are independent of N. On successful tries, $\mathrm{N} \cdot B = \mathrm{L}$. Thus,

$$S(x) \quad = \quad \mathrm{L} - C \cdot \mathrm{N} \cdot e^{-\frac{x}{A \cdot \mathrm{N}}}$$

Differentiating with respect to x we get,

$$\begin{aligned} \frac{\mathrm{d}S(x)}{\mathrm{d}x} &= \frac{C}{A} \cdot e^{-\frac{x}{a \cdot \mathrm{N}}} \\ &= \frac{\mathrm{L} - S(x)}{A \cdot \mathrm{N}} \end{aligned}$$

Now the gradient is a good approximation for D_x, the average size of a flip at x. Hence,

$$D_x \quad = \quad \frac{\mathrm{L} - S(x)}{A \cdot \mathrm{N}}$$

But,

$$D_x = \sum_{j=-\mathrm{L}}^{\mathrm{L}} j \cdot prob(D_x = j)$$

where $prob(D_x = j)$ is the probability that a flip at x is of size j. Our experiments suggest that downward flips and those of more than +1 are very rare on the plateau. Thus, a good (first order) approximation is that,

$$\begin{aligned} D_x &= \sum_{j=0}^{1} j \cdot prob(D_x = j) \\ &= prob(D_x = 1) \end{aligned}$$

Hence,

$$prob(D_x = 1) = \frac{\mathrm{L} - S(x)}{A \cdot \mathrm{N}}$$

That is, on the plateau the probability of making a flip of size +1 is directly proportional to $\mathrm{L} - S(x)$, the average number of clauses remaining unsatisfied and inversely proportional N, to the number of variables. A similar analysis and result can be given for $prob(D_x = j + 1)$ in the hill-climbing region H_j, which would explain the model (4) proposed in 5.2. We expect that such conjectures will be very useful in determining various properties of GSAT like the average runtime and the optimal setting for a parameter like Max-flips. In addition, if we can develop a model of GSAT's search in which $prob(D_x = j)$ is related to the number of unsatisifed clauses and N as in the above equation, then the exponential behaviour and linear scaling of the score we have observed immediately follows.

7 Related Work

GSAT was introduced in [13]. In [3] we describe an empirical study of GenSAT, a family of procedures related to GSAT. This study focuses on the importance of randomness, greediness and hill-climbing for the effectiveness of these procedures. In addition, we determine how performance depends on parameters like Max-tries and Max-flips. We showed also that certain variants of GenSAT could outperform GSAT on random problems. It would be very interesting to perform a similar analysis to that given here of these closely related procedures.

A closely related set of procedures has also been studied by Gu [5]. However, these procedures have a different control structure to GSAT which allows them, for instance, to make sideways moves when upwards moves are possible. This makes it difficult to compare results directly. Nevertheless, we are confident that the approach taken here would apply equally well to these procedures, and that similar results could be expected.

Procedures like GSAT have also been successfully applied to constraint satisfaction problems other than satisfiability. For example, [10] have proposed a greedy local search procedure which performed well scheduling observations on the Hubble Space Telescope, and other constraint problems like the million-queens, and 3-colourability. It would be very interesting to see how the results given here map across to these new problem domains.

8 Conclusions

We have described an empirical study of search in GSAT, an approximation procedure for satisfiability. We performed detailed analysis of the two basic phases of GSAT's search, an initial period of fast hill-climbing followed by a longer period of plateau search. We have shown that the hill-climbing phases can be broken down further into a number of distinct phases each corresponding to progressively slower climbing, and each phase lasting twice as long as the last. We have also shown that, in certain well defined problem classes, the average behaviour of certain important features of GSAT's search (the average score and the average branching rate at a given point) scale in a remarkably simple way: linearly with the number of variables. We have also demonstrated that the behaviour of these features can be modelled very well by simple exponential decay, both in the plateau and in the hill-climbing phase. Finally, we used our experiments to conjecture various properties (*eg.* the probability of making a flip of a certain size) that will be useful in a theoretical analysis of GSAT. These results illustrate how carefully performed experiments can be used to guide theory, and how computers have an increasingly important rôle to play in the analysis of algorithms.

Acknowledgements

This research is supported by a SERC Postdoctoral Fellowship to the first author and a HCM Postdoctoral fellowship to the second. We thank Alan Bundy, Ian Green, and the members of the Mathematical Reasoning Group for their constructive comments and for the trillions of CPU cycles donated to these and other experiments, and Andrew Bremner, Judith Underwood, for other help.

References

[1] 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.

[2] J.M. Crawford and L.D. Auton. Experimental Results on the Cross-Over Point in Satisfiab ility Problems. In *Proceedings of AAAI 1993 Spring Symposium on AI and NP-Hard Pro blems*, 1993.

[3] I. P. Gent and T. Walsh. Towards an Understanding of Hill-climbing Procedures for SAT. In *Proceedings of the 11th National Conference on AI*. American Association for Artificial Intelligence, 1993.

[4] Ian P. Gent and Toby Walsh. The enigma of SAT hill-climbing procedures. Research Paper 605, Dept. of Artificial Intelligence, Edinburgh, 1992.

[5] Jun Gu. Efficient local search for very large-scale satisfiability problems. *SIGART Bulletin*, 3(1), January 1992.

[6] J. N. Hooker. Needed: An empirical science of algorithms. Technical report, Graduate School of Industrial Administration, Carnegie Mellon University, Pittsburgh PA, January 1993.

[7] Elias Koutsoupias and Christos H. Papadimitriou. On the greedy algorithm for satisfiability. *Information Processing Letters*, 43:53–55, August 1992.

[8] 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.

[9] C.C. McGeoch. *Experimental Analysis of Algorithms*. PhD thesis, Carnegie Mellon University, 1986. Also available as CMU-CS-87-124.

[10] Steven Minton, Mark D. Johnston, Andrew B. Philips, and Philip Laird. Solving large-scale constraint satisfaction and scheduling problems using a heuristic repair method. In *AAAI-90, Proceedings Eighth National Conference on Artificial Intelligence*, pages 17–24. AAAI Press/MIT Press, 1990.

[11] David Mitchell, Bart Selman, and Hector Levesque. Hard and easy distributions of SAT problems. In *AAAI-92: Proceedings Tenth National Conference on Artificial Intelligence*. AAAI Press/The MIT Press, July 12-16 1992.

[12] Bart Selman and Henry Kautz. Domain-independent extensions to GSAT: Solving large structured satisfiability problems. In *Proceedings 11th National Conference on Artificial Intelligence*. American Association for Artificial Intelligence, 1993.

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

[14] *SPSS-X User's Guide*. SPSS Inc., 1988. 3rd Edition.