Thresholds of Random k-Sat (2002; Kaporis, Kirousis, Lalas)

A. C. Kaporis, Department of Computer Engineering and Informatics, University of Patras
http://students.ceid.upatras.gr/~kaporis
L. M. Kirousis, Department of Computer Engineering and Informatics, University of Patras
http://lca.ceid.upatras.gr/~kirousis

entry editor: Kazuo Iwama

INDEX TERMS: Algorithmic lower bounds, Satisfiability threshold, Random k-SAT.

1 PROBLEM DEFINITION

Consider n Boolean variables $V = \{x_1, \ldots, x_n\}$ and the corresponding set of $2n$ literals $L = \{x_1, \overline{x}_1 \ldots, x_n, \overline{x}_n\}$. A k-clause is a disjunction of k literals of distinct underlying variables. A random formula $\phi_{n,m}$ in k Conjunctive Normal Form (k-CNF) is the conjunction of m clauses, each selected in a uniformly random and independent way amongst the $2^k\binom{n}{k}$ possible k-clauses on n variables in V. The density r_k of a k-CNF formula $\phi_{n,m}$ is the clauses-to-variables ratio m/n. We say that asymptotically almost all (a.a.a.) formulas $\phi_{n,m}$ of a given density are satisfiable (unsatisfiable) if the ratio of the number of satisfiable (unsatisfiable, respectively) such formulas to the number of all such formulas approaches 1, as $n \to \infty$.

It is conjectured that for each $k \geq 2$ there exists a critical clauses-to-variables ratio r_k^* such that a.a.a. k-CNF formulas with density $r < r_k^*$ ($r > r_k^*$) are satisfiable (unsatisfiable, respectively). So far, the conjecture has been proved only for $k = 2$ [3, 8, 12]. For $k \geq 3$, the conjecture still remains open but is supported by experimental evidence [15] as well as by theoretical, but non-rigorous, work based on Statistical Physics [16]. The value of the putative threshold r_3^* is estimated to be around 4.22. Approximate values of the putative threshold for larger values of k have also been computed.

As far as rigorous results are concerned, Friedgut [11], proved that for each $k \geq 3$ there exists a sequence $r_k^*(n)$ such that for any $\epsilon > 0$, a.a.a. k-CNF formulas $\phi_{n,\lfloor(r_k^*(n)-\epsilon)n\rfloor}$ ($\phi_{n,\lceil(r_k^*(n)+\epsilon)n\rceil}$) are satisfiable (unsatisfiable, respectively). The convergence $\lim_{n\to\infty} r_k^*(n) = r_k^*$ for each $k \geq 3$ is open.

Let now

$$r_k^{*-} = \underline{\lim}_{n\to\infty} r_k^*(n) = \sup\{r_k : \Pr[\phi_{n,\lfloor r_k n\rfloor} \text{is satisfiable} \to 1]\}$$

and

$$r_k^{*+} = \overline{\lim}_{n\to\infty} r_k^*(n) = \inf\{r_k : \Pr[\phi_{n,\lceil r_k n\rceil} \text{is satisfiable} \to 0]\}.$$

Obviously, $r_k^{*-} \leq r_k^* \leq r_k^{*+}$, if r_k^* exists. Bounding from below (from above) r_k^{*-} (r_k^{*+}, respectively) with an as large (as small, respectively) as possible bound has been the subject of intense research work in the past decade.

Upper bounds to r_k^{*+} are computed by counting arguments. To be specific, the standard technique is to compute the expected number of satisfying truth assignments of a random formula with density r_k and find an as small as possible value of r_k for which this expected value approaches zero as n grows large. Then, by Markov's inequality, it follows that for such a value of r_k, a random formula $\phi_{n,\lceil r_k n\rceil}$ is unsatisfiable a.a.a. This argument has been refined in two directions: First by considering not all satisfying truth assignments but a subclass of them with the property that a satisfiable formula always has a satisfying truth assignment in the subclass considered. The restriction to a judiciously chosen such subclass forces the expected value of the number of satisfying truth assignments to get closer to the probability of satisfiability, and thus leads to a better (smaller) upper bound r_k. The catch is that the subclass should be such that the expected value of the number of satisfying truth assignments in the subclass should be computable by the available probabilistic techniques. Second, by making use for the computation of the expected number of satisfying truth assignments of *typical* characteristics of the random formula, i.e. characteristics shared by a.a.a. formulas. Again this often leads to an expected number of satisfying truth assignments that is closer to the probability of satisfiability (non-typical formulas may contribute to the increase of the expected number). Increasingly better upper bounds to r_3^{*+} have been computed using counting arguments as above (see the surveys [6, 14]). Dubois, Boufkhad and Mandler [7] proved $r_3^{*+} < 4.506$. The latter remains the best best upper bound to date.

On the other hand, for fixed and small values of k (especially for $k = 3$) lower bounds to r_k^{*-} are usually computed by algorithmic methods. To be specific, one designs an algorithm that returns a truth assignment to the variables of an input formula and then computes an as large as possible r_k such that the algorithm returns a *satisfying* truth assignment for a.a.a. formulas $\phi_{n,\lfloor r_k n \rfloor}$. Such an r_k is obviously a lower bound to r_k^{*-}. The tradeoff is of course that the simpler the algorithm, the easier to perform the probabilistic analysis of returning a satisfiable truth assignment, but the smaller the r_k's for which a satisfiable truth assignment is returned a.a.a. In this vain, backtrack-free DPLL algorithms [4, 5] of increasing sophistication were rigorously analyzed (see the surveys [1, 10]). At each step of a backtrack-free DPLL algorithm a literal is set to True and then a *reduced* formula is obtained by (i) deleting clauses where this literal appears and by (ii) deleting the negation of this literal from the clauses it appears. At every step at which clauses with a single literal exist (forced steps), the selection of the literal to be set to True is obligatorily made so as a 1-literal clause (1-clause for short) becomes satisfied. Otherwise (free steps), the selection of the literal to be set to True is made according to a heuristic that characterizes the particular DPLL algorithm. A free step is followed by a round of consecutive forced steps. To facilitate the probabilistic analysis of DPLL algorithms, it is assumed that they never backtrack: if the algorithm ever hits a contradiction it stops and reports failure, otherwise it returns a satisfying truth assignment. The best previous lower bound for the satisfiability threshold obtained by such an analysis was $3.26 < r_3^{*-}$ (Achlioptas and Sorkin [2]).

The previously analyzed such algorithms (with the exception of the `Pure Literal` algorithm [9]) at a free step take into account *only* the clause size where the selected literal appears. Due to this limited information exploited on selecting the next literal, the reduced formula in each step of the DPLL algorithm remains random conditional only on the current numbers of 3- and 2-literal clauses and the number of yet unassigned variables. This type of "strong" randomness permits a successful probabilistic analysis of the algorithm in a not very complicated way. However, for $k = 3$ succeeds in showing a.a.a. satisfiability only for densities up to 3.26. Actually, in [2] it is shown that this is the optimal value that can be attained by such algorithms.

2 KEY RESULTS

In the work that this entry refers to [13], a DPLL algorithm is described —and then probabilistically analyzed— that at each free step selects the literal to be set to True taking into account its *degree* (i.e. its number of occurrences) in the current formula.

The first variant of the algorithm is very simple: at each free step set to True a literal τ so as to satisfy the *maximum* number of clauses. Notice that in this greedy variant, τ is chosen irrespectively of the occurrences of the negation $\overline{\tau}$. This algorithm succeeds to return a satisfiable truth assignment for a.a.a. formulas with density $r_3 \leq 3.42$ establishing that $r_3^{*-} \geq 3.42$. Its simplicity, contrasted with the improvement over the previously obtained lower bounds, suggests the importance of analyzing heuristics that take into account degree information of the current formula.

In the second variant, at each free step t, the degree of the negation $\overline{\tau}$ of the literal τ that is set to True at t is also taken into account. The way that the two degrees (of τ and its negation) are taken into account is so as upon the completion of the round of forced steps that follow the free step t, the marginal expected increase of the flow from 2-clauses to 1-clauses per unit of decrease of the flow from 3-clauses to 2-clauses is optimized (minimized). To compute this marginal quantity, we need to know for each i, j the number of literals with degree i whose negation has degree j. This is so because at a forced step, the expected flow from 2-clauses to 1-clauses is equal to the expected degree within 2-clauses of the negation of a literal chosen at random among the *occurrences* of literals in 1-clauses (the flow from 3-clauses to 2 clauses at a forced step is computed similarly). This improved heuristic succeeds to return a satisfying truth assignment for a.a.a. formulas with density $r_3 \leq 3.52$ establishing that $r_3^{*-} \geq 3.52$.

3 OPEN PROBLEMS

The main open problem in the area is to formally show the existence of the threshold r_k^* for all (or at least some) $k \geq 3$. To rigorously compute upper and lower bounds better than the ones mentioned here still attracts some interest. Related results and problems arise in the framework of variants of the satisfiability problem and also the problem of colorability.

4 CROSS REFERENCES

None is reported. Entry editors please feel free to add some.

5 RECOMMENDED READING

[1] D. Achlioptas. Lower bounds for random 3-sat via differential equations. *Theoretical Computer Science*, 265(1-2):159–185, 2001.

[2] D. Achlioptas and G. Sorkin. Optimal myopic algorithms for random 3-sat. In *41st Annual Symposium on Foundations of Computer Science*, pages 590–600. IEEE, 2000.

[3] V. Chvátal and B. Reed. Mick gets some (the odds are on his side). In *33rd Annual Symposium on the Foundation of Computer Science*, pages 620–627. IEEE, 1992.

[4] M. Davis, G. Logemann, and D. Loveland. A machine program for theorem-proving. *Communications of the ACM*, 5:394–397, 1962.

[5] M. Davis and H. Putnam. A computing procedure for quantification theory. *Journal of the Association for Computing Machinery*, 7(4):201–215, 1960.

[6] O. Dubois. Upper bounds on the satisfiability threshold. *Theoretical Computer Science*, 265:187–197, 2001.

[7] O. Dubois, Y. Boufkhad, and J. Mandler. Typical random 3-sat formulae and the satisfiability threshold. In *11th Symposium on Discrete Algorithms*, pages 126–127. ACM-SIAM, 2000.

[8] W. Fernandez de la Vega. On random 2-sat. Technical report, 1992. Manuscript.

[9] J. Franco. Probabilistic analysis of the pure literal heuristic for the satisfiability problem. *Annals of Operations Research*, 1:273–289, 1984.

[10] J. Franco. Results related to threshold phenomena research in satisfiability: Lower bounds. *Theoretical Computer Science*, 265:147–157, 2001.

[11] E. Friedgut. Sharp thresholds of graph properties, and the k-sat problem. *J. AMS*, 12:1017–1054, 1997.

[12] A. Goerdt. A threshold for unsatisfiability. *Journal of Computer and System Sciences*, 33:469–486, 1996.

[13] A. C. Kaporis, L. M. Kirousis, and E. G. Lalas. The probabilistic analysis of a greedy satisfiability algorithm. *Random Structures and Algorithms*, 28(4):444–480, 2006.

[14] L. Kirousis, Y. Stamatiou, and M. Zito. The unsatisfiability threshold conjecture: the techniques behind upper bound improvements. A.G. Percus, G. Istrate, and C. Moore eds., Computational Complexity and Statistical Physics:159–178, 2006.

[15] D. Mitchell, B. Selman, and H. Levesque. Hard and easy distribution of sat problems. In *10th National Conference on Artificial Intelligence*, pages 459–465, 1992.

[16] R. Monasson and R. Zecchina. Statistical mechanics of the random k-sat problem. *Phys. Rev. E*, 56:1357–1361, 1997.