PHASE TRANSITION

SAT–problem

The first problem that was shown to be NP–complete (Cook, 1971).

The problem: Given a Boolean formula in Conjunctive Normal Form (CNF), does there exist a satisfying assignment?

3–SAT NP–complete

n–SAT NP–complete

2 SAT $\in$ *P*–class

Problem instances of many NP–complete problems can be partitioned into two classes:

- instances of problems that are provably easy to solve, i.e. they have polynomial time solutions,
- instances that are hard to solve, i.e. they have no polytime solutions.

Thousands of NP–complete problems are known, all can be encoded as SAT problems (see the examples below).

Example 1

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 wants to snub either Romania or Peru or both. Is there a guest list that will satisfy the whims of the entire royal family?

The problem is represented by the formula:

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

To satisfy the formula, $p = true,\ q = true,\ r = false$ or

$p = false,\ q = false,\ r = true.$

Example 2: Protein folding

Proteins: biochemical molecules that make up cells, organs, organisms.

Ribosome: read codes; link the amino acids to form proteins.

Human body: can build 50000 proteins.

Every protein has a special 3D structure $\rightarrow$ folding process.

Folding: determines the shape and the function of the protein.

Folding → SAT → *NP*–complete

Not known why do they fold, and why on the way as they do. The Folding problem is not solved. Folding problem lies at the heart of several diseases:

Alzheimer's

Mad cow

Crantzfeld–Jacob

Example 3: Planning and scheduling

- Planning is the process of selecting and sequencing processes such that they achieve one or more goals and satisfy a set of domain constraints.
- Scheduling is the process of selecting among alternative plans and assigning resources and time to the set of processes in the plan.

The *Planning and Scheduling* problem exists in manufacturing systems, in publishing houses, universities, hospitals, airports, transportation companies, etc. It is NP-complete, i.e., it is impossible to find an optimal solution without the use of an essentially enumerative algorithm and the computation time increases exponentially with problem size.

Example 4: Computer chip verification

Because of their extreme complexity, computer chips (such as microprocessors) have been a particular target of practical formal verification research. Circuits can be verified by comparing their implementation (called gate-level) to their formal specifications. Such equivalence checking is widely used. This comparison problem solves the satisfiability problem so it is NP-complete. Essentially, it checks all possible input configurations, so as a new "gate" is added, allowing an additional bit to enter the chip's circuit, the number of possible input configurations increases by a magnitude of 2 (exponential time).

Example 5: Fiber optics routing

Optical networking - sending data or voice traffic as light signals over fiber optics cables - is already the technology of choice in long-distance transport. Long-haul optical networks can carry terabits of data, divided into discrete “channels” based on light wavelengths, across long distances with startling speed and signal integrity.

SOLUTION

P – problem

π – instance of the problem

$\exists\ \pi$ of P which cannot be solved in polynomial time $\Rightarrow P \in NP$.

$P \in NP \ \not\Rightarrow \forall \pi$ cannot be solved in polynomial time.

Typically: *NP* problems are solved by trials, heuristics: branch and bound

backtracking

Readings:

1. Marc Mezard, Giorgio Parisi, Riccardo Zecchina. ANALYTIC AND ALGORITHMIC SOLUTION OF RANDOM SATISFIABILITY PROBLEMS. Science, June 27, 2002.
2. B. Hayes. *Can't get no satisfaction*. American Scientist, 85(2):108-112, 1997.

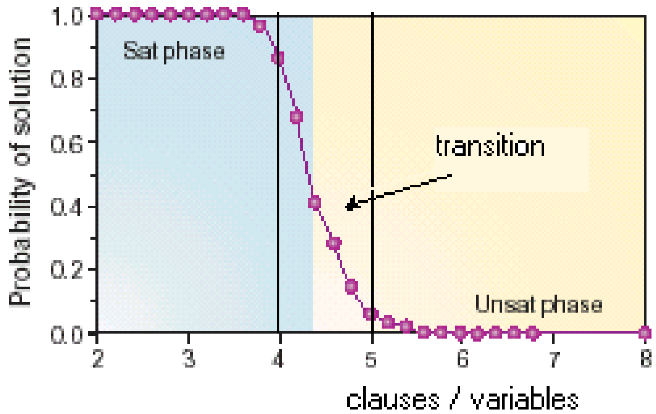

Figure 1 Phase change in 3–SAT. Satisfiable phase is on the left; unsatisfiable phase on the right.

Graph connectivity

Random graph

Let G be a graph over N nodes and E edges. G is a *random graph* if G is formed by selecting E edges uniformly without replacement from the C^2_N possible edges.

Erdős, Rényi:

If $|E| < \dfrac{N \log N}{2} \Rightarrow$ probability(G connected) $\rightarrow 0,\ N \rightarrow \infty$

If $|E| > \frac{N \log N}{2} \Rightarrow$ probability(G connected) $\rightarrow 1$, $N \rightarrow \infty$

Frank, Martel:

Frank and Martel checked the results of Erdős and Rényi for moderate sized graphs. They generated random graphs with different number of nodes *N*. For every *N* they generated 1000 graphs at random. They plotted the proportion of 1000 graphs which were connected against $\frac{|E|}{N \log N}$ for various *E*. They noted that there is a point where all curves intersected each other. The point is at $\frac{|E|}{N \log N} \approx 0.55$. This matches the results of Erdős and Rényi.

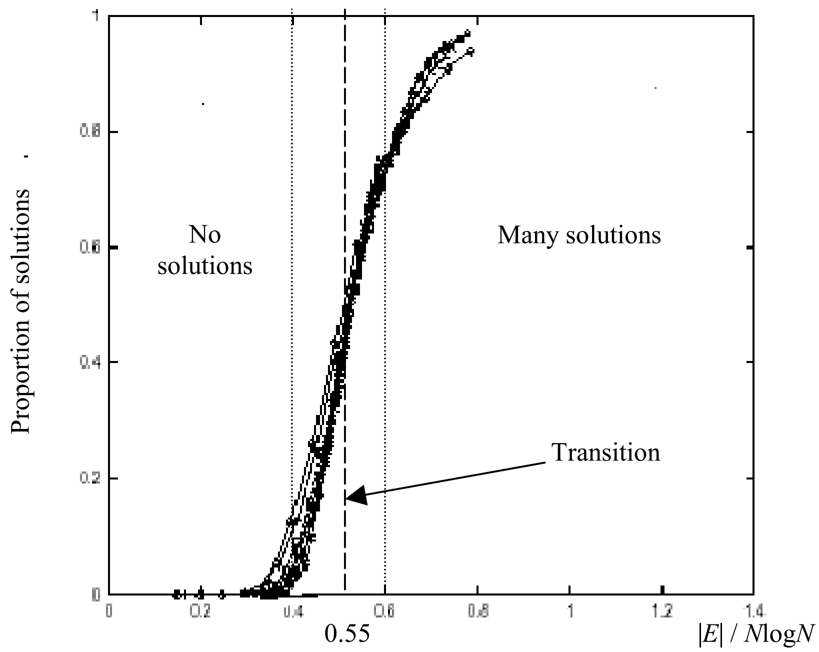

Figure 2 Graph connectivity phase transition

The Phase transition for Hamiltonian Cycle

The *Hamiltonian Cycle Problem* is to decide whether or not there is a Hamiltonian Cycle in a given graph.

<u>Pósa (1976):</u>

If E = $cN \log N$, $\Rightarrow$ probability(G has a Hamiltonian cycle) $\rightarrow 1$, if $N \rightarrow \infty$ and c is sufficiently large.

The proportion of graphs, which contain Hamiltonian Cycles, is plotted against $\frac{|E|}{N \log N}$ for various E. Examination of the curves indicates that the crossover point occur at $\frac{|E|}{N \log N} \approx 0.7$.

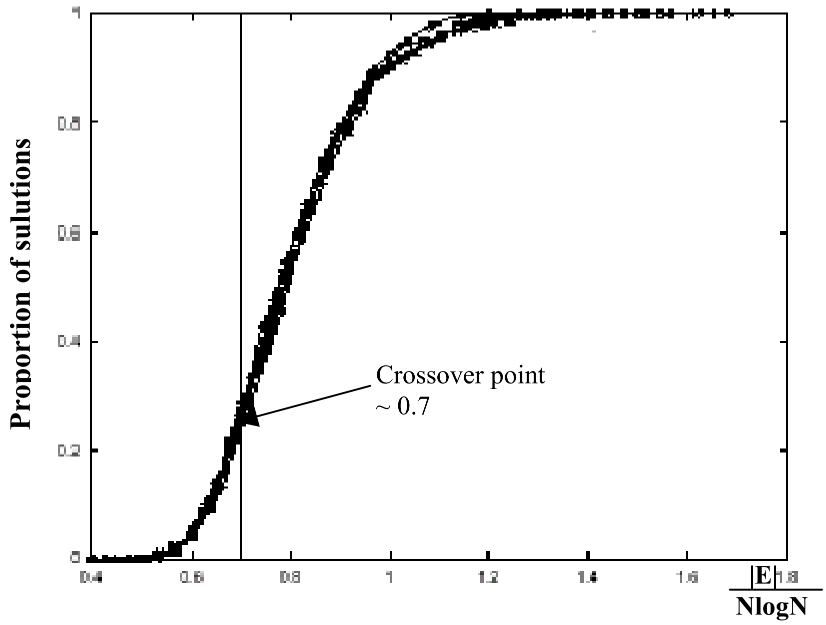

Figure 3 The Hamiltonian Cycle phase transition

A ***phase transition*** is a place where a system undergoes a sudden change of state. In *NP*-Hard problems we are interested in the place where randomly generated problem instances transit from the state of "most instances have solutions" to "most instances have no solution". In most cases there is a problem size independent ***constraint parameter*** which indicates how constrained the problem is.

Phase transitions are typically shown by plotting the proportion of solvable problems with respect to the constraint parameter. A characteristic of phase transitions is that as the problem size increases, the curves become sharper; that is, when plotted against a constraint parameter, the transition occurs in a shorter interval of this parameter. In many cases a constraint parameter can be found such that all of the curves for different problem sizes intersect in the same place; this is called the ***crossover point***.

Constraint parameter *K* in *HC* problem for random graph *G*:

$K \overset{def}{=} 1 - \frac{\log(SOL)}{\log|S|}$, *SOL* = number of solutions, |*S*| = size of the problem

ς: probability of a randomly selected circuit having every edge in *G*.

e: number of edges.

$C_N^2 = \frac{N(N-1)}{2}$: number of possible edges.

Probability that the first edge of a circuit belongs to $G = \frac{2e}{N(N-1)}$

e – 1: the probability for the next edge to belong to $G = \frac{e-1}{\frac{N(N-1)}{2} - 1}$

$$\varsigma = \prod_{i=0}^{N-1} \frac{e-i}{\frac{N(N-1)}{2} - i}$$

Number of distinct potential circuits: $\frac{(N-1)!}{2}$

$SOL = \varsigma \frac{(N-1)!}{2}$ $\qquad S = \frac{(N-1)!}{2}$

$$K = 1 - \frac{\log(SOL)}{\log|S|}$$

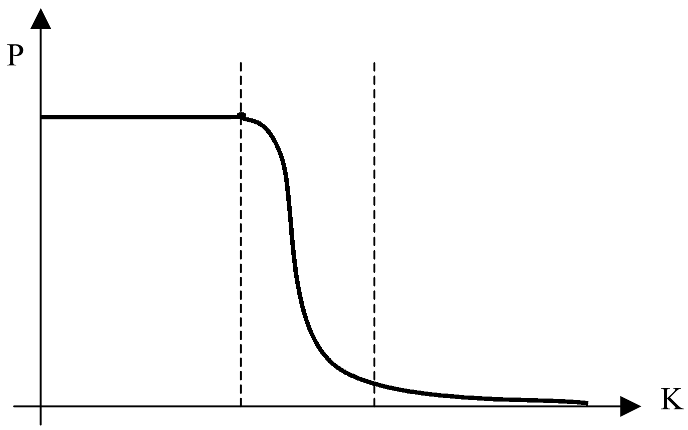

Figure 4 Typical plot for phase transition: sudden change from one state to another.

Catastrophy Theory

It is concerned with using continuous functions to describe phase transitions.

René Thorn: 7 catastrophy types → 2nd type: spike catastrophy <=> phase transition

Spike catastrophy

$x^3 + ax + b$

Phase transition