NEW RESULTS IN STATISTICAL MECHANICS OF COMBINATORIAL OPTIMIZATION

Towards a typical-case complexity theory

- The satisfiability problem
- Computer-science background
- The 3SAT phase transition
- Stat. mech. approach to the transition
- Analysis of algorithms
- Outlook

American Scientist
the magazine of Sigma Xi, the Scientific Research Society

Volume 85, Number 2
March–April, 1997
pages 108–112

CAN'T GET NO SATISFACTION

Brian Hayes

You are chief of protocol for the embassy ball. The crown prince instructs you either to invite Peru or to exclude Qatar. The queen asks you to invite either Qatar or Romania or both. The king, in a spiteful mood, wants to snub either Romania or Peru or both. Is there a guest list that will satisfy the whims of the entire royal family?

This contrived little puzzle is an instance of a problem that lies near the root of theoretical computer science. It is called the satisfiability problem, or SAT, and it was the first member of the notorious class known as NP-complete problems. These are computational tasks that seem intrinsically hard, but after 25 years of effort no one has yet proved that they are necessarily difficult. It remains possible (though unlikely) that we are simply attacking them by clumsy methods, and if we could dream up a clever algorithm they would all turn out to be easy. Settling this question is the most conspicuous open challenge in the theory of computation.

SAT also has practical importance. In artificial intelligence various methods of logical deduction and theorem-proving are related to SAT. And

one phase almost all the propositions can be satisfied, but in another phase almost none can. The cases that are hardest to resolve lie near the transition between these regimes.

The connection between SAT and the physics of phase transitions strikes me as a surprising one—a classic who'd-have-thunk-it result. We are accustomed to using mathematics as a tool for interpreting the physical world, but not the other way around. And yet the phase-transition model of SAT works so well that it cannot be a mere metaphor, much less a coincidence.

P and NP

The problem of the embassy ball is small enough to be solved by even the most plodding of methods. The problem is represented by the formula:

$$\boxed{(p \text{ OR } \sim q) \text{ AND } (q \text{ OR } r) \text{ AND } (\sim r \text{ OR } \sim p)}$$

Here p, q and r are Boolean variables, whose only possible values are *true* or *false*. The ~ symbol indicates negation, so that ~p is read "not p." The logical OR operation is defined so that (p OR q) has the value *true* if either p or q is *true*, whereas (p AND q)

THE K-satisfiability problem (K-SAT)

- N Boolean variables $x_i \in \{\text{FALSE}, \text{TRUE}\}$, $i = 1, \dots, N$
- M K-clauses: $(\mu = 1, \dots, M)$

$$C_\mu = x_{i_1^\mu} \vee \overline{x}_{i_2^\mu} \vee \dots \vee \overline{x}_{i_K^\mu}$$

logical OR

→ at least one variable has to be assigned in "right" way to satisfy C_μ

→ K-SAT formula

$$F = \bigwedge_{\mu=1}^{M} C_\mu$$

Decision problem: Is there an assignment to all Boolean variables such that F evaluates to **TRUE**, i.e., such that all clauses are satisfied simultaneously?

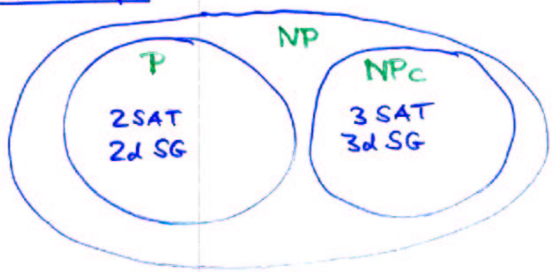
⇒ prototype decision problem

COMPLEXITY THEORY

Classification of decision problems according to their computational complexity = dependence of resolution time on problem size

- class P = polynomially solvable problems
  - → easy problems
- class NP = non-deterministic polynomial problems
  - → testing candidate solution is polynomial
  - → in general exponential number of candidates
  - → $P \subset NP$
- class NPc = NP-complete problems
  - → each problem $\in NP$ can be polynomially mapped to each NP-c problem
  - → hardest problems in NP
  - → exponential ??? $P \neq NP$???
  - → one of the 7 Millenium Prize Problems!

general belief:



FROM THE WORST CASE TO TYPICAL CASES

complexity theory - worst-case scenario

BUT: What happens in typical cases ?

$\rightarrow$ need ensemble of instances

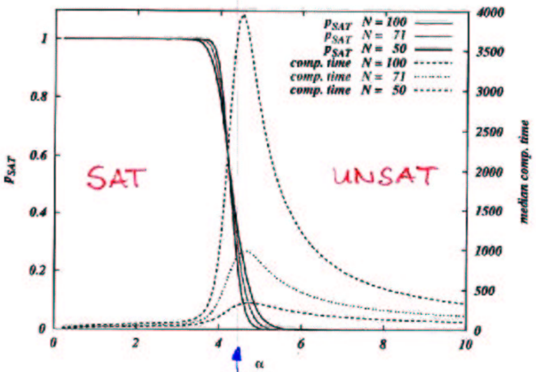
Mitchell et al. '92: Random 3-SAT

$\longrightarrow$ randomly generated clauses C_μ:

- select randomly 3 variables from $\{x_1, \dots, x_N\}$
- negate each variable independently with probability $1/2$

$\longrightarrow$ ensemble parametrized by constraintness

$$\alpha = \frac{M}{N}$$



#{algorithmic steps}

$\alpha_c \simeq 4.25$

$\longrightarrow$ numerical evidence for SAT/UNSAT phase transition

$\longrightarrow$ exponential complexity peak close to α_c

THE SAT/UNSAT TRANSITION

Rigorous results

- Friedgut '99: sharpness of threshold
- Kaporis, Kirousis, Lalas '02:

$$\alpha_c > 3.42$$

$\longrightarrow$ from analysis of heuristic algorithm

- Dubois, Boufkhad, Mandler '00:

$$\alpha_c < 4.51$$

$\longrightarrow$ from refined annealed approximation

Statistical physics approach

- Monasson, Zecchina '96
  Biroli, Monasson, MW '00
  Mézard, Parisi, Zecchina '02

$\longrightarrow$ introduce cost function

$$H = \#\{\text{unsatisfied clauses}\}$$

$$= \frac{\alpha N}{8} - \sum_{i=1}^{N} H_i S_i - \sum_{i<j} T_{ij} S_i S_j - \sum_{i<j<k} J_{ijk} S_i S_j S_k$$

with

$$S_i = \begin{cases} -1 & \text{if } x_i = \text{FALSE} \\ +1 & \text{if } x_i = \text{TRUE} \end{cases}$$

$$H_i = \frac{1}{8} \sum_{\mu=1}^{M} C_{\mu,i} \quad ; \quad T_{ij} = -\frac{1}{8} \sum_{\mu} C_{\mu,i} C_{\mu,j} \quad ; \quad J_{ijk} = \frac{1}{8} \sum_{\mu} C_{\mu i} C_{\mu j} C_{\mu k}$$

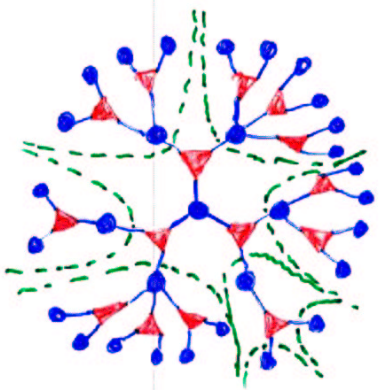
$$C_{\mu,i} = \begin{cases} +1 & \text{if } x_i \in C_\mu \\ -1 & \text{if } \bar{x}_i \in C_\mu \\ 0 & \text{else} \end{cases}$$

→ satisfying assignments = zero-energy ground states of H

→ stat. mech. at T = 0 !

⚠ H is a spin-glass Hamiltonian:
- quenched disorder
- dilution

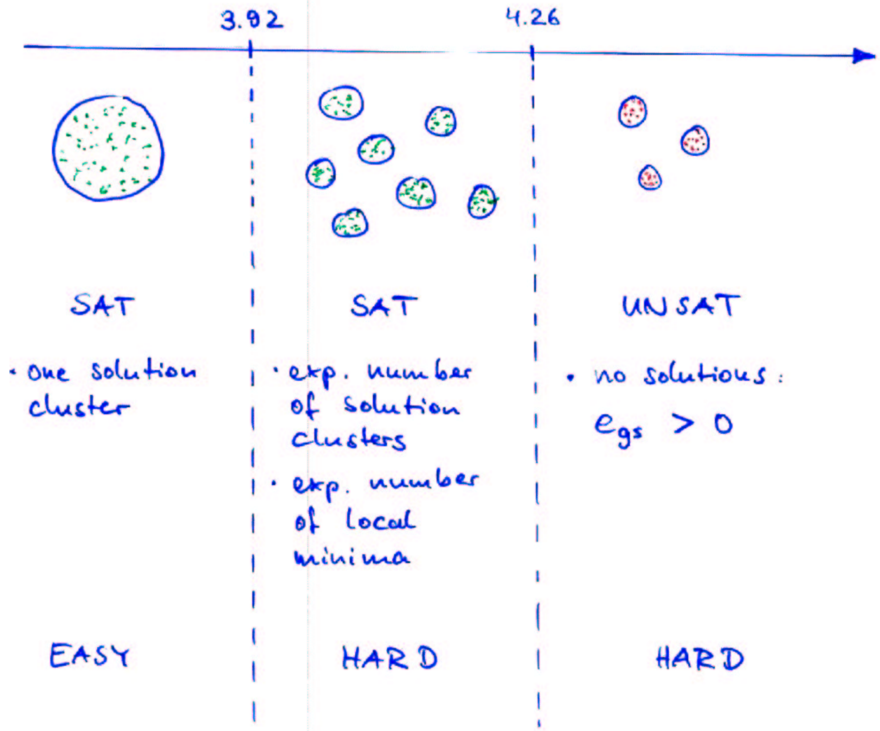
→ methods:
- functional replica method
- cavity method



- variable
- clause

→ iterative procedure
→ loops enter self-consistently
→ • one pure state ↔ replica symmetry ↔ Bethe-Peierls
  • many pure states ↔ replica symmetry breaking

Solution Space Structure



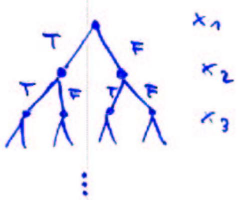
analytical solution → new algorithm
- RS ↔ belief propagation (~ '60s)
- RSB ↔ survey propagation (Mézard, Parisi, Zecchina '02)
  - iteration on single SAT instance
  - reduction of problem by fixing biased variables

→ up to $N \simeq 10^7$ in hard SAT phase!

ANALYSIS OF ALGORITHMS

- complete algorithms
  → prove satisfiability / unsatisfiability

main idea: organize assignments in backtracking tree



- stochastic local search
  → stops if solution found
  → cannot prove unsatisfiability
  → for sat. instances frequently more efficient

ex.:
- random walk
- simulated annealing
- walksat

THE DAVIS-PUTNAM ALGORITHM

step	clauses	search tree
0	$w \vee \bar{x} \vee y$ $\bar{w} \vee x \vee z$ $\bar{w} \vee \bar{x} \vee \bar{y}$ $\bar{w} \vee \bar{x} \vee y$ $x \vee y \vee \bar{z}$	
1	split: $w = T$	
2	$x \vee z$ $\bar{x} \vee \bar{y}$ $\bar{x} \vee y$ $x \vee y \vee \bar{z}$	
3	split: $x = T$	
4	$\bar{y}$ y	
5	unit clause propagation: $y = F$, $y = T$ → contradiction	
6	backtracking to stage 1, $x = F$	
7	z $y \vee \bar{z}$	
8	UC prop. $z = T$	
9	y	
10	UC prop. $y = T$	

⇒ solution $w = T$, $x = F$, $y = T$, $z = T$

THE 2+p - SAT PROBLEM

- 3 - clauses → α

action ⇓ of the algorithm

- 3 - clauses and 2 - clauses

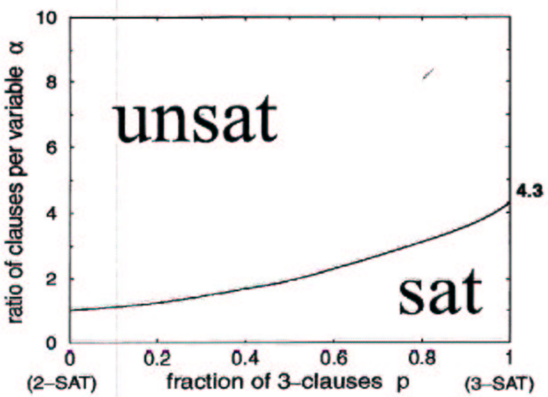
→ $\alpha = \dfrac{\#\ \text{clauses}}{\#\ \text{variables}}$

p = fraction of 3-clauses

→ average clause length : $2+p$

→ SAT / UNSAT transition depends on p :

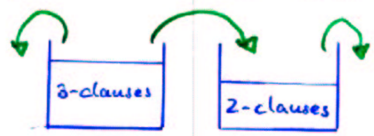
phase diagram of 2+p - SAT :



THE FIRST DESCENT INTO THE BACKTRACKING TREE

→ Markov process

→ can be described by the number of 2-clauses and 3-clauses



→ ordinary differential equations

$$\frac{d\alpha}{dt} = F_\alpha(\alpha, p, t)$$

$$\frac{dp}{dt} = F_p(\alpha, p, t)$$

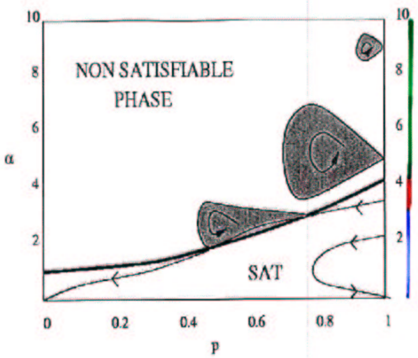
+ initial conditions : $\alpha(0) = \alpha$, $p(0) = 1$

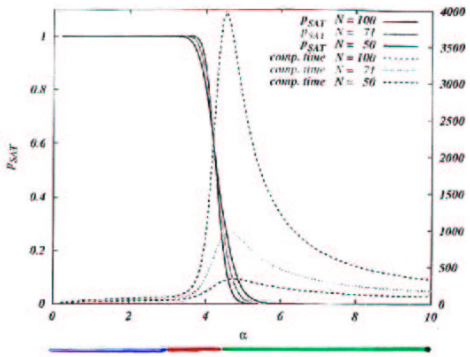
Franco, Chao '90 ; Frieze, Suen '96 :

- $\alpha < 3.003$: solution found with prob. 1
  - → linear solution time
  - → easy SAT phase
- $\alpha > 3.003$: contradiction found with prob. 1

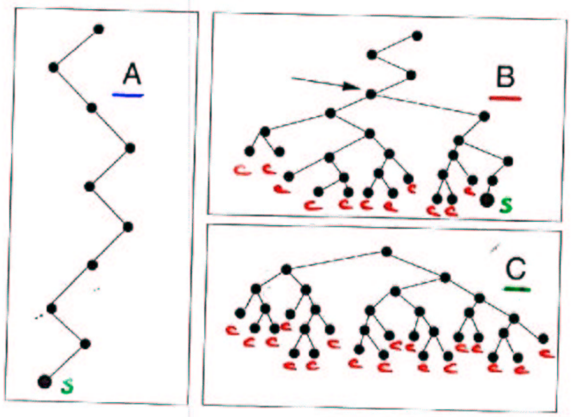
→ method to find lower bounds on α_c

DAVIS-PUTNAM WITH BACKTRACKING





A	B	C
SAT LIN	SAT EXP	UNSAT EXP



peak complexity : $t \sim 1.062^N$ (Cocco, Monasson '01)

$\alpha \gg 1$: $t \sim 2^{0.292 N/\alpha}$ (Beame et al. '98)

OUTLOOK

- Other problems :
  - number partitioning (Mertens '99, Borgs et al. '03)
  - vertex cover (Hartmann, M.W. '00 - '02, Montanari, Zecchina '02)
  - coloring (Mulet, Pagnani, MW, Zecchina '02)
- other algorithms :
  - restart algorithms (Cocco, Monasson '02, Montanari, Zecchina '02)
  - stochastic local search (Barthel, Hartmann, MW '03, Semerjian, Monasson '03)