- How fast does the CGA with elitist selection converge towards an optimal solution?
- Does the CGA without elitist selection converge to a population with mostly optimal solutions, and how fast?

12 Combinatorial Phase Transitions

12.1 Phenomena and Models

"Where the Really Hard Problems Are" (Cheeseman et al. 1991)

Many NP-complete problems can be solved in polynomial time "on average" or "with high probability" for reasonable-looking distributions of problem instances. E.g. Satisfiability in time $O(n^2)$ (Goldberg et al. 1982), Graph Colouring in time $O(n^2)$ (Grimmett & McDiarmid 1975, Turner 1984).

Where, then, are the (presumably) exponentially hard instances of these problems located? Could one tell ahead of time whether a given instance is likely to be hard?

Early studies of this issue done by: Yu & Anderson (1985), Hubermann & Hogg (1987), Cheeseman, Kanefsky & Taylor (1991), Mitchell, Selman & Levesque (1992), Kirkpatrick & Selman (1994), etc.

Hard Instances for 3-SAT

Mitchell, Selman & Levesque (AAAI 1992).

Experiments on the behaviour of the Davis-Putnam[-Logemann-Loveland] (DP[LL]) procedure on randomly generated 3-cnf Boolean formulas.

E.g. satisfiable 3-cnf formula

$$(x_1 \vee \bar{x}_2 \vee x_3) \wedge (\bar{x}_1 \vee x_2 \vee \bar{x}_4)$$

The expressions in parenthesis are *clauses* and the *x*'s are *literals*.

Distribution of test formulas:

- number of variables
- $m = \alpha n$ randomly generated clauses of 3 literals, $2 \leq \alpha \leq 8$

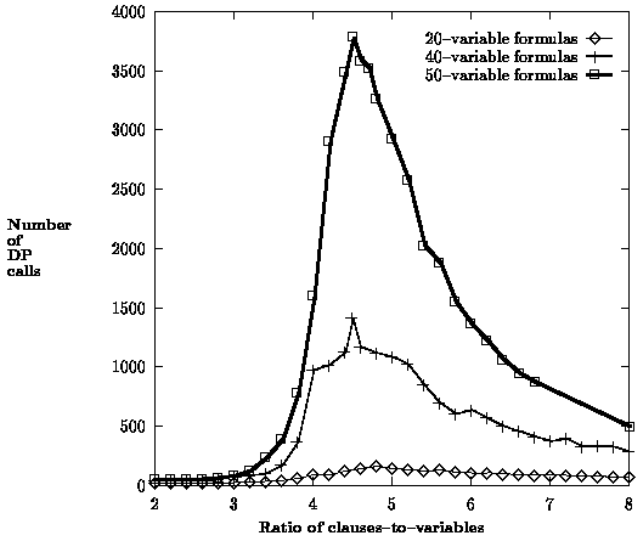

Figure 4: Number of DPLL calls required to determine satisfiability (Mitchell et al. 1992).

TThe Davis-Putnam[-Logemann-Loveland] (DP[LL]) method for testing the satisfiability of a set of clauses Σ on the variable set V:

1. If Σ is empty, return "satisfiable".
2. If Σ contains an empty clause, return "unsatisfiable".
3. If Σ contains a unit clause $c = x^{\pm}$, assign to x a value which satisfies c, simplify the remaining clauses correspondingly, and call DPLL recursively.
4. Otherwise select an unassigned $x \in V$, assign $x \leftarrow 1$, simplify Σ, and call DPLL recursively. If this call returns "satisfiable", then return "satisfiable"; else assign $x \leftarrow 0$, simplify Σ, and call DPLL recursively again.

For each set of 500 formulas, Mitcell et al. plotted the median number of DPLL calls required for solution.

The results of this experiment are illustrated in Figures 4 and 5. Discussion:

- A clear peak in running times (number of DPLL calls) near the point where 50% of formulas are satisfiable.
- The "50% satisfiable" point or "satisfiability threshold" seems to be located at roughly $\alpha \approx 4.25$ for large n.

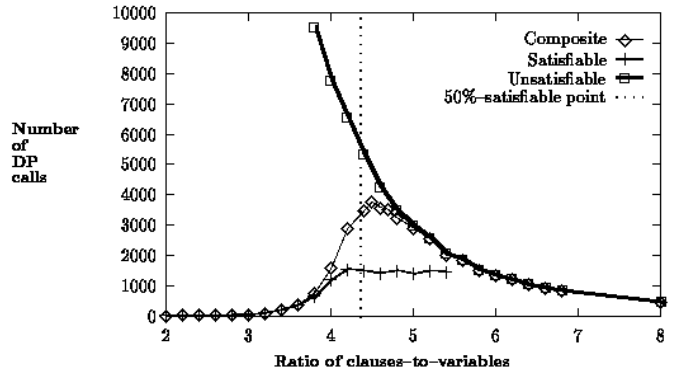

Figure 5: Number of required DPLL calls according to type of formula (Mitchell et al. 1992).

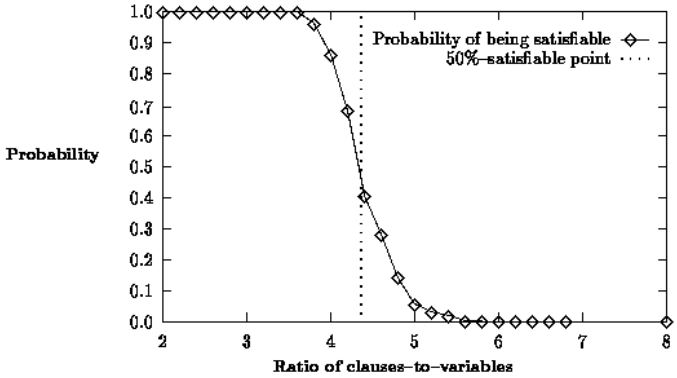

Figure 6: Probability of satisfiability for random 3-cnf formulas (Mitchell et al. 1992).

- The peak seems to be caused by relatively short unsatisfiable formulas.

A fundamental question is whether the connection of the running time peak and the satisfiability threshold a characteristic of the DPLL algorithm, or a (more or less) algorithm independent "universal" feature?

The "50% satisfiable" point or "satisfiability threshold" for 3-SAT seems to be located at $\alpha \approx 4.25$ for large n.

12.2 Statistical Mechanics of k-SAT ("1st-Order Analysis")

Kirkpatrick & Selman (Science 1994)

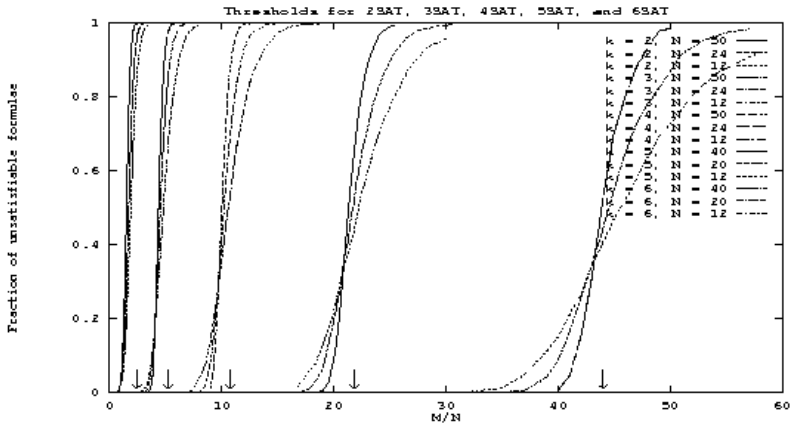

Figure 7: Probability of satisfiability for random k-cnf formulas (Kirkpatrick & Selman 1994).

Similar experiments as above for k-SAT, $k = 2, \ldots, 6$, 10000 formulas per data point. Results illustrated in Figure 7. Further observations:

- The "satisfiability threshold" α_c shifts quickly to larger values of α for increasing k.
- For fixed k, the value of α_c drifths slowly to smaller values for increasing n.

A statistical mechanics model of a k-cnf formula:

- variables x_i $\sim$ spins with states ± 1
- clauses c $\sim$ k-wise interactions between spins
- truth assignment σ $\sim$ state of spin system
- Hamiltonian $H(\sigma)$ $\sim$ number of clauses unsatisfied by σ
- α_c $\sim$ critical "interaction density" point for "phase transition" from "satisfiable phase" to "unsatisfiable phase"

Estimates of α_c for various values of k via "annealing approximation", "replica theory", and observation:

k	α_{ann}	α_{rep}	α_{obs}
2	2.41	1.38	1.0
3	5.19	4.25	4.17 ± 0.03
4	10.74	9.58	9.75 ± 0.05
5	21.83	20.6	20.9 ± 0.1
6	44.01	42.8	43.2 ± 0.2

The "annealing approximation" means simply assuming that the different clauses are satisfied independently. This leads to the following estimate:

- The probability that a given clause c is satisfied by a random σ: $p_k = 1 - 2^{-k}$.
- The probability that a random σ satisfies all $m = \alpha n$ clauses assuming independence: $p_k^{\alpha n}$.
- Total number of satisfying assignments $= 2^n p_k^{\alpha n} \triangleq S_k^n(\alpha)$.
- For large n, $S_k^n(\alpha)$ falls rapidly from 2^n tp 0 near a critical value $\alpha = \alpha_c$. Where is α_c?
- One approach: solve for $S_k^n(\alpha) = 1$.

$$S_k^n(\alpha) = 1 \Leftrightarrow 2p_k^\alpha = 1$$
$$\Leftrightarrow \alpha = -\frac{1}{\log_2 p_k} = -\frac{\ln 2}{\ln(1-2^{-k})} \approx -\frac{\ln 2}{2^{-k}} = (\ln 2) \cdot 2^k.$$

It is in fact known that:

- A sharp satisfiability threshold α_c exists for all $k \geq 2$ (Friedgut 1999).
- For $k = 2$, $\alpha_c = 1$ (Goerdt 1982, Chvátal & Reed 1982). Note that 2-SAT $\in$ P.
- For $k = 3$, $3.14 < \alpha_c < 4.51$ (lower bound due to Achlioptas 2000, upper bound to Dubois et al. 1999).
- Current best empirical estimate for $k = 3$: $\alpha_c \approx 4.27$ (Braunstein et al. 2002).

12.3 Local Search Methods for 3-SAT

Local search methods (e.g. simulated annealing, genetic algorithms) can be used for finding (with high probability) satisfying truth assignments to randomly generated 3-cnf formulas in the satisfiable phase ($m/n = \alpha < \alpha_c$).

Consider first a general objective function $E = E(x)$ to be minimised. Then the basic local search scheme is:

- Start with some randomly chosen feasible solution $x = x_0$.
- If value of $E(x)$ is not "good enough", search for some "neighbour" x' of x that satisfies $E(x') \lesssim E(x)$. If such an x' is found, set $x \leftarrow x'$ and repeat.
- If no improving neighbour is found, then either restart at new random $x = x_0$ or relax the neighbourhood condition [algorithm-dependent].

In the setting of the 3-SAT problem, the objective function to be minimised is $E = E_F(s) =$ the number of unsatisfied clauses in formula F under truth assignment s. When $\alpha < \alpha_c$, an assignment s satisfying $E(s) = 0$ exists with high probability, and local search techniques are surprisingly powerful in finding such assignments.

The first systematically tested algorithm of this type was the following procedure GSAT by (Selman et al. 1992):

```
GSAT(F):
  s = initial truth assignment;
  while flips < max_flips do
    if s satisfies F then output s & halt, else:
    - find a variable x whose flipping causes
      largest decrease in E (if no decrease is
      possible, then smallest increase);
    - flip x.
```

An improvement to GSAT is to augment it with a fraction p of random walk moves, leading to algorithm NoisyGSAT (Selman et al. 1996):

```
NoisyGSAT(F,p):
  s = initial truth assignment;
  while flips < max_flips do
    if s satisfies F then output s & halt, else:
    - with probability p, pick a variable x
      uniformly at random and flip it;
```

```
    - with probability (1-p), do basic GSAT move:
      - find a variable x whose flipping causes
        largest decrease in E (if no decrease is
        possible, then smallest increase);
      - flip x.
```

A subtle but important change to NoisyGSAT is to *focus* the search on the presently unsatisfied clauses. This leads to the current "industry standard" WalkSAT algorithm (Selman et al. 1996):

```
WalkSAT(F,p):
  s = initial truth assignment;
  while flips < max_flips do
    if s satisfies F then output s & halt, else:
    - pick a random unsatisfied clause C in F;
    - if some variables in C can be flipped without
      breaking any presently satisfied clauses,
      then pick one such variable x at random; else:
    - with probability p, pick a variable x
      in C at random;
    - with probability (1-p), pick an x in C
      that breaks a minimal number of presently
      satisfied clauses;
    - flip x.
```

The focusing seems to be important: in the (somewhat unsystematic) experiments performed by Selman et al. (1996), WalkSAT outperforms NoisyGSAT by several orders of magnitude.

Also other local search techniques can be applied to the satisfiability problem. Good results have been obtained e.g. with the following Record-to-Record Travel (RRT) method first introduced in the context of the TSP problem (Dueck 1993):

```
RRT(E,d):
  s = initial feasible solution;
  s* = s; E* = E(s);
  while moves < max_moves do
    if s is a global min. of E then output s & halt,
    else:
      pick a random neighbour s' of s;
      if E(s') <= E* + d then let s = s';
      if E(s') < E* then:
```

```
s* = s'; E* = E(s').
```

In applying RRT to SAT, one chooses again $E(s) =$ number of clauses unsatisfied by truth assignment s, together with single-variable flip neighbourhoods. Imposing the *focusing* heuristic of always selecting the flipped variables from unsatisfied clauses (precisely: one unsatisfied clause is chosen at random, and from there a variable at random) leads to the "focused RRT" (FRRT) algorithm for 3-SAT, which is quite competitive with WalkSAT (Seitz & Orponen 2003).

12.4 Statistical Mechanics of K-SAT ("Replica Analysis")

The analyses in this area are rather technical, so we present just some basic ideas.

Consider again the statistical mechanics model of k-SAT formulas discusses on p. 125. I.e. we consider the ensemble of random k-cnf formulas with n variables and $m = \alpha n$ clauses. The Boolean-valued variables x_i are mapped to binary-state spins as $x_i \in \{\text{true}, \text{false}\} \mapsto \text{spin } S_i \in \{+1, -1\}$.

A formula consists of a set of clauses C_l represented in terms of an "interaction matrix" $C = (C_{li})$:

$$C_{li} = \begin{cases} +1, & \text{if } C_l \text{ includes } x_i \\ -1, & \text{if } C_l \text{ includes } \overline{x_i} \\ 0, & \text{otherwise} \end{cases}$$

Thus,

$$\sum_{i=1}^{n} C_{li} S_i = -K$$

if and only if all the literals in clause C_l are "wrong", i.e. the clause is unsatisfied by truth assignment (spin state) $S = (S_1, \ldots, S_n)$.

We consider the Hamiltonian function

$$E[S,C] = \sum_{l=1}^{m} \delta\left(\sum_{i=1}^{n} C_{li} S_i + K\right) \quad = \text{number of clauses in } C \text{ unsatisfied by } S,$$

$$\delta(u) = \begin{cases} 1, & \text{if } u = 0 \\ 0, & \text{otherwise} \end{cases}$$

The ground state potential (minimum number of unsatisfied clauses) of a given system C is $E^*[C] = \min_S E[S,C]$. For randomly generated C, $\Pr(E^*[C] = 0)$with

high probability when α is small, and we would like to approximate the value $\alpha = \alpha_c(K)$ where this property ceases to hold.

This is however a very difficult problem, so we approach it indirectly by considering rather the average of $E^*[C]$ with respect to C, denoted $E_{GS} = \overline{E^*[C]}$. (Such averages with respect to system parameters are called "quenched averages", as opposed to the more usual "thermal averages" computed with respect to system states.)

For large n, the distribution of $E^*[C]$ is highly concentrated around $E_{GS} = E_{GS}(\alpha, K)$. ($E^*$ is said to be "self-averaging".) In particular:

$$E_{GS} \approx 0 \text{ in the sat. phase } (\alpha < \alpha_c(K)),$$
$$E_{GS} > 0 \text{ in the unsat. phase } (\alpha > \alpha_c(K)).$$

Thus, we use the behaviour of E_{GS} as a guide to determining the value of α_c.

It is known that

$$E_{GS} = -T\overline{\ln Z_T[C]} + O(T^2)$$

as $T \to 0$, where

$$Z_T[C] = \sum_S \exp(-E[S,C]/T).$$

(This follows by averaging from the fundamental thermodynamic formula $F = E - TS = -kT \ln Z$ (p. 60).)

The important, but complicated quantity $\overline{\ln Z}$ can be estimated using the so called "replica method".

Consider the Taylor expansion of Z^ν as a function of ν for small ν:

$$Z^\nu = e^{\nu \ln Z} = 1 + \nu \ln Z + O(\nu^2)$$

Thus, for a fixed $Z > 0$:

$$\ln Z = \lim_{\nu \to 0} \frac{Z^\nu - 1}{\nu}.$$

Applying this to $\ln Z_T[C]$ and averaging over C yields:

$$E_{GS} = -T \lim_{\nu \to 0} \frac{1}{\nu} \left(\overline{Z_T[C]^\nu} - 1 \right) + O(T^2) \tag{11}$$

as $T \to 0$.

Now assume that the "small ν" is in fact an integer. Then:

$$\begin{aligned}\overline{Z_T[C]^\nu} &= \overline{\left(\sum_S \exp(-E[S,C]/T)\right)^\nu} \\ &= \sum_{S_1}\ldots,\sum_{S_\nu}\overline{\exp\left(-\sum_{r=1}^{\nu} E[S_r,C]/T)\right)}\end{aligned}$$

Thus we have transformed the problem of computing $\overline{Z_T^\nu}$ to the consideration of ν interconnected "replicas" of the original system.

This modified structure can further be viewed as a single system consisting of n vector-valued spins $\vec{\sigma}_i \in \{+1,-1\}^\nu, i = 1,\ldots,n$, with (non-random) potential function

$$E_{eff}[\vec{\sigma}_1,\ldots,\vec{\sigma}_n] = -T\ln\left[\overline{\exp\left(-\sum_{r=1}^{\nu} E[S_r,C]/T\right)}\right].$$

One can easily check that with this choice:

$$\overline{Z_T^\nu} = Z_T^{eff} = \sum_{\{\vec{\sigma}_i\}} \exp(-E_{eff}[\{\vec{\sigma}\}]/T).$$

This partition function may in some cases be so concentrated that for large n:

$$\overline{Z_T^\nu} = Z_T^{eff} \approx e^{-n\tilde{f}_T(\nu)} \approx 1 - n\tilde{f}_T(\nu),$$

where $\tilde{f}_T(\nu)$ is some nonlinear function with $\tilde{f}_T(0) = 0$.

Plugging this estimate in formula (11) yields

$$E_{GS} \approx -T\lim_{\nu\to 0}\frac{-n\tilde{f}_T(\nu)}{\nu} = Tn\tilde{f}_T'(0).$$

The replica method has been partially mathematically vindicated, i.e. the requisite "analytic continuation" from integer to real ν is justified under some conditions, although not generally.

From an application point of view, approximating the function $\tilde{f}_T(\nu)$ is the difficult part of the technique.