Electronic Colloquium on Computational Complexity, Report No. 39 (1999)

On approximating CSP-B

Johan Håstad
Royal Institute of Technology
johanh@nada.kth.se

September 24, 1999

Abstract

We prove that any constraint satisfaction problem where each variable appears a bounded number of times admits a nontrivial polynomial time approximation algorithm.

1 Introduction

Some NP-hard optimization problems have the property that the polynomial time approximation algorithm with the best provable performance ratio is rather trivial. Consider Max-E3Sat, i.e., we are given a set of m clauses, each containing exactly 3 literals and the objective is to find an assignment that satisfies the maximal number of clauses. It is easy to see that a random assignment satisfies $7m/8$ clauses on average and it is not difficult to find an assignment that satisfies at least this many clauses by the method of conditional expected values. Since no assignment can satisfy more than all m clauses this gives an approximation algorithm with performance ratio $8/7$. It is a surprising fact [5] that this is best possible in that, unless NP=P, no polynomial time approximation can guarantee a performance ratio $8/7 - \epsilon$ for any $\epsilon > 0$. We conclude that Max-E3Sat does not admit a nontrivial efficient approximation algorithm.

For an NP-hard optimization problem it is a basic question whether it admits a nontrivial efficient approximation algorithm. Both positive and negative results are known along these lines. On the one hand, Max-EkSat for $k \geq 3$, Max linear equations over finite fields, and set-splitting of sets of size at least 4 do not allow nontrivial efficient approximation algorithms [5]. On the other hand, max cut, max-directed cut, Max-2Sat and set-splitting for sets of size at most 3 [4, 3], linear equations with two variables in each equation [1] as well as many constraint satisfaction problems [7] do allow nontrivial efficient approximation algorithms.

In many approximation preserving reductions it is easier to start with an instance of a Max-3Sat where each variable appears at most a bounded number of times. Although it is known [6, 2] that 5 occurrences of each variable is sufficient to make Max-3Sat hard to approximate perfectly, the constant of

ISSN 1433-8092

inapproximability is weaker than the above mentioned 8/7. The goal of this paper is to show that this is no accident and in fact for any constraint satisfaction problem, any constant bound on the number of occurrences of each variable implies the existence of a nontrivial efficient approximation algorithm.

The method of proof turns out to be rather straightforward. We write down a polynomial over the real numbers that counts the number of constraints that are satisfied. The structure of this polynomial is simple enough to allow us to find an assignment of nontrivial quality.

2 Preliminaries

For notational convenience our basic domain is $\{-1, 1\}$ where we think of -1 as "true" and 1 as "false". A constraint satisfaction problem (CSP) is given by a function $f : \{-1, 1\}^k \mapsto \{0, 1\}$ for some constant k. An instance of the CSP is given by a collection, $(C_i)_{i=1}^m$, of k-tuples of literals. An assignment satisfies constraint C_i if f, applied to the values of the literals in C_i, returns 1. As an example, for Max-E3Sat we have

$$f(x, y, z) = 1 - \frac{(1 + x)(1 + y)(1 + z)}{8} = \frac{7 - x - y - z - xy - xz - yz - xyz}{8}. \quad (1)$$

Before we proceed let us give the definition of approximation ratio for an algorithm A.

Definition 2.1 *An approximation algorithm A has* performance ratio *c for a CSP-problem if it, for each instance, returns an assignment that satisfies at least O/c constraints, where O is the number of constraints satisfied by the optimal assignment.*

It is natural to think of f as a multilinear polynomial of degree k and since we have chosen $\{-1, 1\}$ as our basic domain the coefficients of this polynomial are exactly the elements of the discrete Fourier transform of f. We write

$$f(x) = \sum_{\alpha \subseteq [k]} f_\alpha x^\alpha,$$

where $x^\alpha = \prod_{i \in \alpha} x_i$. We need a couple of standard facts.

Lemma 2.2 *The coefficient $f_\emptyset$ gives the probability that a random assignment satisfies f. Each f_α is a multiple of 2^{-k} and*

$$\sum_\alpha f_\alpha^2 = f_\emptyset \leq 1.$$

Proof: The first two facts follow from the formula

$$f_\alpha = 2^{-k} \sum_x f(x) x^\alpha,$$

while the last fact is a consequence of Parseval's identity and $2^{-k} \sum_x f(x)^2 = 2^{-k} \sum_x f(x) = f_\emptyset$. ∎

We derive a simple consequence of the last property.

Lemma 2.3 *We have*

$$\sum_\alpha |f_\alpha| \le 2^{k/2}.$$

Proof: By Cauchy-Schwartz' inequality we have

$$\sum_\alpha |f_\alpha| = (\sum_\alpha 1)^{1/2} (\sum_\alpha f_\alpha^2)^{1/2} \le 2^{k/2}.$$

■

We say that an approximation algorithm is nontrivial if it is provably superior to picking a random assignment or, equivalently, if its performance ratio is smaller than $f_\emptyset^{-1}$.

For an instance $I = (C_i)_{i=1}^m$ of a CSP we define a polynomial P_I. If we let x_{C_i} denote the restriction of an assignment x to literals C_i then

$$P_I(x) \triangleq \sum_{i=1}^{m} f(x_{C_i}) \tag{2}$$

and it is thus a polynomial of degree at most k which simply counts the number of satisfied constraints. We turn to studying such polynomials.

3 Finding good assignments for polynomials

Let P be a polynomial containing only multilinear terms of degree at most k with coefficients p_α. In other words

$$P(x) = \sum_{\alpha \subseteq [n], |\alpha| \le k} p_\alpha x^\alpha.$$

We say that P is an a-polynomial iff each p_α is an integer multiple of a. Furthermore, define

$$|P| \triangleq \sum_{\alpha \neq \emptyset} |p_\alpha|,$$

the sum of the absolute values of all coefficients except the constant term and

$$D_P^i \triangleq \sum_{i \in \alpha} |p_\alpha|$$

the sum of absolute values of all coefficients of terms containing i. Finally, let $D_P^{max} \triangleq \max_i D_i(P)$.

D_P^i is a measure on how much P depends on variable i. Since we are interested in assigning values ± 1 to the inputs, changing the value of x_i can never change the value P by more than $D_i(P)$. Similarly D_P^{max} is a measure on how much P depends on any single variable and we call it the *maximal dependence* of P.

We want to find an assignment $x \in \{-1,1\}^n$ such that $P(x)$ is large. The expected value of $P(x)$ for a random x is $p_\emptyset$ and we want to do better. On the other hand

$$P(x) \leq p_\emptyset + \sum_{\alpha \neq \emptyset} |p_\alpha| = p_\emptyset + |P|, \tag{3}$$

which only could be achieved if all terms of P can be made positive at the same time. The key parameter on how close we can get to this upper-bound is $a(D_P^{max})^{-1}$. The role of a is to be a lower bound on the size of the absolute value of any nonzero coefficient, not only in P but also in any polynomial obtained from P by substituting values for some variables. The role of D_P^{max} is to measure the maximal change to all coefficients of P caused by a substitution of a single variable.

We are now ready for our main lemma.

Lemma 3.1 *Given an a-polynomial P of degree at most k then it is possible, in polynomial time, to find $x \in \{-1,1\}^n$ such that $P(x) \geq p_\emptyset + a|P|(2kD_P^{max})^{-1}$.*

Proof: We construct x by an inductive procedure. Assume that P is non-constant since otherwise $|P| = 0$ making the statement trivial. Take any set α corresponding to a minimal nonzero term, i.e., such that $p_\alpha \neq 0$ but such that $p_\beta = 0$ for $\emptyset \neq \beta \subset \alpha$. Now, find an assignment in $\{-1,1\}^\alpha$ to the variables in α such that $p_\alpha x^\alpha = |p_\alpha|$ and substitute these values into P making it a polynomial Q of $n - |\alpha|$ variables. We want to prove that this is a good partial substitution by establishing that

$$q_\emptyset + a|Q|(2kD_Q^{max})^{-1} \geq p_\emptyset + a|P|(2kD_P^{max})^{-1}. \tag{4}$$

If we establish (4) we claim that the lemma follows since if we iterate this procedure we eventually get to an assignment which makes P reduce to a constant which then must be at least $p_\emptyset + a|P|(2kD_P^{max})^{-1}$. Note also that the procedure clearly can be implemented in polynomial time. We turn to establishing (4).

The constant term $q_\emptyset$ of Q is $p_\emptyset + |p_\alpha| \geq p_\emptyset + a$ and Q is of degree at most k. Since each q_β with $i \in \beta$ is the sum of some $p_{\beta'}$ with $i \in \beta'$ we have $D_Q^i \leq D_P^i$ for any i which implies $D_Q^{max} \leq D_P^{max}$.

We turn to studying $|Q|$ which might be smaller than $|P|$ due to cancellation of terms. However only terms of P containing elements from α can create such cancellation. Since α is of size at most k, the sum of the absolute values of all coefficients of all terms affected is bounded by kD_P^{max}. Each such term affected can at most cancel another term and hence we have

$$|Q| \geq |P| - 2kD_P^{max}.$$

Summing up, we get

$$\begin{aligned} q_\emptyset + a|Q|(2kD_Q^{max})^{-1} &\geq p_\emptyset + a + a|Q|(2kD_P^{max})^{-1} \geq \\ p_\emptyset + a + a(|P| - 2kD_P^{max})(2kD_P^{max})^{-1} &\geq p_\emptyset + a|P|(2kD_P^{max})^{-1} \end{aligned}$$

and we have established (4) and the lemma follows. ■

4 Application to CSPs

We now present the theorem of the paper.

Theorem 4.1 *Consider a CSP given by f defined on k-tuples of literals. On the class of instances where each variable appears at most B times this problem can be approximated within $(f_\emptyset + (1-f_\emptyset)2^{-3k/2}(2kB)^{-1})^{-1}$ in polynomial time. In other words, we have a nontrivial efficient approximation algorithm for any f and any constant B.*

Proof: Given an instance I, consider the polynomial P_I defined by (2). We want to apply our main lemma to this polynomial. It is of degree at most k and by Lemma 2.2 we conclude that each coefficient is a multiple of 2^{-k} and that it has constant term $mf_\emptyset$. Furthermore we have

Lemma 4.2 $D_{P_I}^{max} \le B2^{k/2}$.

Proof: Each term $p_\alpha x^\alpha$ where $j \in \alpha$ comes from a term $f(x_{C_i})$ in (2) such that the variable x_j appears in C_i. Since x_j appears in at most B constraints and, by Lemma 2.3, $\sum |f_\beta| \le 2^{k/2}$ it follows that $D_P^j \le B2^{k/2}$. Since j was arbitrary, the lemma follows. ■

We now have all the information to apply Lemma 3.1 to P_I. The result is an assignment that satisfies at least $mf_\emptyset + |P_I|2^{-k}(2kB2^{k/2})^{-1}$ of the constraints. On the other hand, by (3), no assignment can satisfy more than $mf_\emptyset + |P_I|$ constraints and another upper bound is given by all constraints m. Thus the performance ratio of the algorithm is bounded by

$$\frac{\min(m, mf_\emptyset + |P_I|)}{mf_\emptyset + |P_I|2^{-k}(2kB2^{k/2})^{-1}}.$$

This is maximized when the two terms in the minimum are equal in which case $|P_I| = (1-f_\emptyset)m$ and this gives performance ratio

$$(f_\emptyset + (1-f_\emptyset)2^{-3k/2}(2kB)^{-1})^{-1}.$$

■

Let us apply the theorem to one of the most popular problems, Max-E3Sat-B. Since $k = 3$ and $f_\emptyset = 7/8$, we see that it can be approximated within $(7/8 + c/B)^{-1}$ for $c = 2^{-17/2}3^{-1}$. A tighter analysis below improves the value of c.

Remember the explicit formula for f given by (1) and let us go over the steps of the proof. Suppose we choose the α in the proof of Lemma 3.1 to be of minimal size among all sets corresponding to a nonzero term.

If $|\alpha| = 1$ then we note that for any occurrence of a variable x in a clause we get a total contribution $3/8$ to coefficients of terms containing x together with other variables. Thus we get that the sum of absolute values of coefficients of such terms is bounded by $3B/8$. Since each term might cancel another term, we conclude that $|P_I|$ decreases by at most $3B/4$.

If $|\alpha| = 2$ then two variables, x and y, are involved, but since P_I does not contain any linear terms we can get improved estimates of the cancellation by a more careful analysis. Of terms containing x or y, the only degree-two terms in P_I that can get cancelled are terms cancelling each other. The sum of absolute values of coefficients of such terms is bounded by $B/2$. Terms of degree 3 involving x or y have coefficients of total absolute value at most $B/4$, but since they can cancel other terms they may cause cancellation of terms of total absolute value at most $B/2$. Thus, in total, we conclude that $|P_I|$ decreases by at most B in the case $|\alpha| = 2$.

If $|\alpha| = 3$ we only have terms of degree 3 in P_I. The total absolute value of coefficients of terms containing one of 3 variables is $3B/8$ and thus cancellation in this case is bounded by $3B/4$.

Summing up, we see that we increase the constant term by at least $1/8$ and decrease $|P_I|$ by at most B. We conclude that we find an assignment that satisfies at least $7m/8 + |P_I|/(8B)$ clauses. Since we should compare this to the minimum of m and $7m/8 + |P_I|$ we get performance ratio $(7/8 + 1/(64B))^{-1}$.

Thus a more careful analysis did give a substantial improvement in the constant but we do not believe, however, that the inverse linear dependence of B can be improved using the methods of this paper.

5 Discussion

Assuming familiarity with [5] let us give a brief discussion of the optimality of these results. In that paper, to establish inapproximability $2 - \epsilon$ for Max-Lin-2, an instance is constructed where each variables appears at most $2^{2^{O(v)}}$ times. The parameter v satisfies $c^v < \epsilon^{O(1)}$ for a constant $c < 1$. Thus we get $B = O(2^{\epsilon^{-d}})$ for some constant d.

Since the same relationship applies to the approximability of Max-E3Sat we see that, unless P=NP, we could not hope to get performance ratio better than

$$(7/8 + c(\log B)^{-d})^{-1}.$$

for some positive constants c and d by an algorithm running in polynomial time. Of course, we are still far from getting such strong results.

Acknowledgment I am grateful to Gunnar Andersson and Madhu Sudan for comments on the presentation of this paper.

References

[1] G. Andersson, L. Engebretsen, and J. Håstad A new way to use semidefinite programming with applications to linear equations mod p. *Proceedings 10th Annual ACM-SIAM Symposium on Discrete Algorithms*, 1999, pp 41–50.

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

[3] U. Feige and M. Goemans. Approximating the value of two prover proof systems, with applications to MAX 2SAT and MAX DICUT. 3rd Israeli Symposium on the theory of Computing and Systems, 1995, pp 182–189.

[4] M. Goemans and D. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. Journal of ACM, vol 42, 1995, pp 1115–1145.

[5] J. Håstad. Some optimal inapproximability results. Proceedings of 29th Annual ACM Symposium on Theory of Computation, El Paso, 1997, pp 1–10.

[6] C. Papadimitriou and M. Yannakakis. Optimization, approximation and complexity classes. Journal of Computer and System Sciences, Vol 43, 1991, pp 425–440.

[7] U. Zwick. Approximation algorithms for constraint satisfaction problems involving at most three variables per constraint. *Proceedings 9th Annual ACM-SIAM Symposium on Discrete Algorithms*, 1998, pp 201–210.

ECCC ISSN 1433-8092
http://www.eccc.uni-trier.de/eccc
ftp://ftp.eccc.uni-trier.de/pub/eccc
ftpmail@ftp.eccc.uni-trier.de, subject 'help eccc'