On the Hardness of Approximating Max-Satisfy

Uriel Feige and Daniel Reichman
The Weizmann Institute, Rehovot 76100, Israel,
{uriel.feige@weizmann.ac.il, daniel.reichman@gmail.com}

September 14, 2006

Abstract

Max-Satisfy is the problem of finding an assignment that satisfies the maximum number of equations in a system of linear equations over $\mathbb{Q}$. We prove that unless NP$\subset$BPP Max-Satisfy cannot be efficiently approximated within an approximation ratio of $1/n^{1-\epsilon}$, if we consider systems of n linear equations with at most n variables and $\epsilon > 0$ is an arbitrarily small constant. Previously, it was known that the problem is NP-hard to approximate within a ratio of $1/n^{\alpha}$, but $0 < \alpha < 1$ was some specific constant that could not be taken to be arbitrarily close to 1.

Keywords. Approximation algorithms, computational complexity.

1 Introduction

MAX-SATISFY is the problem finding an assignment that satisfies as many equations as possible in a system of linear equations over the field of rational numbers. This problem appears in various contexts such as pattern recognition, operations research and artificial neural networks (see the references in [2], for example). MAX-SATISFY is NP-hard. We say that an algorithm approximates MAX-SATISFY within a ratio of ρ (where $0 < \rho < 1$) if on every instance I of MAX-SATISFY the algorithm returns an assignment that satisfies at least $\rho \cdot \mathrm{opt}(I)$ equations, where $\mathrm{opt}(I)$ is the maximum number of equations of I that can be satisfied simultaneously.

If the system is satisfiable then one can find an assignment satisfying all equations in polynomial time, using Gaussian elimination. However, if the system is not satisfiable, then even approximating MAX-SATISFY within a ratio of $1/n^{\alpha}$ (where n is the number of equations and an upper bound on the number of variables, α is some specific constant less than 1) is NP-hard [2, 3]. The best approximation algorithm for the problem (due to Halldorsson [9]) achieves approximation ratio $O(\log n/n)$.

One may hope that the constant α in the $1/n^{\alpha}$ hardness of approximation result can be taken to be arbitrarily close to 1. In particular, the construction of arbitrarily low amortized free bits PCPs ([10, 12]) together with the appropriate

reductions ([7, 6]) has resulted in such an inapproximability result for MAX-CLIQUE (unless NP=ZPP). In this work we show that this methodology works for MAX-SATISFY as well. However, there are some difficulties in applying this methodology compared to the case of MAX-CLIQUE. One of them is that the relation between amortized free bits and the approximation of MAX-SATISFY is not as direct as their relation with MAX-CLIQUE, and in fact was not present in the works of [2, 3]. This we handle in Lemma 8, which arithmetizes the low amortized free bits PCP of [12] (using principles taken from [8]). A somewhat more challenging difficulty is that the number of possible assignments to the variables of a system over $\mathbb{Q}$ is infinite (whereas the number of possible cliques in an n-vertex graph is at most 2^n). This complicates the analysis of a randomized reduction (originally due to [7]) that we use. In order to overcome this difficulty (in Lemma 7) we make use of a theorem of [5] regarding the number of so called *zero patterns* of a system of polynomials. We remark that also [2, 3] needed to overcome such a difficulty, but their method of doing so would not give hardness of approximation results beyond $1/n^{1/2-\epsilon}$, not even if coupled with our Lemma 8, and not even if the deterministic amplification technique that they use (based on [1]) is replaced by its more efficient randomized counterpart.

Our main theorem is the following.

Theorem 1 *Unless NP$\subset$BPP, MAX-SATISFY cannot be approximated within a ratio of $\frac{1}{n^{1-\epsilon}}$, where n is the number of equations in the system and $\epsilon > 0$ is an arbitrarily small constant.*

2 Preliminaries

We denote by $[n]$ the set of integers between 1 and n. $\mathbb{Q}$ is the field of rational numbers. $\mathbb{N}$ is the set of natural numbers.

We now define the concept of a *probabilistically checkable proof* (PCP):

Definition 1 *Let r and q be two functions from $\mathbb{N}$ to $\mathbb{N}$. A randomized polynomial-time Turing machine V with access to an oracle (string) π, is called an (r, q)* restricted verifier *if, for every oracle π and for every input x, V uses $\mathrm{O}(r(|x|))$ random bits and queries $\mathrm{O}(q(|x|))$ bits of π.*

We deal exclusively with non-adaptive verifiers. A nonadaptive verifier decides which queries to make based only on the input and on its random bits (but not on answers it gets for previous queries).

Definition 2 *Let $0 \leq s < c \leq 1$. A language L is said to belong to the class $PCP_{c,s}[r, q]$ if there is an (r, q) restricted verifier V s.t.*

1. *If $x \in L$, then there exists an oracle π such that V (with oracle π) accepts x with probability at least c, where the probability is taken over the random bits used by V.*

2. *If $x \notin L$ then, for* any *oracle π, the probability V accepts with oracle access to π is at most s.*

We refer to c as the *completeness* of the PCP and to s as the *soundness* of the PCP. If $c = 1$, we say that the verifier has perfect completeness.

Arora et al. [4] proved the following important result which is widely known as the PCP Theorem:

Theorem 2 $NP = PCP_{1,1/2}(\log n, 1)$.

The discovery of the PCP Theorem was followed by attempts to optimize various parameters in proof systems. One such parameter of importance is the *amortized free bit complexity* [6].

Definition 3 *We say that a PCP with query complexity q has free bit complexity $f \leq q$ if for every set of q queries (that the verifier may make) there are at most 2^f assignments to the queried locations that cause the verifier to accept. The* amortized free bit complexity *of a PCP with free bit complexity f and soundness s is $f/\log s^{-1}$.*

The amortized free bit complexity is related to the hardness of approximating Max-Clique. In particular, PCP with arbitrarily small amortized free bit complexity implies that Max-Clique cannot be approximated within a ratio $1/n^{1-\epsilon}$ for any positive ϵ, unless NP=ZPP (n is the number of vertices in the graph) [6]. A PCP with arbitrarily low amortized free bit complexity was first given by Hástad [10]. His construction was quite involved. A simpler construction was given by Samorodnitsky and Trevisan [12]:

Theorem 3 *For every positive ϵ and integer f there is a PCP characterization of NP with free bit complexity f and query complexity $q = f^2/4 + f$, such that a correct proof is accepted with probability at least $1 - \epsilon$ and a wrong proof is accepted with probability at most $2^{-f^2/4}$.*

We shall need the notion of zero patterns:

Definition 4 *Let $e_1 \ldots e_N$ be linear equations over a field F. We say $\zeta \in \{0,1\}^N$ is a* zero pattern *of these equations if there is an assignment $\sigma(\zeta)$ to the variables of the equations $e_1, \ldots, e_N$ such that $\sigma(\zeta)$ zeros e_i iff the ith coordinate of ζ is zero.*

It is clear that the number of zero patterns of N equations is bounded by 2^N. It turns out that if d is small, we can get a better bound, as shown in Lemma 4. (We remark that Lemma 4 is a special case of a more general result that is proved in [5], where it is shown that the number of zero patterns of a system of polynomials of maximum degree D is at most $\binom{DN+d}{d}$. In our case, $D = 1$.)

Lemma 4 *For every system of n linear equations over d variables, the number of zero patterns is at most $\sum_{i=0}^{d}\binom{n}{i}$.*

Proof: Let $e_1, ..., e_n$ be linear equations. With every equation e_i we associate the d dimensional vector $c(e_i)$ corresponding to the coefficients of the variables of the equation. For example, when $d = 3$ the 3-dimensional vector associated with the equation $2x_1 - x_3 + 5 = 0$ is $(2, 0, -1)$. Say that a set of equations is independent if their associated vectors are linearly independent. Now we show that the number of zero patterns is at most as large as the number of sets of independent equations. With every zero pattern we associate a maximal independent set of equations among those equations that are zeroed by the pattern. (If there is more than one maximal independent set, we choose one of them arbitrarily.) An independent set S of equations cannot be maximal for two different zero patterns, because the fact that all equations in S are zeroed uniquely determines the value of every equation that depends on S, Hence no independent set is associated with two different zero patterns, and indeed the number of zero patterns is at most the number of independent sets. Since the vectors associated with the equations are d dimensional, the size of an independent set is at most d, and the number of independent sets is at most $\sum_{i=0}^{d}\binom{n}{i}$, proving the lemma. □

Note: In our context d will be much smaller than n. Hence we can bound $\sum_{i=1}^{d}\binom{n}{i}$ by $d\binom{n}{d}$.

Finally we shall need the following tail estimate widely known as the Chernoff bound:

Corollary 5 *Assume $X_1 \dots X_n$ are mutually independent $\{0,1\}$-valued random variables. Let $X = \sum_{i=1}^{n} X_i$. For any $\delta \in (0,1)$ we have:*

$$\Pr(X > (1+\delta)\mathbf{E}(X)) < e^{-\delta^2\mathbf{E}(X)/3}$$

$$\Pr(X < (1-\delta)\mathbf{E}(X)) < e^{-\delta^2\mathbf{E}(X)/2}$$

where $\mathbf{E}(X)$ is the expectation of the random variable X.

For a proof see [11].

3 The Main Result

We begin by describing a way to increase the gap in MAX-SATISFY beyond any constant. This approach was suggested in [3] as well as in [2].

Our starting point is the existence of universal constants $\delta, \eta \in (0,1)$ such that it is NP-hard to distinguish between instances of MAX-SATISFY with n equations in which OPT $\geq \eta \cdot n$ and instances in which OPT $\leq \eta \cdot \delta \cdot n$ (see lemma 8). Take k and T to be integers that will be determined later. Let E be an instance of MAX-SATISFY with n equations, $p_1 = 0, ..., p_n = 0$ (Of course the equations need not be homogenous. The free coefficient is on the left

hand side). For every k-tuple $(i_1, ...i_k) \in [n]^k$, construct a *block* of T equations $\sum_{j=1}^{k} p_{i_j} \cdot y^j = 0$, where y ranges over all integers in $[T]$. If all equations p_{i_j} are satisfied in a given assignment, then all the equations in the corresponding block will be satisfied. If even one of the equations p_{i_j} is not satisfied, then at most k out of the T equations in the block will be satisfied, as $p(x) = \sum_{j=1}^{k} p_{i_j} \cdot x^j$ is a polynomial of degree at most k which is not identically 0, and such a polynomial can have at most k distinct roots. The total number of equations in this instance is $T \cdot n^k$. We name the new system of equations obtained in this way E^k. If we can satisfy in the original instance $\eta \cdot n$ equations, the same assignment will satisfy at least $T(\eta \cdot n)^k$ equations. If, on the other hand, we can satisfy at most $\eta \cdot \delta \cdot n$ equations in the original system we can satisfy at most $T(\eta \cdot \delta \cdot n)^k + k \cdot n^k$ equations. Taking T large enough (larger than $k/(\delta\eta)^k$, say) we get a gap of approximately $(1/\delta)^k$. Taking k to be an arbitrarily large constant, the gap can be made arbitrarily large, ruling out the exitance of a constant ratio approximation algorithm for MAX-SATISFY.

By taking k to be a function of n such as $\log n$ (ignoring for the moment the fact that in this case the reduction is no longer polynomial) one may increase the gap in the approximation ratio to some function that depends on n. However, at the same time the number of equations increases, and this approach does not prove that MAX-SATISFY is hard to approximate within $1/n^\alpha$ for some positive α. We can overcome this obstacle by using random sampling, as pointed out by Berman and Schnitger [7] in a related context. We remark that in [2, 3] a similar idea was also used, but the sampling there was taken from a different distribution that is easier to analyse, but does not give as good results as we get here.

Throughout n will denote the number of equations in the original system. We construct a new linear system from the original one as follows. Let $k = \Theta(\log n)$. In order to simplify the notation we denote $\eta \cdot \delta$ by b and η by a. As we will see later, we have such a and b where $b \leq a^p$ for arbitrarily large p. Choose k such that $a^{-k} = \Theta(n^2)$. We now create a system RE^k containing $a^{-(p+1)k}$ equations chosen at random from the system E^k described above. We cannot afford to first construct E^k explicitly (because E^k contains superpolynomially many equations when $k = \Theta(\log n)$) but we can still construct RE^k in random polynomial time. Pick uniformly at random $i_1 \in [n], i_2 \in [n], \ldots, i_k \in [n]$, and $y \in [T]$. Add the equation $\sum_{j=1}^{k} p_{i_j} \cdot y^j = 0$ to the system. Repeat this process independently $a^{-(p+1)k}$ times. We thus get a total of $a^{-(p+1)k}$ equations.

What is the idea in this construction? Assume we have an assignment τ that satisfies $a \cdot n$ equations in the original instance. We fall into a block in which all equations are satisfied with probability a^k. Hence, the expected number of equations in RE^k satisfied by τ is at least a^{-pk}. Suppose we have an assignment σ which satisfies at most $a^p \cdot n$ equations in the original system. Then the expected number of equations satisfied by σ in the new instance is at most $a^{-(p+1)k} \cdot (a^{pk} + k/T)$. If we take T to be larger than $k(\frac{1}{a})^{pk}$ then the gap between the two instances is $\Omega(a^{-k(p-1)})$. As we have $a^{-(p+1)k}$ equations we get that MAX-SATISFY is hard to approximate within roughly $n^{-(1-\frac{2}{p+1})}$ (recall

that n is the number of equations in the instance). Since p can be taken to be arbitrarily large we get that the problem is hard to approximate within $\frac{1}{n^{1-\epsilon}}$ for arbitrarily small positive ϵ.

We first prove:

Lemma 6 *If the original instance E is a-satisfiable, then, with probability at least $(1 - e^{-\Omega(n^2)})$ RE^k is $\frac{1}{2} \cdot a^{-pk}$ satisfiable.*

Proof: Let σ be an assignment satisfying $a \cdot n$ equations in the original system E. Give to the variables of RE^k the *same values* given to them by σ. The expected number of equations satisfied in RE^k is at least a^{-pk} (where expectation is taken over the random choice of RE^k). Applying the Chernoff bound, we get that with probability at least $1 - e^{-\Omega(n^2)}$ (remember (a^{-pk} is $\Omega(n^2)$), the number of satisfied equations in RE^k with this assignment is at least $\frac{1}{2} \cdot a^{-pk}$. $\square$

We now want to use the Chernoff bound along with the union bound in order to show that if our original system E was at most a^p satisfiable, then *every* assignment to the variables of RE^k will satisfy "few" equations. There is a problem, however, in applying the union bound as the number of assignments to the variables of RE^k is infinite. The crucial observation in overcoming this problem is that if two assignments have identical zero patterns over E^k (namely, they satisfy the same subset of equations from E^k, but may differ on the nonzero values that they give to other equations), then their zero patterns with respect to RE^k are also identical.

The number of equations in E^k is $T \cdot n^k$. Recall k was chosen s.t. $a^{-k} = \Theta(n^2)$. Choose T to be $k \cdot a^{-pk}$. Thus, T is $O(n^l)$ for some constant l where l depends only on the constants a and p. The number of variables in E^k is the same as in E which is $O(n)$ (see Lemma 8; Recall n is the number of equations in the original system). Using lemma 4 we get that the number of zero patterns of the set E^k of linear equations is at most

$$O(n) \cdot \binom{n^{\Theta(\log n)}}{O(n)} = 2^{O(n \cdot \log^2 n)}$$

Now we can prove:

Lemma 7 *If the original instance E was at most a^p-satisfiable, then with probability at least $1 - e^{-\Omega(n^2)}$ RE^k is at most $3 \cdot a^{-k}$ satisfiable.*

Proof: Consider an arbitrary zero pattern of E^k, σ. Once we choose the equations in RE^k, σ induces a zero pattern on RE^k. Denote this zero pattern by σ'. The expected number of satisfied equations in σ' is at most

$$a^{-k} + a^{-k(p+1)} \frac{k}{T}$$

¿From the way we choose T the expression above is at most $2a^{-k}$. Applying the Chernoff bound we infer that the probability that the number of satisfied

equations will be more than $3a^{-k}$ is at least $e^{-\Omega(n^2)}$ (Since we choose k so that a^{-k} is $\Omega(n^2)$). We have seen that the number of zero patterns of E^k is bounded by $2^{\Theta(n \cdot \log^2 n)}$. Taking the union bound over all zero patterns, we get the required result. $\square$

We now justify our assumption that for arbitrarily large p there exists some $a \in (0,1)$ such that it is NP-hard to distinguish between linear systems that are a satisfiable and linear systems that are at most a^p satisfiable. This follows from the following lemma:

Lemma 8 *For arbitrarily small positive δ there are $0 < b < a < 1$ with $\frac{\log a}{\log b} < \delta$ such that it is NP-hard to tell whether a linear system over the rationals is at most b-satisfiable or at least a-satisfiable.*

Proof: We take the PCP from Theorem 3. Recall this PCP has f free bits, the completeness is at least $1-\epsilon$ and the soundness is at most $2^{-f^2/4}$. Create a set of linear equations over rational as follows: We introduce a variable for every position in the proof that has positive probability of being queried. Thus, the number of variables is at most $2^{O(r)} \cdot q$. For the sake of simplicity we write 2^r rather than 2^{hr} for some suitable constant h. For every possible choice of random bits we have at most 2^f possibilities for the queried positions that make the verifier accept. Fix l to equal $2^{f^2/4}$. If the values $b_1, ..., b_q$ cause the verifier to accept, we add l equations $\sum_{i=1}^{q} (x_i - b_i) = 0$, $\sum_{i=1}^{q} (x_i - b_i)2^i = 0 \ldots \sum_{i=1}^{q} (x_i - b_i)l^i = 0$. (Note – the indices of the variables should correspond to the queried positions and not to $1, 2, ..., q$. We write it like we did to avoid notational difficulties). Clearly, if for every i, $x_i = b_i$ then all l equations are satisfied. If $x_i \neq b_i$ for some i, then at most q out of the l equations are satisfied.

We get a total of $l(2^{r+f}) = 2^{r+f+f^2/4}$ equations. (Note that the number of equations is larger than the number of variables.) If we have success probability at least $1-\epsilon$ in the above proof system then we can satisfy at least $(1-\epsilon)l2^r = 2^{r+f^2/4+o(1)}$ equations. If, on the other hand, our success probability is at most $2^{-f^2/4}$ then we can satisfy at most $2^r l 2^{-f^2/4} + q2^{r+f} = 2^{r+O(f)}$ equations. (Note that $q = 2^{O(\log f)}$.) In the first case, we can satisfy at least $2^{((r)+(f^2)/4+o(1))-(r+f+f^2/4)=2^{-O(f)}}$ fraction of the equations. In the second case, we can satisfy at most $2^{r+O(f)-(r+f+f^2/4)} = 2^{-\Omega(f^2)}$ fraction of the equations. Hence, we get $0 < b < a < 1$ such that $\log b = -\Omega(f^2)$, $\log a = -O(f)$ and it is NP-hard to distinguish between equations over the rationals that are at least a-satisfiable to equations that are at most b-satisfiable. As f can be taken to be arbitrarily large, we are done. $\square$

The proof of Theorem 1 follows by combining the three lemmas above.

Acknowledgements

This research was supported by the Israeli Science Foundation (grant number 263/02).

References

[1] N. Alon, U. Feige, A. Wigderson, D. Zuckerman. Derandomized Graph Products. *Computational Complexity*, 5: 60–75, 1995.

[2] E. Amaldi and V. Kann. The complexity and approximability of finding maximum feasible subsystems of linear relations. *Theoretical Computer Science*, 147:181-210, 1995.

[3] S. Arora, L. Babai, J. Stern, and Z. Sweedyk. The hardness of approximate optima in lattices, codes, and systems of linear equations. *Journal of Computer and System Sciences*, 54(2):317–331, 1997.

[4] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and the hardness of approximation problems. *J. ACM*, 45(3):501–555, 1998.

[5] L. Babai, L. Ranyai, M. K. Ganapathy. On the number of zero-patterns of a sequence of polynomials. Journal American Mathematical Society. 14 , 717–735, 2001.

[6] M. Bellare, O. Goldreich, M. Sudan. Free Bits, PCPs, and Nonapproximability-Towards Tight Results. *SIAM Journal of Computing*. 27(3): 804–915, 1998.

[7] P. Berman, G. Schnitger. On the Complexity of Approximating the Independent Set Problem. *Information and Computation*, 96: 77–94, 1992.

[8] U. Feige, D. Reichman. On systems of linear equations with two variables per equation. In the proceedings of APPROX 2004 (LNCS 3122), 117–127, Springer, 2004.

[9] M. M. Halldorsson. Approximations of Weighted Independent Set and Hereditary Subset Problems. J. Graph Algorithms Appl. 4(1), 2000.

[10] J.Hástad. Clique is Hard to Approximate within n to the power 1-epsilon. *Acta Mathematica*, 182: pp 105–142, 1999.

[11] R. Motwani, P. Raghavan Randomized Algorithms. Cambridge University Press, 1995.

[12] A. Samorodnitsky, L. Trevisan. A PCP characterization of NP with optimal amortized query complexity. Proceedings of STOC, 191–199, 2000.