Phase Transitions in Satisfiability and Coloring

Rigorously Proved Results

Lefteris M. Kirousis[a]

Department of Computer Engineering and Informatics, University of Patras and

RA Computer Technology Institute

[a]Research partially supported by the Research and Academic Computer Technology Institute; by the Future and Emerging Technologies programme of the EU under contract 001907 *"Dynamically Evolving, Large-Scale Information Systems (DELIS);"* and by the European Social Fund (ESF), Operational Program for Educational and Vacational Training II *(EPEAEK II)*, particularly the programme *Pythagoras*

The satisfiability problem (SAT)

- Boolean variables $x_1, x_2, \ldots, x_n$.
- Literals: Boolean variables or negations of them.
- Clause: Disjunction of literals $(x_1 \wedge x_2 \wedge \neg x_3)$.
- Formula: Conjunction of clauses (Conjunctive Normal Form – CNF).

The satisfiability problem (SAT)

- Boolean variables $x_1, x_2, \ldots, x_n$.
- Literals: Boolean variables or negations of them.
- Clause: Disjunction of literals $(x_1 \wedge x_2 \wedge \neg x_3)$.
- Formula: Conjunction of clauses (Conjunctive Normal Form – CNF).

- Satisfying truth assignment: One that has at least one true literal in every clause.
- Satisfiability Problem (SAT): Given a formula ϕ, is it satisfiable?
- k-SAT: Every clause contains exactly k literals.

The DPLL procedure for SAT

- At every step, set a literal to either “true” or “false.”
- Clauses where a “true” occurrence appears are deleted.
- “False” occurrences of literals are deleted from all clauses they occur.
- If the current formula is empty, return “yes”; if it contains an empty clause, return “no”.

The DPLL procedure for SAT

- At every step, set a literal to either “true” or “false.”
- Clauses where a “true” occurrence appears are deleted.
- “False” occurrences of literals are deleted from all clauses they occur.
- If the current formula is empty, return “yes”; if it contains an empty clause, return “no”.

- **Unit clause Rule:** Always set “true” clauses with a single literal.
- **Splitting Rule:** if no unit clauses exist, first try setting to true a suitably selected literal. In case of failure, backtrack and try setting it to false.

The k-colorability problem (k-COL)

- Given a graph $G = (V, E)$ decide whether its vertices can be colored with at most k colors so that no adjacent vertices get the same color.

The List Coloring algorithm

Input: A graph G together with a list of possible k colors for each of its vertices.

- At every step, choose a color from a list and assign it to its vertex.
- Delete this vertex and also delete the selected color from neighboring vertices.
- If the graph becomes empty, return “yes”; if a vertex with an empty list appears, return “no”.

The List Coloring algorithm

Input: A graph G together with a list of possible k colors for each of its vertices.

- At every step, choose a color from a list and assign it to its vertex.
- Delete this vertex and also delete the selected color from neighboring vertices.
- If the graph becomes empty, return “yes”; if a vertex with an empty list appears, return “no”.

- Vertices with one element in their list are given priority.
- If one color from a list fails, backtrack and try the others.

Average complexity

- Having an “appropriate” distribution (model of randomness) is crucial: there are results stating that SAT is “easy” on average, but they assume ad hoc distributions of the instances.
- Example of a distribution that gives rise to easy instances of SAT: Each literal gets into each clause independently with probability p (the length of a clause is a random variable).
- We will consider random instances of k-SAT, for fixed k.

Two models for random k-SAT

- $G_{n,p}$: Each clause is independently selected with probability p to be included in the formula (the number of clauses, i.e. the length of the formula, is a random variable).
- $G_{n,m}$: Exactly m clauses are uniformly, independently and with replacement selected to be included in the formula.

Density

Density (ratio of number of clauses to number of variables ratio):

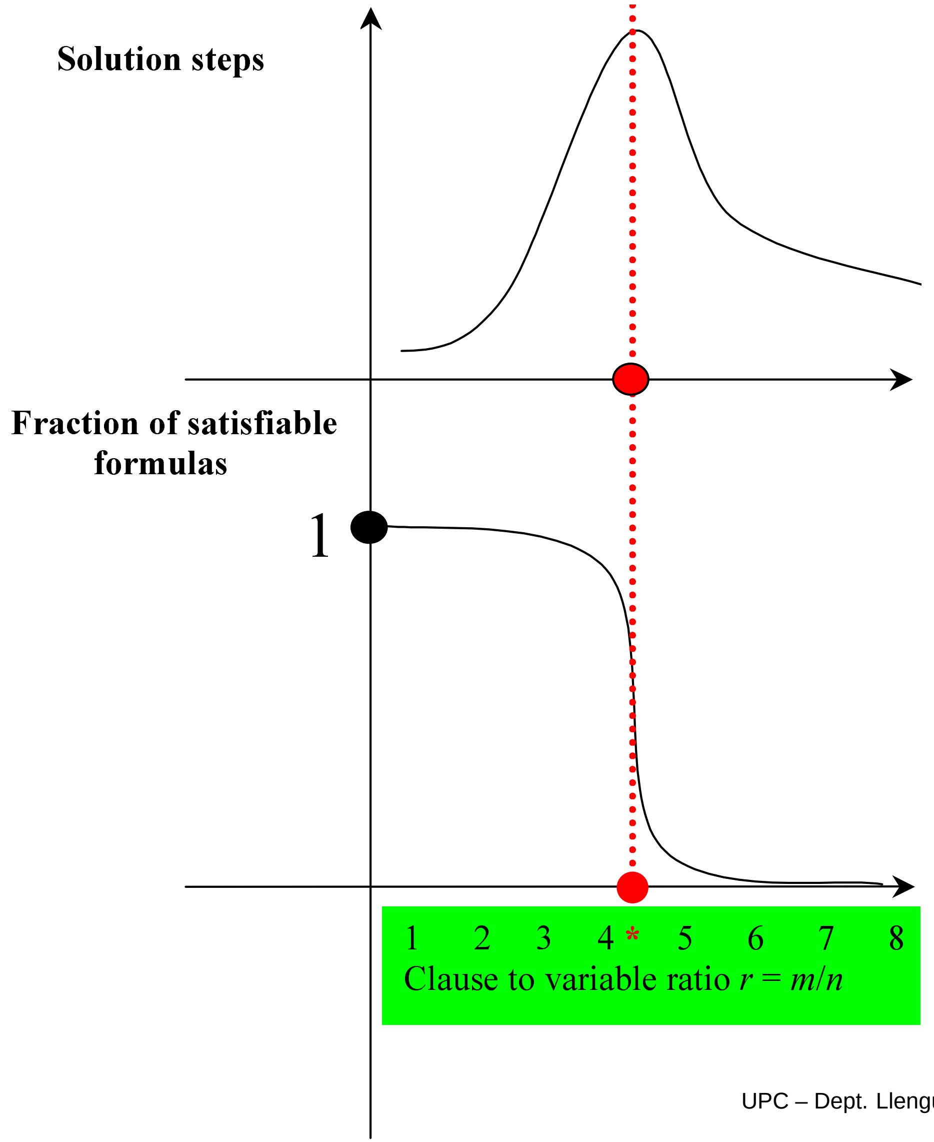
- In $G_{n,m} : r = \frac{m}{n}$.
- In $G_{n,p} : r = \frac{8p\binom{n}{3}}{n}$.

Density is assumed to be constant.

If $p = \frac{3r}{4n^2}$, then in $G_{n,p}$ expected length (number of clauses) of a formula is rn.

Phase transition — experiments

[Mitchell et al. 1992] and other groups:



Phase transition II

General experimental observation: for each fixed k (amenable to experimentation), there is a constant r_k^* such that:

- If r is "away" from r_k^*, then the number of DPLL calls is small, while if r is close to r_k^*, it is large.
- If $r < r_k^*$, then the formula is a.s. satisfiable, while if $r > r_k^*$ the formula is a.s. unsatisfiable ("a.s." means with probability approaching 1 asymptotically with the number of variables).

Analytic (but non-rigorous) verification of the later "decision" threshold (cavity and replica symmetry breaking methods of Statistical Physics) in papers by Mertens, Mézard, Parisi, and Zecchina.

Phase transition III

For $k = 3$ experiments and analytic heuristics suggest that $r_3^* \simeq 4.2$.

Also, it has been rigorously settled that for 2-SAT: $r_2^* = 1$ [de la Vega, Goerdt, Chvátal & Reed].

Open problem for 3-SAT: (i) Formally prove the existence of r_3^* and (ii) formally compute its value.

Friedgut's theorem

Theorem Friedgut '97 *There is a sequence* $r^*(n)$ *such that* $\forall\epsilon:$

- $\Pr[\phi_{r^*(n)-\epsilon}$ is SAT$] \to 1$.
- $\Pr[\phi_{r^*(n)+\epsilon}$ is SAT$] \to 0$.

In other words, the transition interval can be made arbitrarily thin (sharp transition).

Friedgut's theorem

Theorem Friedgut '97 *There is a sequence* $r^*(n)$ *such that* $\forall \epsilon :$

- $\Pr[\phi_{r^*(n)-\epsilon} \text{ is SAT}] \to 1$.
- $\Pr[\phi_{r^*(n)+\epsilon} \text{ is SAT}] \to 0$.

In other words, the transition interval can be made arbitrarily thin (sharp transition).

- **Still open question:** Does $r^*(n)$ *converge*? If yes, to what value?

Corollary *If* $\Pr[\phi \text{ is SAT}] > \epsilon$ *finally for all* n*, then* $\Pr[\phi \text{ is SAT}] \to 1.$

Upper bounds for 3-SAT

- First: 5.19 (circa 1983) – observed by several researchers independently – *Markov's inequality*
- Current best: 4.506 Dubois, Boukhad, & Mandler (1999) – *typical formulas*
- Several *rigorous* (as opposed to experimental) intermediate results between 1983 and 1999.

The basic upper bound technique

Let ϕ be a random formula and $S(\phi)$ the random class of its satisfying truth assignments (s.t.a.). Then

$$\Pr[\phi \text{ is satisfiable}] = \Pr[|S(\phi)| \geq 1] \leq \mathbb{E}(|S(\phi)|).$$

Since $\mathbb{E}(|S(\phi)|) = \left(2(\frac{7}{8})^r\right)^n$ we obtain $r^* < 5.19$.

But why experiments suggest $r^* \simeq 4.2$?

Rare formulas with *many* s.t.a. contribute *too much* to the expectation $\mathbb{E}(|S(\phi)|)$ **(Lottery paradox)**.

How to improve the upper bound

Thin $S(\phi)$ (the set of all satisfying truth assignments) so that the “unrealistic" expectation of the cardinality of $S(\phi)$, due to the Lottery paradox, gets smaller.

Examine not the whole space of possible formulas, but a subset of it such that the formulas of the subset are:

- *typical* and
- less prone to “unrealistic” expectation of the number of satisfying truth assignments (Dubois et al.).

Algorithms for lower bounds

- Let r_l denote the lower bound for r^* that we try to compute.
- Consider DPLL with a suitable heuristic for choosing a literal in the splitting step.
- Prove that for $r < r_l$, the heuristic a.s. succeeds.
- Analyze DPLL without backtracking (i.e. if failure is reported before an application of the splitting rule, this failure is considered permanent).
- The more sophisticated a heuristic is, the more difficult or impossible it's probabilistic analysis is.

Algorithmic lower bounds for 3-SAT

- First: 2.99 Chao and Franco (1986). They introduced many basic analysis techniques
- Current best: 3.52 Kaporis, Kirousis and Lalas (2001).
- Progress in the case of lower bounds is much slower and the techniques more involved (compared with upper bounds).
- Also, there is strong experimental evidence that the algorithmic approach cannot overcome a barrier around 3.9, which is considerably below the putative threshold value (4.25). Why?

The Y2K problem

By the first moment method it is easy to show the the putative threshold for k-SAT is at most $(\ln 2)2^k$.

An easy algorithmic bound [Chvátal and Reed, 1992] shows the the threshold is $\Omega((1/k)2^k)$ (as k grows large).

Many attempts to algorithmically close the gap from below failed. Why?

PS The term Y2K refers to the question "Why 2^k?" The term was coined in September **1999** during a workshop in Trieste on NP-completeness and phase transitions.

The geometry of truth assignments

Consider the space of all truth assignments with the Hamming distance. Then:

- For densities below certain value, all satisfying truth assignments from a unique cluster.
- As the density increases, the clusters break down into exponentially many.
- Moreover, exponentially many clusters correspond to a local minimum (with respect to the number of unsatisfiable clauses).

The geometry of assignments cont'ed

- As a consequence, local search algorithms are not expected to produce results beyond the breaking down of the unique cluster.
- Hard satisfiability region (densities below but close to the threshold).

Results by Biroli and Monasson; Mézard, Parisi and Zecchina and others, e.g. recently by Mézard, Palassini, and Rivoire.

Recent rigorous results: Achlioptas and Ricci-Tersenghi; Mézard, Mora, and Zecchina (2005).

Survey Propagation

- An algorithm for k-SAT (for small values of k) that behaves very nicely up to the threshold.
- Difficult, or perhaps impossible, to rigorously analyze Survey Propagation.
- Inspired from the Statistical Mechanics approach to phase transition phenomena like SAT (reminiscent of Belief Propagation).

Seminal exposition paper: Mézard, Parisi, and Zecchina, Science, 2002.

What is the way out?

Conclusion: Non-algorithmic approaches for lower bounds should be tried.

The second moment method

Let X be a non-negative variable (usually a *counting* variable) that depends on n.

- Lottery Paradox: As n grows large $\mathbb{E}(X)$ may also grow large, but yet $\Pr[X > 0]$ may approach zero.
- However if $\mathbb{E}(X^2)$ (the *second moment*. essentially the *variance*) does not approach infinity too fast compared to $\mathbb{E}(X)$, then it may turn that $\Pr[X > 0]$ stays away from zero. Formally:

$$\Pr[X > 0] \geq \frac{(\mathbb{E}(X))^2}{\mathbb{E}(X^2)}.$$

First used for k-SAT by Achlioptas and Moore, 2002.

The solution of the Y2K problem

- Try the Second Moment Method, by defining X to be the number of satisfying truth assignments of a random instance of k-SAT.
- Unfortunately the method does not work, because X has a large second moment.
- Achlioptas and Moore, 2002: Count in X the satisfying truth assignments that in each clause falsify at least one literal (in addition to satisfying one). This symmetry requirement reduces the variance.
- They showed that $(\ln 2/2)2^k - O(1)$ is a lower bound to the threshold.
- Achlioptas and Peres, 2003 improved the above to $(\ln 2)2^k - O(k)$.

What about k-COL?

- Łuczak, 1991: The chromatic number of almost all $G_{n,p=d/n}$ graphs, with a given fixed average degree $2d$, may take only two consecutive integer values.
- Unfortunately, until very recently, not much was known about the value of these integers.
- Achlioptas and Naor (2004). Let k_d be the smallest integer k such that $d < k \ln k$ Almost all $G_{n,d/n}$ random graphs have chromatic number either k_d or $k_d + 1$.

k-COL continued

- Method of Proof: Second moment where X counts the number of balanced k-colorings of $G_{n,d/n}$.
- Difficulty: The second moment of X turns out to be a sum of exponential terms. Locating the term with the largest base, which essentially gives the value of the sum, proved out to be a difficult task.

Random regular graphs?

Progress is much slower.

- Achlioptas and Moore (2004): The chromatic number of random regular graphs with n vertices and degree $2d$ (d fixed) may take only the values $k_d, k_d + 1, k_d + 2$, where k_d is again the smallest integer k such that $d < k \ln k$.
- Shi and Wormald (2004): Algorithmic analogous results for values of d up to 10.
- Also, almost all 4-regular graphs have chromatic number 3, and
- almost all 6-regular graphs have chromatic number 4.

5-regular graphs?

- Replica Symmetry Breaking – Survey Propagation (Krząkala, Pagnani, Weigt, 2004): almost all 5-regular graphs have chromatic number 3.
- Also, the solution space of 3-colorings of 5-regular graphs has many clusters. Therefore, algorithmic techniques are expected to be hard.
- Until recently, the only rigorous result for 5-regular graphs is that almost all of them have chromatic number 3 or 4.
- Second Moment: Fails when X counts 3-colorings, even if they are balanced. Not enough symmetry, so the second moment (variance) is large.

Stable 3-colorings

Colorings where each vertex has neighbors of both the other two legal colors.

Balanced: each color is assigned to an equal number of vertices.

Díaz, Grammatikopoulos, Kaporis, Kirousis, Pérez and Sotiropoulos (2005): 5-regular graphs are 3-colorable with positive probability.

Method of proof: Apply second moment to the number of stable, balanced 3-colorings on 5-regular graphs.

Result: A 5-regular graph is 3-colorable with positive probability independent of its size.

“Off the press”

Kemkes and Wormald: Padded up this probability to 1 (asymptotically).

Technique: Using the previous result that:

$$\Pr[X > 0] \geq \frac{(\mathbb{E}(X))^2}{\mathbb{E}(X^2)} \sim \text{constant}.$$

They showed that:

$$\Pr_Y[X > 0] \geq \frac{(\mathbb{E}_Y(X))^2}{\mathbb{E}_Y(X^2)} \sim 1,$$

by conditioning over the number of the cycles of the subgraph that “work against colorability”.