On Local versus Global Satisfiability

[Preliminary Version]

Luca Trevisan*

March 19, 1997

Abstract

We prove an extremal combinatorial result regarding the fraction of satisfiable clauses in boolean CNF formulae enjoying a locally checkable property, thus solving a problem that has been open for several years.

We then generalize the problem to arbitrary constraint satisfaction problems. We prove a tight result even in the generalized case.

1 Introduction

We deal with the notion of k-satisfiable CNF formulae introduced and studied by Lieberherr and Specker [4, 5]. A CNF boolean formula (from now on referred to as *formula*) is k-satisfiable if any subset of k clauses is satisfiable. For any k, let r_k be the largest real (or, better, the supremum of the set of reals) such that in any k-satisfiable set of m clauses, at least $r_k m$ clauses are simultaneously satisfied. Roughly speaking, r_k somewhat shows how local satisfiability implies (fractional) global satisfiability. It has been known that $r_2 = 2/(1+\sqrt{5}) > .618$ [4] (the inverse of the golden ratio), that $r_3 = 2/3$ [5], and that $\lim_{k\rightarrow\infty} r_k \leq 3/4$ [3]. Yannakakis [7] has given simplified proofs of the bounds $r_2 \geq 2/(1+\sqrt{5})$ and $r_3 \geq 2/3$ using the probabilistic method.

To the best of our knowledge, it was still an open question to determine the exact value of $\lim_{k\rightarrow\infty} r_k$.

Our Results

We prove that $\lim_{k\rightarrow\infty} r_k = 3/4$. Our proof is constructive: for any $r < 3/4$ we show that a k exists such that given a k-satisfiable formula we can find a probability distribution over its variables in such a way that any clause is satisfied with probability at least r. It thus follows that an assignment satisfying at least a fraction r of clauses must exist. It can even be found in linear time using the greedy algorithm in [7].

We then consider a similar question for general Constraint Satisfaction Problems (CSP). An instance of a CSP is a set of boolean predicates (or *constraints*) over boolean variables. For a fixed integer h, the hCSP is the restriction of CSP where the arity of the constraints is at most h. Note that if a hCSP instance does not contain identically false constraints, then the random assignment where each variable is true with probability $1/2$ will satisfy at least a fraction 2^{-h} of the constraints. We say that a CSP instance is k-satisfiable if any subset of k constraints is satisfiable. For any

*trevisan@cui.unige.ch. Centre Universitaire d'Informatique, Université de Genève, Rue Général-Dufour 24, CH-1211, Genève, Switzerland. Part of this work was done while the author was visiting the IBM T.J. Watson Research Laboratories, Yorktown Heights, NY, in October and November 1995.

integers h and k, we define $r_k^{(h)}$ as the supremum of the reals such that for any k-satisfiable instance of hCSP with m constraints, at least $r_k^{(h)}m$ are satisfiable.

We prove $\lim_{k\to\infty} r_k^{(h)} = 2^{1-h}$. For the lower bound, it will be easy to use the probabilistic method to obtain $r_{h+1}^{(h)} \geq 2^{1-h}$. In order to prove the upper bound $r_k^{(h)} \leq 2^{1-h}$ for all k we will need a construction of hypergraphs that generalizes the known construction of graphs with small maximum cut and large girth [1].

Preliminary Definitions

A *CNF boolean formula* (or, simply, a *formula*) is a set $\{C_1, \ldots, C_m\}$ of *disjunctive clauses* over a set of *variables* $X = \{x_1, \ldots, x_n\}$. A disjunctive clause is a disjunction of *literals* where each literal is either a variable x_i or a *negated* variable $\neg x_i$. An *assignment* for ϕ is a mapping $\tau : X \to \{\mathsf{true}, \mathsf{false}\}$ that associates a *truth value* with any variable. If l is a literal, then we say that τ *satisfies* l if either $l = x$ and $\tau(x) = \mathsf{true}$ or $l = \neg x$ and $\tau(x) = \mathsf{false}$. If $C = l_1 \vee \ldots \vee l_h$ is a clause, we say that τ satisfies C if τ satisfies l_j for some $j \in \{1, \ldots, h\}$. A formula ϕ is *k-satisfiable* [4] if any subset of k clauses of ϕ is satisfiable.

An istance of CSP is set $\{C_1, \ldots, C_m\}$ of *constraints* over a set of *variables* $X = \{x_1, \ldots, x_n\}$. A constraint is a boolean predicate applied to variables from X. An instance of hCSP (where h is an integer) is an instance of CSP where the arity of all the predicates is at most h. We define assignments, satisfiability, and k-satisfiability as for formulae, with "clauses" replaced by "constraints" in the definitions.

A *random* assignment is a probability distribution over all the assignment. We will restrict ourselves to random assignments where each variable is assigned true with a certain probability, independently of the assignments to the other variables (it would actually suffice bounded independence). Thus a random assignment τ_R is entirely specified by the probabilities $\{p_x\}_{x\in X}$, where $\mathbf{Pr}[\tau_R(x) = \mathsf{true}] = p_x$. To save notation, we will write $\mathbf{Pr}[x = \mathsf{true}]$ in place of $\mathbf{Pr}[\tau_R(x) = \mathsf{true}]$ when the random assignment is clear from the context.

2 The CNF result

2.1 Yannakakis' Argument and How to Extend it: an Informal Account

In order to present the main ideas underlying our proof, let us first recall Yannakakis' proof that $r_3 \geq 2/3$. Given a 3-satisfiable formula he shows how to find a probability distribution over the variables that satisfies all clauses with probability at least $2/3$. If a literal l occurs in a unary clause, then we set $\mathbf{Pr}[l = \mathsf{true}] = 2/3$. Note that this definition is consistent since it is impossible to have the clauses (x) and $(\neg x)$ in the same 3-satisfiable formula. To all the other variables (the ones that do not occur in unary clauses), if any, we give value true with probability $1/2$. Ternary clauses, or longer ones, are satisfied with probability at least $1-(2/3)^3 = .7037\ldots > 2/3$; for longer clauses probabilities are even better. It remains to consider binary clauses. If at least one of the literals in a binary clause is true with probability at least $1/2$, then the probability that the clause be satisfied is at least $1 - (2/3)1/2 = 2/3$. The only bad case happens when both literals are true only with probability $1/3$, but this is impossible because it would mean that the formula contains clauses $(l_1), (l_2), (\neg l_1 \vee \neg l_2)$ which contradicts the fact that it is 3-satisfiable.

When we want to achieve the same construction with an arbitrary $r < 3/4$ in place of $2/3$ we run into some troubles. Let us try with $r = .74$. Literals occurring in unary clauses must be true with probability $.74$. If l occurs in a unary clause, and we have the clause $\neg l \vee x$, then x must be true with probability at least $1-(1-r)/r = .6486\ldots$ Then we have to consider literals occurring with $\neg x$ in a binary clause: they have to be true with probability at least $.5991\ldots$ There are three more cases to be considered (probabilities will be, respectively, $0.566\ldots$, $0.5406\ldots$, and $0.5191\ldots$) And we still have to make sure that we are not introducing any inconsistency, and we have to deal with ternary and 4-ary clauses (clauses with 5 or more literal are satisfied with probability at least $1-(.74)^5 > .74$.)

The above discussion leaves us with the idea that the range of values for the probabilities of the literals should be $p_1 = r$, $p_2 = 1-(1-r)/r$, $p_3 = 1-(1-r)/p_2$, $\ldots$ $p_k = 1-(1-r)/p_{k-1}$. It is conforting that this sequence will eventually go below 1/2, where it can be stopped (Lemma 2).

We also note that, when we want to achieve a ratio close to 3/4, the numbers of cases to be considered explodes, and that a uniform method to deal with them has to be found.

In order to attribute probabilities to the literals in a uniform way, we introduce the idea of *ranking* them according to the depth of *proofs* of the literals in a simple propositional proof system, whose axioms are the clauses of the formula. This gives at the same time a uniform way to deal with clauses of different lenght and a simple method to show that the assignment of probabilities is consistent.

2.2 The Actual Proof

The following definition gives the values that we will use in the probability distribution.

Definition 1 *For any real $r \neq 0$, we define the sequence $\{a_i^r\}_{i\geq 1}$ as follows:*

- $a_1^r = r$;
- $a_{i+1}^r = 1-(1-r)/a_i^r$.

If we start from a number $r < 3/4$, the sequence eventually goes below 1/2.

Lemma 2 *For any r such that $1/2 < r < 3/4$, a $h(r)$ exists such that $a_{h(r)}^r < .5$*

PROOF: Assume not. Note that if $a_i^r > 0$, then $a_{i+1}^r < a_i^r$, as can be easily proved by induction. Then we have a monotonically decreasing sequence that is lower bounded by 0.5: such a sequence must have a limit, let it be x. Then x is a real root of the equation

$$x = 1-(1-r)/x,$$

that is,

$$x^2 - x + 1 - r = 0\ .$$

But such an equation has no real root when $1-4(1-r) < 0$, that is when $r < 3/4$. □

The following definition allows to *rank* literals and will be used to assign to each of them the right probability.

Definition 3 (Provability) *Given a CNF formula ϕ,*

- *If $(l) \in \phi$ then l is 1-provable in ϕ.*
- *If $(l_1 \vee \ldots l_h) \in \phi$ and $\neg l_j$ is i_j-provable in ϕ for $j = 1,\ldots,h-1$, then l_h is $(1+\max\{i_1,\ldots,i_{h-1}\})$-provable in ϕ.*

A literal is exactly i-provable *in ϕ if i is the smallest integer such that it is i-provable in ϕ.*

Lemma 4 *Let ϕ be a formula with clauses of lenght at most 4. If x is i-provable in ϕ and $\neg x$ is j-provable in ϕ, then ϕ is not $(3^{i+1} + 3^{j+1} - 2)$-satisfiable.*

PROOF: Simple induction shows that when a literal l is i-provable in ϕ, then a set S_l of at most $3^{i+1} - 1$ clauses of ϕ exists such that any assignment that satisfies all the clauses in S_l must also satisfy l. Then, the set $S_x \cup S_{\neg x}$ has at most $3^{i+1} + 3^{j+1} - 2$ clauses, and no assignment can satisfy all of them. □

The next theorem is clearly a sufficient condition to have $\lim_{k\to\infty} r_k \geq 3/4$.

Theorem 5 *For any r such that $1/2 < r < 3/4$ a k exists (depending on r) such that for any k-satisfiable formula ϕ we can find in polynomial time a probability distribution over the variables in such a way that any clause is satisfied with probability at least r.*

PROOF: For any variable x, the probability p_x of x to be true will be a rational between r and $1 - r$, and, in particular, between $1/4$ and $3/4$. This implies that any 5-ary clause is satisfied with probability at least $1 - (3/4)^5 > 3/4$. Thus we only have to care about unary, binary, ternary and 4-ary clauses. Let us fix $r < 3/4$ and let $k = 2 \cdot 3^{h(r)+1} - 1$. Let ϕ be a k-satisfiable formula, and let ϕ_4 be the subset of clauses of ϕ of lenght at most 4. Observe that if some literal is i-provable in ϕ_4 for some $i \leq h(r)$, then it is not possible that its complement is j-provable in ϕ_4 for some $j \leq h(r)$.

We shall use the values $a_1^r, \ldots, a_{h(r)-1}^r, 0.5$ in our probability distribution. Let $p_i = a_i^r$ for $i = 1, \ldots, h(r) - 1$ and $p_{h(r)} = 1/2$. The probability distribution is as follows.

$$\mathbf{Pr}[x = \mathsf{true}] = \begin{cases} p_i & \text{if } x \text{ is exactly } i\text{-provable in } \phi_4, \text{ for } i \leq h(r) - 1 \\ 1 - p_i & \text{if } \neg x \text{ is exactly } i\text{-provable in } \phi_4, \text{ for } i \leq h(r) - 1 \\ \frac{1}{2} & \text{otherwise} \end{cases}$$

It should be clear that the definition above is consistent. Recall that the sequence $p_1, \ldots, p_{h(r)}$ is decreasing. So if a variable x is exactly i-provable for some $i < h(r)$, the smaller is i, the larger is $\mathbf{Pr}[x = \mathsf{true}]$.

Claim 6 *Under the probability distribution above, any clause of ϕ is false with probability at most $1 - r$.*

PROOF: The statement is easy for unary clauses and for clauses with five or more literals.

Let $C = (l_1 \vee \ldots \vee l_h)$ be a clause with two or more literals; we assume $\mathbf{Pr}[l_1 = \mathsf{false}] \leq \mathbf{Pr}[l_1 = \mathsf{false}] \leq \ldots \leq \mathbf{Pr}[l_h = \mathsf{false}]$. If $\mathbf{Pr}[l_2 = \mathsf{false}] \leq 1/2$ then also $\mathbf{Pr}[l_1 = \mathsf{false}] \leq 1/2$ and $\mathbf{Pr}[C \text{ is false}] \leq 1/4 < 1 - r$. It remains to consider the case $\mathbf{Pr}[l_2 = \mathsf{false}] > 1/2$. Then $\neg l_2$ is exactly i_2-provable for some $i_2 \leq h(r) - 1$; and also $\neg l_3$ and $\neg l_4$ (if present) are exactly i_3-provable (resp. i_4-provable) for some $i_3 \leq i_2$ (resp. $i_4 \leq i_2$). It follows that l_1 is exactly i_1-provable for some $i_1 \leq i_2 + 1$, and thus $\mathbf{Pr}[l_1 = \mathsf{false}] = 1 - p_{i_1} \leq 1 - a_{i_1} = (1 - r)/a_{i_1 - 1}$[1], while $\mathbf{Pr}[l_2 = \mathsf{false}] = p_{i_2} = a_{i_2} \leq a_{i_1 - 1}$. As a consequence, we have

$$\mathbf{Pr}[C \text{ is false }] \leq \mathbf{Pr}[l_1 = l_2 = \mathsf{false}] \leq 1 - r$$

□

The theorem thus follows. □

[1]Note that if $i_1 = h(r)$ then l_1 will be assigned probability $1/2$ (that is exactly p_{i_1}) not because it is exactly $h(r)$-provable, but because it is not i-provable for $i < h(r)$ and, of course, neither its complement is (so l_1 falls in the "otherwise" part of the definition).

3 Constraint Satisfaction Problems

Lemma 7 *Let ϕ be a $(h+1)$-satisfiable instance of hCSP. Then it is possible to satisfy at least a fraction 2^{1-h} of the constraints.*

Proof: We describe a random assignment that satisfies each constraint with probability at least 2^{1-h}.

We say that a constraint is *conjunctive* if there is only one assignment of its variables that satisfies it. For any variable that occurs in a conjunctive constraint we set it to the value imposed by the constraint. This is consistent (otherwise the instance would not be 2-satisfiable). This partial assignment does not contradict any (non-conjunctive) constraint (otherwise the instance would not be $(h+1)$-satisfiable). We give probability $1/2$ to all the other variables. It is easy to see that any constraint that is not satisfied by the partial assignment is true with probability at least $2/2^h$: indeed, either it is still h-ary and has two or more satisfying assignments, or its arity has been decreased by the partial assignment, and so it is true with probability at least $1/2^{h-1}$. □

Let h, $r < 2^{1-h}$, and k be fixed. We will show how to find a k-satisfiable instance of hCSP such that only a fraction r of its constraints is simultaneously satisfiable.

We will use only one type of constraint, the HYPERCUT^h constraint, defined as follows

$$\text{HYPERCUT}^h(x_1, \ldots, x_{h-1}, y) \equiv (x_1 \neq y) \wedge (x_1 = \cdots = x_{h-1})$$

For $h = 2$ this is the xor constraint, that gives rise to a constraint satisfaction problem that is equivalent to 2-colorability.

For a set ϕ of HYPERCUT^h constraints, if HYPERCUT$^h(x_1, \ldots, x_{h-1}, y) \in \phi$ then we say that, for any $i = 1, \ldots, h-1$, x_i is *adjacent* to y (and that y is adjacent to x_i) in ϕ. A *cycle of lenght l* ($l \geq 3$) is a sequence of variables $x_1, \ldots, x_l$ such that x_l is adjacent to x_1 and x_i is adjacent to x_{i+1} for $i = 1, \ldots, l-1$. The reader should easily convince himself that ϕ is satisfiable if and only if it does not contain a cycle of odd lenght. The next theorem is well known for the case $h = 2$ [1].

Lemma 8 *For any integers k, h, and any $\epsilon > 0$, there exists a family of m HYPERCUT^h constraints such that no more than $(2^{1-h} + \epsilon)m$ are simultaneously satisfiable and any k of them are satisfiable*

Proof: [Sketch] To meet the second requirement we just have to construct an istance without short cycles of odd lenght. The following construction will work for all sufficiently large n. We fix a (small) constant $\delta > 0$ and a (large) constant c such that

$$2^{1-h}(1+\delta)/(1-2\delta) < 2^{1-h} + \epsilon$$

$$2k(2c)^k \leq \delta cn$$

$$c \geq 6 \log e \log \frac{1}{\delta^2} 2^{h-1} \ .$$

Let $m = cn$, and let $s(n) = n\binom{n-1}{h-1}$ be all the possible HYPERCUT^h constraints over the variable set $\{x_1, \ldots, x_n\}$. Fix also we construct a random instance of hCSP by choosing each of the $s(n)$ constraints independently with probability $m/s(n)$. We make the following claims:

1. With probability at least .9, the number of constraints in the random instance is at least $m(1-\delta)$.

2. With probability at least .9, the generated instance is such that any assignment satisfies at most $2^{1-h}(1+\delta)m$ constraints.

3. With probability at least .5, there are at most $2k(2c)^k$ cycles of length $\leq k$ in the generated instance.

With positive probability a random instance will satisfy all the three properties. In particular, there will exist an instance satisfying such properties. By removing from it a constraint for each cycle of length $\leq k$, we obtain a new instance with no cycle of length $\leq k$, $m' \geq m(1-2\delta)$ constraints, and such that no assignment satisfies more than $(2^{1-h}+\epsilon)m'$ constraints. This modified instance proves the lemma.

We now prove the three claims.

1. The average number of constraints is m. By Chernoff bounds, it will be at least $(1-\delta)m$ with probability at least $1-e^{-\delta^2 m/2}$ which is larger than .9 for sufficiently large n.

2. If we fix one the 2^n possible assignments, that gives value true to tn variables, and value false to $(1-t)n$, it will satisfy a randomly chosen constraint with probability

$$t^{h-1}(1-t)+(1-t)^{h-1}t \leq (1/2)^{h-1} \ .$$

   From Chernoff bounds, the probability that, for a random instance, there exists an assignment satisfying more than $m2^{1-h}(1-\delta)$ constraints is at most

$$2^n e^{-\delta^2 2^{1-h} cn/3} \leq 2^{-n} \leq .1$$

   for sufficiently large n.

3. There are $n(n-1)\cdots(n-l+1)$ possible cycles of length l. Thus, there are at most kn^k cycles of length $\leq k$. Two fixed nodes are adjacent with probability at most $2c/n$. Thus the cycle exists with probability at most $(2c/n)^k$. The average is at most $k(2c)^k$; with probability at most .5 the actual number is more than twice the average.

□

Theorem 9 *For any $h \geq 2$, $\lim_{k\to\infty} r_k^{(h)} = 2^{1-h}$.*

4 Conclusions

It is a startling coincidence that 3/4 is the integrality gap of the tighter known linear programming relaxation of MAX SAT [2] and that 2^{1-h} is the integrality gap of the tighter known linear programming relaxation of MAX hCSP [6]. It would be interesting to understand if this fact has some explanation.

References

[1] P. Erdös. On bipartite subgraphs of graphs. *Math. Lapok.*, 18:283–288, 1967.

[2] M. Goemans and D. Williamson. New 3/4-approximation algorithms for the maximum satisfiability problem. *SIAM Journal on Discrete Mathematics*, 7(4):656–666, 1994. Preliminary version in *Proc. of IPCO'93.*

[3] M.A. Huang and K.J. Lieberherr. Implications of forbidden structures for extremal algorithmic problems. *Theoretical Computer Science*, 40:195–210, 1985.

[4] K. Lieberherr and E. Specker. Complexity of partial satisfaction. *Journal of the ACM*, 28(2):411–422, 1981.

[5] K. Lieberherr and E. Specker. Complexity of partial satisfaction II. Technical Report 293, Dept. of EECS, Princeton University, 1982.

[6] L. Trevisan. Positive linear programming, parallel approximation, and PCP's. In *Proceedings of the 4th European Symposium on Algorithms*, pages 62–75. LNCS 1136, Springer-Verlag, 1996.

[7] M. Yannakakis. On the approximation of maximum satisfiability. *Journal of Algorithms*, 17:475–502, 1994. Preliminary version in *Proc. of SODA'92.*