Inapproximability– Some history and some open problems

Johan Håstad
Royal Institute of Technology
S-100 44 Stockholm
SWEDEN
johanh@nada.kth.se

Abstract

The purpose of this talk is to give an overview of the status of some problems in approximability.

1 Introduction

Many natural optimization problems are NP-hard which implies that they are probably hard to solve exactly in the worst case. In practice, however, it is often sufficient to get reasonably good solutions for all (or even most) instances. In this talk we discuss the existence of polynomial time approximation algorithms for some of the basic NP-complete problems. For a maximization problem we say that an algorithm is a C-approximation algorithm if it, for each instance, produces a solution whose objective value is at least OPT/C where OPT is the global optimum. A similar definition applies to minimization problems.

The most basic NP-complete problem is satisfiability of CNF-formulas and probably the most used variant of this is 3Sat where each clause contains at most 3 variables. For simplicity, let us assume that each clause contains exactly 3 variables. The optimization variant of this problem is to satisfy as many clauses as possible. It is not hard to see that a random assignment satisfies each clause with probability 7/8 and hence if there are m clauses it is not hard (even deterministically) to find an assignment that satisfies $7m/8$ clauses. Since we can never satisfy more than all the clauses this gives an $8/7$-approximation algorithm. This was one of the first approximation algorithms considered [8] and it is surprising that this is optimal to within an arbitrary additive constant $\epsilon > 0$ [7].

The result for Max-Sat is just one of many strong results that has appeared in the last decade. It is also known that it is hard[1] to approximate Set cover on a universe of size n within a factor $\ln n(1 + o(1))$ [4], Max-clique on graphs with n nodes within $n^{1-\epsilon}$ [6] and Graph-coloring within the same factor [10]. In the latter two results the value of ϵ can even be allowed to tend to 0 as the size of the graph increases [3, 9].

[1] In each of these cases an algorithm giving the stated approximation factor would imply very efficient (at most quasi polynomial time) algorithms for NP-complete problems.

2 The beginning

The purpose of this talk is not to give a detailed history and hence we take the liberty to, in the name of brevity, start with the high point of the first development stage of this line of research; the PCP-theorem which was first proved in [1] but which relied heavily on previous work in [5, 2].

The complexity class NP can be viewed as the statements which have polynomial size proofs that can be verified by polynomial time verifiers. Such a verifier would normally read the entire proof. PCP is short for Probabilistically Checkable Proofs and in such a proof a probabilistic verifier only makes spot checks and reads a very small portion of the proof. The PCP-theorem can, semi-formally, be stated as follows.

Theorem 2.1 *[1]Given an arbitrary NP-statement. Then there exists a proof and a probabilistic polynomial time verifier that uses $O(\log n)$ random bits and reads a constant number of bits of the proof. The verifier always accepts a correct proof for a correct statement and rejects any proof of an incorrect statement with probability 1/2.*

Speaking very broadly the results mentioned above are derived from this theorem by changing the proof system to make sure it has additional properties targeted towards the computational problem studied. To see how this is done lets us outline the proof system that gives the result for Max-3Sat.

In this proof system the verifier would flip $O(\log n)$ random bits, compute three bits b_1, b_2, b_3 and finds three positions in the proof. The verifier accepts if the bit at the first location is b_1, or if the bit at the second location is b_2, or if

Proceedings of The 18th IEEE Annual Conference on Computational Complexity
ISBN 0-7695-1879-6/03 $17.00 © 2003 IEEE

the bit at the third location is b_3. The proof system further has the property that for a correct proof of a correct statement the verifier always accepts while it accepts a proof of an incorrect statement with probability at most $(7/8 + \epsilon)$.

To see the connection to satisfiability one has to put oneself into the shoes of a "cheating" prover that has no respect for truth and only wants to convince the verifier with the maximal probability. Thus given the statement to be proved we want to find the proof that convinces the verifier with the highest possible probability. To this end we introduce Boolean variables x_i for the bits of the proof. For each random choice of the verifier the condition that the verifier accepts is clearly a clause of size 3. Consider the set of clauses generated by all random choices of the verifier. This set is of polynomial size due to the fact that the verifier only uses $O(\log n)$ random bits.

If the NP-statement is true, by the property that the verifier always accepts a correct proof of a correct statement, all these clauses can be simultaneously satisfied. On the other hand if the NP-statement is false, by the second property of the verifier, no assignment to the variables can satisfy more than a fraction $(7/8 + \epsilon)$ of the clauses. If we could determine the maximal number of simultaneously satisfiable clauses within a factor smaller than $(7/8 + \epsilon)^{-1}$ then we could in fact determine whether the original statement is true. This implies that this must be an NP-hard computational problem and we get the result.

3 The plan

For some other problems the connection is not as direct as the one for Max-3Sat sketched above and indeed finding this connection is sometimes the most difficult part on the road to establishing the result.

The talk will address a few aspects of this area of research and focus around the following themes.

- To survey what is known for some central problems.
- To discuss what properties of the proof system is needed to derive these bounds.
- To bring up and discuss some of the major open problems of the area. I would consider Bisection, Vertex cover, and Coloring of graphs of low chromatic index with the minimal number of colors as favorites among these problems.

Rather than trying to be complete we focus on some themes that apply to the speaker.

References

[1] S. Arora, C. Lund, R. Motwani, M. Sudan, and M.Szegedy. Proof verification and intractability of approximation problems. *Journal of the ACM*, 45:501–555, 1998.

[2] S. Arora and S. Safra. Probabilistic checking of proofs: a new characterization of np. *Journal of the ACM*, 45:70–122, 1998.

[3] L. Engebretsen and J. Håstad. Clique is hard to approximate within $n^{1-o(1)}$. In *Proceedings of 27th International Colloquium on Automata, Languages and Programming, Volume 1853, LNCS*, pages 2–12, 2000.

[4] U. Feige. A threshold of $\ln n$ for approximating set cover. *Journal of the ACM*, 45:634–652, 1998.

[5] U. Feige, S. Goldwasser, L. Lovász, S. Safra, and M. Szegedy. Interactive proofs and the hardness of approximating cliques. *Journal of the ACM*, 43:268–292, 1996.

[6] J. Håstad. Clique is hard to approximate within $n^{1-\epsilon}$. *Acta Mathematica*, 182:105–142, 1999.

[7] J. Håstad. Some optimal inapproximability results. *Journal of the ACM*, 48:798–859, 2001.

[8] D. Johnson. Approximation algorithms for combinatorial problems. *Journal Computer and System Sciences*, 1974:256–278, 9.

[9] S. Khot. Improved inapproximability results for max-clique, chromatic number and approximate graph coloring. In *Proceedings of 42nd Annual IEEE Symposium on Foundations of Computer Science*, pages 600–609, Las Vegas, 2001. IEEE.

[10] U.Feige and J. Kilian. Zero-knowledge and the chromatic number. *Journal of Computer and System Sciences*, 57:187–200, 1998.

Proceedings of The 18th IEEE Annual Conference on Computational Complexity
ISBN 0-7695-1879-6/03 $17.00 © 2003 IEEE