Near-Optimal Nonapproximability Results for Some NPO PB-Complete Problems

P. Jonsson

Department of Computer and Information Science
Linköpings universitet, S-581 83 Linköping, Sweden

Abstract

We show that a number of NPO PB-complete problems, including MIN ONES and MAX ONES, are hard to approximate within $n^{1-\epsilon}$ for arbitrary $\epsilon > 0$.

Keywords: Approximation, Computational complexity, Combinatorial problems.

1 Introduction

NPO PB is the class of NP optimization problems whose objective function is bounded by some polynomial in the size of the input. It is well-known that many problems that are complete for NPO PB are notoriously hard to approximate and near-optimal lower bounds on the approximability of NPO PB-complete problems such as MIN # SAT and MIN PB 0-1 PROGRAMMING have appeared in the literature [5]. However, there are still many problems for which tight bounds are not known. In this paper we provide such bounds for five NPO PB-complete problems: MIN ONES, MIN DONES, MAX ONES, MAX DONES and MAX PB 0/1 PROGRAMMING. For each of these problems we show that they cannot be approximated within $n^{1-\epsilon}$ for any $\epsilon > 0$ where n is the number of variables. Since all of these problems can trivially be approximated within n, this bound is essentially optimal.

2 Preliminaries

Throughout this paper, we assume that P $\neq$ NP. Solving an NP optimization problem A given the input instance x means finding a solution y such that the value of the objective function $m_A(x, y)$ is maximum or minimum depending on if A is a maximization or minimization problem. As usual, we assume that

m_A is polynomial-time computable. Let the *optimal value function* opt_A return the optimal value of m_A for arbitrary instances x of A. For technical reasons, we follow Berman and Schnitger's [1] convention that whenever the optimal value function returns value 0, this value is to be redefined to 1 instead.

The class NPO is the set of all NP optimization problems. An NPO problem A is said to be *polynomially bounded* if a polynomial q exists such that for any instance x, $opt_A(x) \leq q(|x|)$. The class NPO PB is the set of polynomially bounded NPO problems.

Definition 1 *Given an* NPO *problem A and a function $\alpha : \mathbb{N} \to (1, \infty)$, we say that a polynomial-time algorithm P is an α-*approximation algorithm *for A iff for every instance x of A of size n, P produces a solution in the range $[opt(x)/\alpha(n), \alpha(n)opt(x)]$. We say that A is* approximable within a factor α *if such an algorithm exists.*

Lemma 2 *Let $a, b : \mathbb{N} \to (1, \infty)$ be functions such that given natural numbers i, j, it can be decided in polynomial time whether $a(i) < j$ or not. Let A be an NP maximization problem and T a polynomial-time computable transformation from an NP-complete decision problem F to A. Assume that for every instance $I \in F$, $|T(I)| = \ell(|I|)$ for some function $\ell : \mathbb{N} \to \mathbb{N}$. Then, A is not approximable within a factor $\alpha(n) = b(n)/a(n)$ if T has the following two properties: (1) If $I \in F$ then $opt_A(T(I)) > b(|T(I)|)$; and (2) if $I \notin F$ then $opt_A(T(I)) < a(|T(I)|)$.*

Similarly, if A is a minimization problem then A is not approximable within a factor $\alpha(n) = b(n)/a(n)$ if T satisfies the following: (1) If $I \in F$ then $opt_A(T(I)) < a(|T(I)|)$; and (2) if $I \notin F$ then $opt_A(T(I)) > b(|T(I)|)$.

Proof. We prove the maximization case; the minimization case is analogous. Assume the existence of an polynomial-time algorithm K that can approximate A within α. Then, for an arbitrary instance x of A,

$$\frac{a(|x|) \cdot opt_A(x)}{b(|x|)} \leq m_A(x, K(x)) \leq opt_A(x).$$

Arbitrarily select an instance I of F and let $I' = T(I)$ and $m = |I'|$. Note that the value of m does not depend on whether I is in F or not. If $I \in F$ then $opt_A(I') > b(m)$ and $a(m) < m_A(I', K(I'))$. Assume to the contrary that $I' \notin F$. Then $opt_A(I') < a(m)$ and $m_A(I', K(I')) < a(m)$. Hence, $I \in F$ iff $a(m) < m_A(I', K(I'))$. Since this condition can be decided in polynomial time, the lemma follows.

The nonapproximability of problems is described as a function of the size of the problem instance, or more often, as a function of some size parameter such as the number of variables in the instance. For all problems considered in this paper, we will assume that $|\cdot|$ returns the number of variables in the given instance. Further, we assume that there exists a trivial solution $triv(x)$ for each input x so that it can be ensured that an approximation algorithm always finds a feasible solution.

We continue by giving formal definitions of the problems that we will investigate. Proofs of NPO PB-completeness of these problems can be found in [1,4,5]. We say that a propositional formula is 3CNF iff it is the conjunction of clauses with at most three literals per clause. The first problem we consider is MIN DONES:

INSTANCE: Disjoint sets X, Z of variables and a 3CNF formula F over $X \cup Z$.
SOLUTION: Truth assignment satisfying F.
MEASURE: The number of variables in Z that are set to true in the assignment.

Arbitrarily choose $\epsilon > 0$. MIN DONES cannot be approximated within $|Z|^{1-\epsilon}$ [5]. MIN ONES, the variation in which all variables are distinguished, *i.e.*, $X = \emptyset$, is not approximable within $|Z|^{1/2-\epsilon}$ [5]. The maximization version of MIN DONES (denoted MAX DONES) is not approximable within $|Z|^{1-\epsilon}$ and not within $(|X| + |Z|)^{1/2-\epsilon}$ [6]. Let MAX ONES denote the MAX DONES problem restricted to $X = \emptyset$. MAX ONES is not approximable within $|Z|^{1/3-\epsilon}$ [6].

Finally, we present the problem MAX PB 0-1 PROGRAMMING:

INSTANCE: Integer $m \times n$ matrix A, integer m-vector b and nonnegative binary n-vector $c \in \{0,1\}^n$.
SOLUTION: A binary n-vector $x \in \{0,1\}^n$ such that $Ax \geq b$.
MEASURE: The scalar product of c and x, *i.e.*, $\sum_{i=1}^{n} c_i x_i$.

Kann [6] has shown that this problem cannot be approximated within $n^{1/2-\epsilon}$. A strong lower bound on the minimization version of MAX PB 0-1 PROGRAMMING has been established by Kann [5].

Since the optimal value cannot equal 0 by definition, these problems can trivially be approximated within n (where n is the number of variables) since we have assumed the existence of trivial solutions.

It is easy to see that if MAX ONES is not approximable within α then MAX DONES is not approximable within α. Using the well-known connection between propositional logic and zero/one programming [7], it is also easy to see that MAX PB 0-1 PROGRAMMING cannot be approximated within α. Similarly, nonapproximability of MIN ONES carry over to MIN DONES and MIN PB 0-1 PROGRAMMING.

3 Minimization Problems

The first problem that we will consider is Min Ones. Define the NP-complete decision problem 3Sat as usual [2]: Given a 3CNF formula F, is there a satisfying truth assignment for F?

Lemma 3 *Let Min_1 denote the optimal value function for* Min Ones*. Let $c > 4$ be a fixed integer. There exists a polynomial-time tranformation T_c on 3CNF formulae satisfying the following: for an arbitrary 3CNF formula F,*

(i) $|T_c(F)| = |F|^c$;
(ii) $Min_1(T_c(F)) \leq |F|^4$ *if F is satisfiable;*
(iii) $Min_1(T_c(F)) \geq |F|^c - 8|F|^3 - |F|$ *otherwise.*

Proof. Let F be an arbitrary 3CNF formula over the propositions $P = \{p_1, \ldots, p_k\}$. We begin by describing $T_c(F)$.

There exists less than $8k^3$ distinct clauses of length 3 over the propositional symbols in P. We can thus assume that $F = C_1 \wedge \ldots \wedge C_s$ where $s \leq 8k^3$. Let w be a fresh proposition and $Q = \{q_1, \ldots, q_{k^c-k-s-1}\}$, $R = \{r_1, \ldots, r_s\}$ be sets of fresh propositions.

Consider an arbitrary clause $C_i = (l_i^1 \vee l_i^2 \vee l_i^3)$, $1 \leq i \leq s$. Since the clauses in F contain *at most* three literals, we do not require that l_i^1, l_i^2 and l_i^3 are distinct. Let

$$K_i = (r_i \Leftrightarrow (l_i^1 \vee l_i^2)) \wedge (l_i^3 \vee r_i \vee w)$$

where $\Leftrightarrow$ denotes logical equivalence, *i.e.*, $\alpha \Leftrightarrow \beta$ iff $(\neg\alpha \vee \beta) \wedge (\alpha \vee \neg\beta)$. Define $T_c(F)$ as follows:

$$T_c(F) = \bigwedge_{1 \leq i \leq s} K_i \quad \wedge \bigwedge_{1 \leq j \leq k^c-k-s-1} (\neg w \vee q_j).$$

Note that $|T_c(F)| = s + k + 1 + k^c - k - s - 1 = k^c$ and that $T_c(F)$ is not a 3CNF formula since it contains the $\Leftrightarrow$ symbol. However, it is easy to show that $T_c(F)$ can be written as a 3CNF formula without introducing any extra variables. This boils down to showing that the expression $(r_i \Leftrightarrow (l_i^1 \vee l_i^2))$ can be transformed to an equivalent 3CNF formula without extra variables. This is a simple exercise. As a final observation, $T_c(F)$ can be computed in polynomial time since c is fixed.

We continue by showing that T_c satisfies the given requirements. Let $F' = T_c(F)$. We begin by showing that $Min_1(F') \leq k^4$ if F is satisfiable. Let $\mathcal{M}$ be a satisfying assignment for F. We extend $\mathcal{M}$ to also act on literals in the obvious way: $\mathcal{M}(\neg p) = \mathbf{F}$ if $\mathcal{M}(p) = \mathbf{T}$ and vice versa. Construct a model $\mathcal{M}'$ for F' as follows:

$$
\begin{aligned}
\mathcal{M}'(p) &= \mathcal{M}(p) && p \in P; \\
\mathcal{M}'(w) &= \mathbf{F}; && \\
\mathcal{M}'(q) &= \mathbf{F} && q \in Q; \\
\mathcal{M}'(r_i) &= \begin{cases} \mathbf{F} \text{ if } \mathcal{M}(l_i^1) = \mathcal{M}(l_i^2) = \mathbf{F} \\ \mathbf{T} \text{ otherwise} \end{cases} && 1 \le i \le s;
\end{aligned}
$$

In this case, $Min_1(F') \le k + s \le k + 8k^3 \le k^4$ for k sufficiently large.

Assume to the contrary that F is not satisfiable, that is, no truth assignment can satisfy all clauses in F simultaneously. Note that a clause of the type $(l_i^3 \vee r_i \vee w)$, $1 \le i \le s$, is logically equivalent to the clause $(l_i^1 \vee l_i^2 \vee l_i^3 \vee w)$. This follows from the fact that, by the construction of F', r_i is logically equivalent to $(l_i^1 \vee l_i^2)$. Consequently, the clauses $(l_i^3 \vee r_i \vee w)$, $1 \le i \le s$, in F' cannot be simultaneously satisfied unless w is assigned $\mathbf{T}$. But then q_j, $1 \le j \le k^c - k - s - 1$, must also be assigned $\mathbf{T}$ in order to satisfy F'. Hence, a model $\mathcal{M}'$ for F' which assigns $\mathbf{T}$ to as few propositions as possible has the following appearance:

$$
\begin{aligned}
\mathcal{M}'(p) &= \mathbf{F} \quad p \in P; \\
\mathcal{M}'(w) &= \mathbf{T}; \\
\mathcal{M}'(q) &= \mathbf{T} \quad q \in Q; \\
\mathcal{M}'(r) &= \mathbf{F} \quad r \in R.
\end{aligned}
$$

Thus, $Min_1(F') \ge 1 + k^c - k - s - 1 \ge k^c - k - 8k^3$ since $s \le 8k^3$.

Theorem 4 Min Ones *is not approximable within $n^{1-\epsilon}$ for any $\epsilon > 0$ where n is the number of variables.*

Proof. Arbitrarily choose an $\epsilon > 0$ and let the natural number $c > 6$ satisfy $1 - 6/c \ge \epsilon$. Define $a(n) = n^{5/c}$ and $b(n) = n^{1-1/c}$. Given arbitrary natural numbers i, j, we can decide (in polynomial-time) whether $a(i) < j$ since $a(i) < j$ iff $i^5 < j^c$. By Lemma 3, there exists a polynomial-time reduction T_c from 3Sat to MinOnes with the following properties: For an arbitrary instance I of 3Sat containing k variables,

(i) $T_c(I)$ contains k^c variables.
(ii) $Min_1(T_c(I)) < k^5 = |T_c(I)|^{5/c} = a(|T_c(I)|)$ if I is satisfiable.
(iii) $Min_1(T_c(I)) > k^{c-1} = |T_c(I)|^{1-1/c} = b(|T_c(I)|)$ if I is not satisfiable.

Hence, Min Ones is not approximable within $\alpha(n) = b(n)/a(n) = n^{1-6/c}$ by Lemma 2. Since $1-6/c \geq \epsilon$ and ϵ was arbitrarily chosen, the theorem follows.

Corollary 5 Min Dones *is not approximable within $n^{1-\epsilon}$ for any $\epsilon > 0$, where n is the number of variables.*

4 Maximization Problems

To show the nonapproximability of Max Ones, we use a construction similar to the construction in Lemma 3.

Theorem 6 Max Ones *is not approximable within $n^{1-\epsilon}$ for any $\epsilon > 0$ where n is the number of variables.*

Proof sketch. Let Max_1 denote the optimal value function for Max Ones. Let $c > 4$ be a fixed integer. We give a polynomial-time tranformation U_c on 3CNF formulae satisfying the following: for an arbitrary 3CNF formula F,

(i) $|U_c(F)| = |F|^c$;
(ii) $Max_1(U_c(F)) \leq |F|^4$ if F is not satisfiable;
(iii) $Max_1(U_c(F)) \geq |F|^c - 8|F|^3 - |F|$ otherwise.

Arbitrarily choose a 3CNF formula over the propositions $P = \{p_1, \ldots, p_k\}$. As noted earlier, we can safely assume that $F = C_1 \wedge \ldots \wedge C_s$ where $s \leq 8k^3$. Let w be a fresh proposition and $Q = \{q_1, \ldots, q_{k^c-k-s-1}\}$, $R = \{r_1, \ldots, r_s\}$ be sets of fresh propositions.

For arbitrary clauses $C_i = (l_i^1 \vee l_i^2 \vee l_i^3)$, $1 \leq i \leq s$, let

$$K_i = (r_i \Leftrightarrow (l_i^1 \vee l_i^2)) \wedge (l_i^3 \vee r_i \vee \neg w).$$

Define $U_c(F)$ as follows:

$$U_c(F) = \bigwedge_{1 \leq i \leq s} K_i \;\; \wedge \bigwedge_{1 \leq j \leq k^c-k-s-1} (w \vee \neg q_j).$$

Showing that U_c satisfies the stated requirements is similar to the proof of Lemma 3. Proving the non-approximability result is analogous to the proof of Theorem 4.

Corollary 7 Max Dones *and* Max PB 0/1 Programming *are not approximable within $n^{1-\epsilon}$ for any $\epsilon > 0$, where n is the number of variables.*

Under the additional assumption that NP $\neq$ ZPP, it is possible to show that MAX ONES cannot be approximated within $n^{1-\epsilon}$ even if restricted to 2CNF formulae. Håstad [3] has shown that the CLIQUE problem cannot be approximated within $n^{1-\epsilon}$ (where n denotes the number of nodes in the graph), unless NP = ZPP. The reduction of CLIQUE to MAX ONES is straightforward; for each missing edge in the given graph G (say, between node x and node y), add a clause of the form $(\neg x \vee \neg y)$. Obviously, the optimal value of the resulting MAX ONES instance equals the size of the largest clique in G.

Acknowledgements

Thanks to Viggo Kann, Thomas Drakengren and the anonymous referee for helpful discussions and comments.

References

[1] P. Berman and G. Schnitger. On the complexity of approximating the independent set problem. *Information and Computation* **96** (1992) 77–94.

[2] S. A. Cook. The complexity of theorem-proving procedures, in: *Proc. 3rd Ann. ACM Symp. on Theory of Computing* (1971) 151–158.

[3] J. Håstad. Clique is hard to approximate within $n^{1-\epsilon}$, Electronic Colloquium on Computational Complexity, report TR97–038, 1997. See also an early version in Proc. 37th Ann. IEEE Symp. on Foundations of Computer Science (1996) 627–636.

[4] V. Kann. *On the approximability of NP-complete optimization problems.* PhD thesis, Royal Institute of Technology, Stockholm, 1992.

[5] V. Kann. Polynomially bounded minimization problems that are hard to approximate. *Nordic Journal of Computing* **1** (1994) 317–331.

[6] V. Kann. Strong lower bounds on the approximability of some NPO PB-complete maximization problems, in: *Proc. 20th Int. Symp. on Mathematical Foundations of Computer Science* (1995) 227–236.

[7] C. H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization: Algorithms and Complexity* (Prentice-Hall, New Jersey, 1982).