Parameterizing MAX SNP problems above Guaranteed Values

Meena Mahajan, Venkatesh Raman, and Somnath Sikdar

The Institute of Mathematical Sciences, C.I.T Campus, Taramani, Chennai 600113.
{meena,vraman,somnath}@imsc.res.in

Abstract. We show that every problem in MAX SNP has a lower bound on the optimum solution size that is unbounded and that the above guarantee question with respect to this lower bound is fixed parameter tractable. We next introduce the notion of "tight" upper and lower bounds for the optimum solution and show that the parameterized version of a variant of the above guarantee question with respect to the tight lower bound cannot be fixed parameter tractable unless P = NP, for a class of NP-optimization problems.

1 Introduction

In this paper, we consider the parameterized complexity of NP-optimization problems Q with the following property: for non-trivial instance I of Q, the optimum opt(I), is lower-bounded by an increasing function of the input size. That is, there exists a function $f : \mathbb{N} \rightarrow \mathbb{N}$ which is increasing such that for non-trivial instances I, opt$(I) \geq f(|I|)$. For such an optimization problem Q, the standard parameterized version $\tilde{Q}$ defined below is easily seen to be fixed parameter tractable. For if $k \leq f(|I|)$, we answer 'yes'; else, $f(|I|) < k$ and so $|I| < f^{-1}(k)$[1] and we have a kernel.

$$\tilde{Q} = \{(I, k) \,:\, I \text{ is an instance of } Q \text{ and opt}(I) \geq k\}$$

Thus for such an optimization problem it makes sense to define an "above guarantee" parameterized version $\bar{Q}$ as

$$\bar{Q} = \{(I, k) \,:\, I \text{ is an instance of } Q \text{ and opt}(I) \geq f(|I|) + k\}.$$

Such above guarantee parameterized problems were first considered by Mahajan and Raman in [5]. The problems dealt with by them are MAX SAT and MAX CUT. An instance of the MAX SAT problem is a boolean formula ϕ in conjunctive normal form and the standard parameterized version asks whether ϕ has at least k satisfiable clauses, k being the parameter. Since any boolean formula ϕ with m clauses has at least $\lceil m/2 \rceil$ satisfiable clauses (see Motwani and Raghavan [6]), by the above argument, this problem is fixed parameter tractable. The above guarantee MAX SAT question considered in [5] asks whether a given formula ϕ has at least $\lceil m/2 \rceil + k$ satisfiable clauses, with k as parameter. This was shown to be fixed parameter tractable.

[1] Assuming f to be invertible; the functions considered in this paper are.

The standard parameterized version of the MAX CUT problem asks whether an input graph G has a cut of size at least k, where k is the parameter. This problem is also fixed parameter tractable since any graph G with m edges has a cut of size $\lceil m/2 \rceil$. The above guarantee MAX CUT question considered in [5] asks whether an input graph G on m edges has a cut of size at least $\lceil m/2 \rceil + k$, where k is the parameter. This problem was shown to be fixed parameter tractable too.

In this paper, we consider above guarantee questions for problems in the class MAX SNP. This paper is structured as follows. In Section 2, we introduce the necessary ideas about parameterized complexity and state some basic definitions needed in the rest of the paper. In Section 3, we show that every problem in the class MAX SNP has a guaranteed lower bound that is an unbounded function of the input size and that the above guarantee problem with respect to this lower bound is fixed parameter tractable. In Section 4, we define a notion of *tight lower bound* and show that a variant of the above guarantee question with respect to tight lower bounds is hard (unless P = NP) for a number of NP-maximization problems. Finally in Section 5, we end with a few concluding remarks.

2 Preliminaries

We briefly introduce the necessary concepts concerning optimization problems and parameterized complexity.

To begin with, a parameterized problem is a subset of $\Sigma^* \times \mathbb{N}$, where Σ is a finite alphabet and $\mathbb{N}$ is the set of natural numbers. An instance of a parameterized problem is therefore a pair (I, k), where k is the parameter. In the framework of parameterized complexity, the run time of an algorithm is viewed as a function of two quantities: the size of the problem instance *and* the parameter. A parameterized problem is said to be *fixed parameter tractable* (fpt) if there exists an algorithm for the problem with time complexity $O(f(k) \cdot |I|^{O(1)})$, where f is a recursive function of k alone. The class FPT consists of all fixed parameter tractable problems.

A parameterized problem π_1 is *fixed-parameter-reducible* to a parameterized problem π_2 if there exist functions $f, g : \mathbb{N} \to \mathbb{N}$, $\Phi : \Sigma^* \times \mathbb{N} \to \Sigma^*$ and a polynomial $p(\cdot)$ such that for any instance (I, k) of π_1, $(\Phi(I, k), g(k))$ is an instance of π_2 computable in time $f(k) \cdot p(|I|)$ and $(I, k) \in \pi_1$ if and only if $(\Phi(I, k), g(k)) \in \pi_2$.

An NP-optimization problem Q is a 4-tuple $Q = \{\mathscr{I}, S, V, \text{opt}\}$, where

1. $\mathscr{I}$ is the set of input instances. (w.l.o.g., $\mathscr{I}$ can be recognized in polynomial time.)
2. $S(x)$ is the set of feasible solutions for the input $x \in \mathscr{I}$.
3. V is a polynomial-time computable function called the *cost function* and for each $x \in \mathscr{I}$ and $y \in S(x)$, $V(x, y) \in \mathbb{N}$.
4. opt $\in$ {max, min}.
5. The following decision problem (called the *underlying decision problem*) is in NP: Given $x \in \mathscr{I}$ and an integer k, does there exist a feasible solution $y \in S(x)$ such that $V(x, y) \geq k$, when Q is a maximization problem (or, $V(x, y) \leq k$, when Q is a minimization problem).

The class MAX SNP was defined by Papadimitriou and Yannakakis [7] using logical expressiveness. They showed that a number of interesting optimization problems such as MAX 3-SAT, INDEPENDENT SET-B, MAX CUT, MAX k-COLORABLE SUBGRAPH etc. lie in this class. They also introduced the notion of MAX SNP-completeness by using a reduction known as the *L-reduction*. We define this next.

Let Q_1 and Q_2 be two optimization (maximization or minimization) problems. We say that Q_1 L-reduces to Q_2 if there exist polynomial-time computable functions f, g, and constants $\alpha, \beta > 0$ such that for each instance I_1 of Q_1:

1. $f(I_1) = I_2$ is an instance of Q_2, such that $\text{opt}(I_2) \leq \alpha \cdot \text{opt}(I_1)$.
2. Given any solution y_2 of I_2, g maps (I_2, y_2) to a solution y_1 of I_1 such that $|V(I_1, y_1) - \text{opt}(I_1)| \leq \beta \cdot |V(I_2, y_2) - \text{opt}(I_2)|$

We call such an L-reduction from Q_1 to Q_2 an $\langle f, g, \alpha, \beta\rangle$ reduction.

A problem Q is MAX SNP-*hard* if every problem in the class MAX SNP L-reduces to Q. A problem Q is MAX SNP-*complete*, if Q is in MAX SNP and is MAX SNP-hard. Cai and Chen [1] established that all maximization problems in the class MAX SNP are fixed parameter tractable. In the next section, we show that for all problems in MAX SNP, a certain above-guarantee question is also fixed parameter tractable.

3 Parameterizing above Guaranteed Values

Consider the problem MAX 3-SAT which is complete for the class MAX SNP. An instance of MAX 3-SAT is a boolean formula f in conjunctive normal form with at most three literals per clause. As already stated, any boolean formula with m clauses has at least $\lceil m/2 \rceil$ satisfiable clauses, and the following above guarantee parameterized problem is fixed parameter tractable.

L = $\{(f, k)$: f is a MAX 3-SAT instance and $\exists$ an assignment satisfying at least $k + \lceil m/2 \rceil$ clauses of the formula f $\}$.

Since MAX 3-SAT is MAX SNP-complete and has a guaranteed lower bound, we have

Proposition 1 *If Q is in* MAX SNP*, then for each instance x of Q there exists a positive number γ_x such that $\gamma_x \leq opt(x)$. Further, if Q is* NP*-hard, then the function $\gamma : x \rightarrow \gamma_x$ is unbounded, assuming* P $\neq$ NP.

Proof. Let Q be a problem in MAX SNP and let $\langle f, g, \alpha, \beta\rangle$ be an L-reduction from Q to MAX 3-SAT. Then for an instance x of Q, $f(x)$ is an instance of MAX 3-SAT such that $\text{opt}(f(x)) \leq \alpha \cdot \text{opt}(x)$. If $f(x)$ is a formula with m clauses, then $\lceil m/2 \rceil \leq \text{opt}(f(x))$ and therefore $\text{opt}(x)$ is bounded below by $\lceil m/2 \rceil/\alpha$. This proves that each instance x of Q has a lower bound. We can express this lower bound in terms of the parameters of the L-reduction. Since $f(x)$ is an instance of MAX 3-SAT, we can take the size of $f(x)$ to be m. Then $\gamma_x = |f(x)|/(2 \cdot \alpha)$. Further, note that if m is not unbounded, then we can solve Q in polynomial time via this reduction. ■

Note that this lower bound γ_x depends on the complete problem to which we reduce Q. By changing the complete problem, we might construct different lower bounds for the problem at hand. It is also conceivable that there exist more than one L-reduction between two optimization problems. Different L-reductions should give different lower bounds. Thus the polynomial-time computable lower bound that we exhibit in Proposition 1 is a special lower bound obtained from a specific L-reduction to a specific complete problem (MAX 3-SAT) for the class MAX SNP. Call the lower bound of Proposition 1 a MAX 3-SAT-lower bound for the problem Q.

Since the above guarantee parameterized version L of MAX 3-SAT is known to be in FPT, we immediately have the following.

Theorem 1. *For a maximization problem Q in MAX SNP, let $\langle f, g, \alpha, \beta \rangle$ be an L-reduction from Q to MAX 3-SAT, and for an instance x of Q, let γ_x represent the corresponding MAX 3-SAT-lower bound. Then the following problem is in FPT:*

$$L_Q = \{\langle x, k\rangle : \text{ } x \text{ is an instance of } Q \text{ and } opt(x) \geq \gamma_x + k\}$$

Proof. We make use of the fact that there exists a fixed parameter tractable algorithm $\mathscr{A}$ for MAX 3-SAT which takes as input, a pair of the form $\langle \psi, k\rangle$, and in time $O(|\psi| + h(k))$, returns YES if there exists an assignment to the variables of ψ that satisfies at least $\lceil m/2 \rceil + k$ clauses, and NO otherwise. See [5, 9] for such algorithms.

Consider an instance $\langle x, k\rangle$ of L_Q. Then $f(x)$ is an instance of MAX 3-SAT. Let $f(x)$ have m clauses. Then the guaranteed lower bound for the instance x of Q, $\gamma_x = \frac{m}{2\alpha}$, and $\text{opt}(f(x)) \leq \alpha \cdot \text{opt}(x)$. Apply algorithm $\mathscr{A}$ on input $\langle f(x), k\alpha\rangle$. If $\mathscr{A}$ outputs YES, then $\text{opt}(f(x)) \geq m/2 + k \cdot \alpha$, implying $\text{opt}(x) \geq \frac{m}{2 \cdot \alpha} + k = \gamma_x + k$. Thus $\langle x, k\rangle \in L_Q$.

If $\mathscr{A}$ answers NO, then $\lceil \frac{m}{2} \rceil \leq \text{opt}(f(x)) < \lceil \frac{m}{2} \rceil + k\alpha$. Apply algorithm $\mathscr{A}$ $k\alpha$ times on inputs $(f(x), 1), (f(x), 2), \ldots, (f(x), k\alpha)$ to obtain $\text{opt}(f(x))$. Let $c' = \text{opt}(f(x))$. Then use algorithm g of the L-reduction to obtain a solution to x with cost c. By the definition of L-reduction, we have $|c - \text{opt}(x)| \leq \beta \cdot |c' - \text{opt}(f(x))|$. But since $c' = \text{opt}(f(x))$, it must be that $c = \text{opt}(x)$. Therefore we simply need to compare c with $\gamma_x + k$ to check whether $\langle x, k\rangle \in L_Q$.

The total time complexity of the above algorithm is $O(k\alpha \cdot (|f(x)| + h(k\alpha)) + p_1(|x|) + p_2(|f(x)|))$, where $p_1(\cdot)$ is the time taken by algorithm f to transform an instance of Q to an instance of MAX 3-SAT, and $p_2(\cdot)$ is the time taken by g to output its answer. Thus the algorithm that we outlined is indeed an FPT algorithm for L_Q. ■

Note that the proof of Proposition 1 also shows that every minimization problem in MAX SNP has a MAX 3-SAT-lower bound. For minimization problems whose optimum is lower bounded by some function of the input, it makes sense to ask how far removed the optimum is with respect to the lower bound. The parameterized question asks whether for a given input x, $\text{opt}(x) \leq \gamma_x + k$, with k as parameter. The following result can be proved similarly to Theorem 1.

Theorem 2. *For a minimization problem Q in MAX SNP, let $\langle f, g, \alpha, \beta \rangle$ be an L-reduction from Q to MAX 3-SAT, and for an instance x of Q, let γ_x represent the corresponding MAX 3-SAT-lower bound. Then the following problem is in FPT:*

$$L_Q = \{\langle x, k\rangle : \text{ } x \text{ is an instance of } Q \text{ and } opt(x) \leq \gamma_x + k\}$$

Examples of minimization problems in MAX SNP include VERTEX COVER-B and DOMINATING SET-B which are, respectively, the restriction of the VERTEX COVER and the DOMINATING SET problems to graphs whose vertex degree is bounded by B.

4 Hardness Results

For an optimization problem, the question of whether the optimum is at least lower bound + k, for some lower bound and with k as parameter, is not always interesting because if the lower bound is "loose" then the problem is trivially fixed parameter tractable. For instance, for the MAX CUT problem, the question of whether an input graph has a cut of size at least $\frac{m}{2} + k$ is fpt since any graph G with m edges, n vertices and c components has a cut of size at least $\frac{m}{2} + \lceil \frac{n-c}{4} \rceil$ [8]. Thus if $k \leq \lceil \frac{n-c}{4} \rceil$, we answer YES; else, $\lceil \frac{n-c}{4} \rceil < k$ and we have a kernel.

We therefore examine the notion of a *tight lower bound* and the corresponding above guarantee question. A tight lower bound is essentially the best possible lower bound on the optimum solution size. For the MAX SAT problem, this lower bound is $m/2$: if ϕ is an instance of MAX SAT, then $\text{opt}(\phi) \geq m/2$, and there are infinitely many instances for which the optimum is *exactly* $m/2$. This characteristic motivates the next definition.

Definition 1 (Tight Lower Bound) Let $Q = \{\mathscr{I}, S, V, \text{opt}\}$ be an NP-optimization problem and let $f : \mathbb{N} \to \mathbb{N}$. We say that f is a ***tight lower bound*** for Q if the following conditions hold:

1. $f(|I|) \leq \text{opt}(I)$ for all $I \in \mathscr{I}$.
2. There exists an infinite family of instances $\mathscr{I}' \subseteq \mathscr{I}$ such that $\text{opt}(I) = f(|I|)$ for all $I \in \mathscr{I}'$.

Note that we define the lower bound to be a function of the *input size* rather than the input itself. This is in contrast to the lower bound of Proposition 1 which depends on the input instance. We can define the notion of a tight *upper* bound analogously.

Definition 2 (Tight Upper Bound) Let $Q = \{\mathscr{I}, S, V, \text{opt}\}$ be an NP-optimization problem and let $g : \mathbb{N} \to \mathbb{N}$. We say that g is a ***tight upper bound*** for Q if the following conditions hold:

1. $\text{opt}(I) \leq g(|I|)$ for all $I \in \mathscr{I}$.
2. There exists an infinite family of instances $\mathscr{I}' \subseteq \mathscr{I}$ such that $\text{opt}(I) = g(|I|)$ for all $I \in \mathscr{I}'$.

Some example optimization problems which have tight lower and upper bounds are given below. The abbreviations TLB and TUB stand for tight lower bound and tight upper bound, respectively.

1. MAX EXACT c-SAT

INSTANCE	A boolean formula F with n variables and m clauses with each clause having *exactly* c distinct literals.
QUESTION	Find the maximum number of simultaneously satisfiable clauses.
BOUNDS	TLB $= (1 - \frac{1}{2^c})m$; TUB $= m$.

The expected number of clauses satisfied by the random assignment algorithm is $(1 - \frac{1}{2^c})m$; hence the lower bound. To see tightness, note that if $\phi(x_1, \ldots, x_c)$ denotes the EXACT c-SAT formula comprising of all possible combinations of c variables, then ϕ has 2^c clauses of which exactly $2^c - 1$ clauses are satisfiable. By taking disjoint copies of this formula one can construct EXACT c-SAT instances of arbitrary size with exactly $(1 - \frac{1}{2^c})m$ satisfiable clauses.

2. CONSTRAINT SATISFACTION PROBLEM (CSP)

INSTANCE A system of m linear equations modulo 2 in n variables, together with positive weights w_i, $1 \leq i \leq m$.

QUESTION Find an assignment to the variables that maximizes the total weight of the satisfied equations.

BOUNDS TLB $= \frac{W}{2}$, where $W = \sum_{i=1}^{m} w_i$; TUB $= W$.

If we use $\{+1, -1\}$-notation for boolean values with -1 corresponding to true then we can write the ith equation of the system as $\prod_{j \in \alpha_i} x_j = b_i$, where each α_i is a subset of $[n]$ and $b_i \in \{+1, -1\}$. To see that we can satisfy at least half the equations in the weighted sense, we assign values to the variables sequentially and simplify the system as we go along. When we are about to give a value to x_j, we consider all equations reduced to the form $x_j = b$, for a constant b. We choose a value for x_j satisfying at least half (in the weighted sense) of these equations. This procedure of assigning values ensures that we satisfy at least half the equations in the weighted sense. A tight lower bound instance, in this case, is a system consisting of pairs $x_j = b_i, x_j = \bar{b}_i$, with each equation of the pair assigned the same weight. See [3] for more details.

3. MAX INDEPENDENT SET-B

INSTANCE A graph G with n vertices such that the degree of each vertex is bounded by B.

QUESTION Find a maximum independent set of G.

BOUNDS TLB $= \frac{n}{B+1}$; TUB $= n$.

A graph whose vertex degree is bounded by B can be colored using $B + 1$ colors, and in any valid coloring of the graph, the vertices that get the same color form an independent set. By the pigeonhole principle, there exists an independent set of size at least $n/(B + 1)$. The complete graph K_{B+1} on $B + 1$ vertices has an independence number of $\frac{n}{B+1}$. By taking disjoint copies of K_{B+1} one can construct instances of arbitrary size with independence number exactly $\frac{n}{B+1}$.

4. MAX PLANAR INDEPENDENT SET

INSTANCE A planar graph G with n vertices and m edges.

QUESTION Find a maximum independent set of G.

BOUNDS TLB $= \frac{n}{4}$; TUB $= n$.

A planar graph is 4-colorable, and in any valid 4-coloring of the graph, the vertices that get the same color form an independent set. By the pigeonhole principle, there exists an independent set of size at least $\frac{n}{4}$. A disjoint set of K_4's can be use to construct arbitrary sized instances with independence number exactly $\frac{n}{4}$.

5. MAX ACYCLIC DIGRAPH

INSTANCE A directed graph G with n vertices and m edges.

QUESTION Find a maximum acyclic subgraph of G.

BOUNDS TLB $= \frac{m}{2}$; TUB $= m$.

To see that any digraph with m arcs has an acyclic subgraph of size $\frac{m}{2}$, place the vertices $v_1, \ldots, v_n$ of G on a line in that order with arcs (v_i, v_j), $i < j$, drawn above the line and arcs (v_i, v_j), $i > j$, drawn below the line. Clearly, by deleting all arcs either above or below the line we obtain an acyclic digraph. By the pigeonhole principle, one of these two sets must have size at least $\frac{m}{2}$. To see that this bound is tight, consider the digraph D on n vertices: $v_1 \leftrightarrows v_2 \leftrightarrows v_3 \leftrightarrows \ldots \leftrightarrows v_n$ which has a maximum acyclic digraph of size exactly $\frac{m}{2}$. Since n is arbitrary, we have an infinite set of instances for which the optimum matches the lower bound exactly.

6. MAX PLANAR SUBGRAPH

INSTANCE A connected graph G with n vertices and m edges.

QUESTION Find an edge-subset E' of maximum size such that $G[E']$ is planar.

BOUNDS TLB $= n - 1$; TUB $= 3n - 6$.

Any spanning tree of G has $n - 1$ edges; hence any maximum planar subgraph of G has at least $n - 1$ edges. This bound is tight as the family of all trees achieves this lower bound. An upper bound is $3n - 6$ which is tight since for each n, a maximal planar graph on n vertices has exactly $3n - 6$ edges.

7. MAX CUT

INSTANCE A graph G with n vertices, m edges and c components.

QUESTION Find a maximum cut of G.

BOUNDS TLB $= \frac{m}{2} + \lceil \frac{n-c}{4} \rceil$; TUB $= m$.

The lower bound for the cut size was proved by Poljak and Turzík [8]. This bound is tight for complete graphs. The upper bound is tight for bipartite graphs.

A natural question to ask in the above-guarantee framework is whether the language

$$L = \{\langle I, k\rangle : \text{ opt}(I) \geq \text{TLB}(I) + k\}$$

is in FPT. The parameterized complexity of such a question is not known for most problems. To the best of our knowledge, this question has been resolved only for the MAX SAT and MAX c-SAT problems [5] and, very recently, for the LINEAR ARRANGEMENT problem [2].

In this section, we study a somewhat different, but related, parameterized question: Given an NP-maximization problem Q which has a tight lower bound (TLB) a function of the input size, what is the parameterized complexity of the following question?

$$Q(\epsilon) = \{\langle I, k\rangle : \text{ opt}(I) \geq \text{TLB}(I) + \epsilon \cdot |I| + k\}$$

Here $|I|$ denotes the input size, ϵ is some fixed positive rational and k is the parameter. We show that this question is not fixed parameter tractable for a number of problems, unless P = NP.

Theorem 3. *For any problem Q in the following, the $Q(\epsilon)$ problem is not fixed parameter tractable unless* P = NP*:*

	Problem	$\text{TLB}(I) + \epsilon \cdot \lvert I\rvert + k$	*Range of* ϵ
1.	MAX SAT	$(\frac{1}{2} + \epsilon)m + k$	$0 < \epsilon < \frac{1}{2}$
2.	MAX c-SAT	$(\frac{1}{2} + \epsilon)m + k$	$0 < \epsilon < \frac{1}{2}$
3.	MAX EXACT c-SAT	$(1 - \frac{1}{2^c} + \epsilon)m + k$	$0 < \epsilon < \frac{1}{2^c}$
4.	CSP	$(\frac{1}{2} + \epsilon)m + k$	$0 < \epsilon < \frac{1}{2}$
5.	PLANAR INDEPENDENT SET	$(\frac{1}{4} + \epsilon)n + k$	$0 < \epsilon < \frac{3}{4}$
6.	INDEPENDENT SET-B	$(\frac{1}{B+1} + \epsilon)n + k$	$0 < \epsilon < \frac{B}{B+1}$
7.	MAX ACYCLIC SUBGRAPH	$(\frac{1}{2} + \epsilon)m + k$	$0 < \epsilon < \frac{1}{2}$
8.	MAX PLANAR SUBGRAPH	$(1 + \epsilon)n - 1 + k$	$0 < \epsilon < 2$
9.	MAX CUT	$\frac{m}{2} + \lceil \frac{n-c}{4} \rceil + \epsilon n + k$	$0 < \epsilon < \frac{1}{4}$
10.	MAX DICUT	$\frac{m}{4} + \sqrt{\frac{m}{32} + \frac{1}{256}} - \frac{1}{16} + \epsilon m + k$	$0 < \epsilon < \frac{3}{4}$

The proof, in each case, follows this outline: Assume that for some ϵ in the specified range, $Q(\epsilon)$ is indeed in FPT. Now consider an instance $\langle I, s\rangle$ of the underlying decision version of Q. Here is a P-time procedure for deciding it. If $s \leq \text{TLB}$, then the answer is trivially YES. If s lies between TLB and $\text{TLB} + \epsilon\lvert I\rvert$, then "add" a gadget of suitable size corresponding to the TUB, to obtain an equivalent instance $\langle I', s'\rangle$. This increases the input size, but since we are adding a gadget whose optimum value matches the upper bound, the increase in the optimum value of I' is more than proportional, so that now s' exceeds $\text{TLB} + \epsilon\lvert I'\rvert$. If s already exceeds $\text{TLB} + \epsilon\lvert I\rvert$, then "add" a gadget of suitable size corresponding to the TLB, to obtain an equivalent instance $\langle I', s'\rangle$. This increases the input size faster than it boosts the optimum value of I', so that now s' exceeds $\text{TLB} + \epsilon\lvert I'\rvert$ by only a constant, say c_1. Use the hypothesized fpt algorithm for $Q(\epsilon)$ with input $\langle I', c_1\rangle$ to correctly decide the original question.

Rather than proving the details for each item separately, we use this proof sketch to establish a more general theorem (Theorem 4 below) which automatically implies items 1 through 10 above. We first need some definitions.

Definition 3 (Dense Set) Let $Q = \{\mathscr{I}, S, V, \text{opt}\}$ be an NPO problem. A set of instances $\mathscr{I}' \subseteq \mathscr{I}$ is said to be ***dense with respect to a set of conditions*** $\mathcal{C}$ if there exists a constant $c \in \mathbb{N}$ such that for all closed intervals $[a, b] \subseteq \mathbb{R}^+$ of length $\lvert b - a\rvert \geq c$, there exists an instance $I \in \mathscr{I}'$ with $\lvert I\rvert \in [a, b]$ such that I satisfies all the conditions in $\mathcal{C}$. Further, if such an I can be found in polynomial time (polynomial in b), then $\mathscr{I}'$ is said to be ***dense poly-time uniform with respect to*** $\mathcal{C}$.

For example, for the MAXIMUM ACYCLIC SUBGRAPH problem, the set of all oriented digraphs is dense (poly-time uniform) with respect to the condition: opt(G) $= \lvert E(G)\rvert$.

We also need the notion of a partially additive NP-optimization problem.

Definition 4 (Partially Additive Problems) An NPO problem $Q = \{\mathscr{I}, S, V, \text{opt}\}$ is said to be ***partially additive*** if there exists an operator $+$ which maps a pair of instances I_1 and I_2 to an instance $I_1 + I_2$ such that

1. $|I_1 + I_2| = |I_1| + |I_2|$, and
2. $\text{opt}(I_1 + I_2) = \text{opt}(I_1) + \text{opt}(I_2)$.

A partially additive NPO problem that also satisfies the following condition is said to be additive in the framework of Khanna, Motwani et al [4]: there exists a polynomial-time computable function f that maps any solution s of $I_1 + I_2$ to a pair of solutions s_1 and s_2 of I_1 and I_2, respectively, such that $V(I_1 + I_2, s) = V(I_1, s_1) + V(I_1, s_2)$.

For many graph-theoretic optimization problems, the operator $+$ can be interpreted as disjoint union. Then the problems MAX CUT, MAX INDEPENDENT SET-B, MINIMUM VERTEX COVER, MINIMUM DOMINATING SET, MAXIMUM DIRECTED ACYCLIC SUBGRAPH, MAXIMUM DIRECTED CUT are partially additive. For other graph-theoretic problems, one may choose to interpret $+$ as follows: given graphs G and H, $G + H$ refers to a graph obtained by placing an edge between some (possibly arbitrarily chosen) vertex of G and some (possibly arbitrarily chosen) vertex of H. The MAX PLANAR SUBGRAPH problem is partially additive with respect to both these interpretations of $+$. For boolean formulae ϕ and ψ in conjunctive normal form with disjoint sets of variables, define $+$ as the conjunction $\phi \wedge \psi$. Then the MAX SAT problem is easily seen to be partially additive.

Let $Q = \{\mathscr{I}, S, V, \max\}$ be an NP-maximization problem with tight lower bound $f : \mathbb{N} \to \mathbb{N}$ and tight upper bound $g : \mathbb{N} \to \mathbb{N}$. We assume that both f and g are increasing and satisfy the following conditions

P1 For all $a, b \in \mathbb{N}$, $f(a + b) \leq f(a) + f(b) + c^*$, where c^* is a constant (positive or negative),

P2 There exists $n_0 \in \mathbb{N}$ and $r \in \mathbb{Q}^+$ such that $g(n) - f(n) > rn$ for all $n \geq n_0$.

Property $P1$ is satisfied by linear functions ($f(n) = an + b$) and by some sub-linear functions such as $\sqrt{n}, \log n, \frac{1}{n}$. Note that a super-linear function cannot satisfy $P1$. Define $\mathscr{R}$ to be the set

$$\mathscr{R} = \{r \in \mathbb{Q}^+ : g(n) - f(n) > rn \text{ for all } n \geq n_0\},$$

and $p = \sup \mathscr{R}$. For $0 < \epsilon < p$, define $Q(\epsilon)$ as follows

$$Q(\epsilon) = \{(I, k) : I \in \mathscr{I} \text{ and } \max(I) \geq f(|I|) + \epsilon|I| + k\}.$$

Note that for $0 < \epsilon < p$, the function h defined by $h(n) = g(n) - f(n) - \epsilon n$ is strictly increasing, and $h(n) > 0 \; \forall n \geq n_0 \in \mathbb{N}$.

Theorem 4. *Let $Q = \{\mathscr{I}, S, V, \max\}$ be a polynomially bounded NP-maximization problem such that the following conditions hold.*

1. *Q is partially additive.*
2. *Q has a tight lower bound (TLB) f, which is increasing and satisfies condition $P1$. The infinite family of instances $\mathscr{I}'$ witnessing the tight lower bound is dense poly-time uniform with respect to the condition $\max(I) = f(|I|)$.*
3. *Q has a tight upper bound (TUB) g, which with f satisfies condition $P2$. The infinite family of instances $\mathscr{I}'$ witnessing the tight upper bound is dense poly-time uniform with respect to the condition $\max(I) = g(|I|)$.*

4. *The underlying decision problem $\tilde{Q}$ of Q is* NP*-hard.*

For $0 < \epsilon < p$, define $Q(\epsilon)$ to be the following parameterized problem

$$Q(\epsilon) = \{(I, k) : \max(I) \geq f(|I|) + \epsilon|I| + k\}$$

where $p = \sup \mathscr{R}$. If $Q(\epsilon)$ is FPT for any $0 < \epsilon < p$, then P = NP.

Proof. Suppose that for some $0 < \epsilon < p$, the parameterized problem $Q(\epsilon)$ is fixed parameter tractable and let $\mathscr{A}$ be an fpt algorithm for it with run time $O(t(k)\text{poly}(|I|))$. We will use $\mathscr{A}$ to solve the underlying decision problem of Q in polynomial time.

Let (I, s) be an instance of the decision version of Q. Then (I, s) is a YES-instance if and only if $\max(I) \geq s$. We consider three cases and proceed as described below.

Case 1: $s \leq f(|I|)$.
Since $\max(I) \geq f(|I|)$, we answer YES.

Case 2: $f(|I|) < s < f(|I|) + \epsilon|I|$.
In this case, we claim that we can transform the input instance (I, s) into an 'equivalent' instance (I', s') such that

1. $f(|I'|) + \epsilon|I'| \leq s'$.
2. $|I'| = \text{poly}(|I|)$.
3. $\text{opt}(I) \geq s$ if and only if $\text{opt}(I') \geq s'$.

This will show that we can, without loss of generality, go to Case 3 below directly. Add a TUB instance I_1 to I. Define $I' = I + I_1$ and $s' = s + g(|I_1|)$. Then it is easy to see that $\max(I) \geq s$ if and only if $\max(I') \geq s'$. We want to choose I_1 such that $f(|I'|) + \epsilon|I'| \leq s'$. Since $|I'| = |I| + |I_1|$ and $s' = s + g(I_1)$, and since $f(|I|) < s$, it suffices to choose I_1 satisfying

$$f(|I| + |I_1|) + \epsilon|I| + \epsilon|I_1| \leq f(|I|) + g(|I_1|)$$

By Property $P1$, we have $f(|I| + |I_1|) \leq f(|I|) + f(|I_1|) + c^*$, so it suffices to satisfy

$$f(|I_1|) + c^* + \epsilon|I| + \epsilon|I_1| \leq g(|I_1|)$$

By Property P2 we have $g(|I_1|) > f(|I_1|) + p|I_1|$, so it suffices to satisfy

$$c^* + \epsilon|I| \leq (p - \epsilon)|I_1|$$

Such an instance I_1 (of size polynomial in $|I|$) can be chosen because $0 < \epsilon < p$, and because the tight upper bound is polynomial-time uniform dense.

Case 3: $f(|I|) + \epsilon|I| \leq s$
In this case, we transform the instance (I, s) into an instance (I', s') such that

1. $f(|I'|) + \epsilon|I'| + c_1 = s'$, where $0 \leq c_1 \leq c_0$ and c_0 is a fixed constant.
2. $|I'| = \text{poly}(|I|)$.
3. $\max(I') \geq s'$ if and only if $\max(I) \geq s$.

We then run algorithm $\mathscr{A}$ with input (I', c_1). Algorithm $\mathscr{A}$ answers YES if and only if $\max(I') \geq s'$. By condition 3 above, this happens if and only if $\max(I) \geq s$. This takes time $O(t(c_1) \cdot \text{poly}(|I'|))$.

We want to obtain I' by adding a TLB instance I_1 to I. What if addition of any TLB instance yields an I' with $s' < f(I') + \epsilon|I'|$? In this case, s must already be very close to $f(|I|) + \epsilon|I|$; the difference $k \triangleq s - f(|I|) - \epsilon|I|$ must be at most $\epsilon d + c^*$, where d is the size of the smallest TLB instance I_0. (Why? Add I_0 to I to get $s + f(d) < f(|I| + d) + \epsilon(|I| + d)$; applying property P1, we get $s + f(d) < f(|I|) + f(d) + c^* + \epsilon|I| + \epsilon d$, and so $k < c^* + \epsilon d$.) In such a case, we can use the fpt algorithm $\mathscr{A}$ with input (I, k) directly to answer the question "Is $\max(I) \geq s$?" in time $O(t(\epsilon d + c^*) \cdot \text{poly}(|I|))$.

So now assume that $k \geq c^* + \epsilon d$, and it is possible to add TLB instances to $|I|$. Since f is an increasing function, there is a *largest* TLB instance I_1 we can add to I to get I' while still satisfying $s' \geq f(I') + \epsilon|I'|$. The smallest TLB instance bigger than I_1 has size at most $|I_1| + c$, where c is the constant that appears in the definition of density. We therefore have the following inequalities

$$f(|I'|) + \epsilon|I'| \leq s' < f(|I'| + c) + \epsilon(|I'| + c).$$

Since f is increasing and satisfies property P1, we have $[f(|I'| + c) + \epsilon(|I'| + c)] - [f(|I'|) + \epsilon|I'|] \leq f(c) + c^* + \epsilon c \triangleq c_0$, and hence $s' = f(|I'|) + \epsilon|I'| + c_1$, where $0 \leq c_1 \leq c_0$. Note that c_0 is a constant independent of the input instance (I, s). Also, since Q is a polynomially bounded problem, $|I_1|$ is polynomially bounded in $|I|$. ■

Remark. Note that there are some problems, notably MAX 3-SAT, for which the constant c_0 in Case 3 of the proof above, is 0. For such problems, the proof of Theorem 4 actually proves that the problem $Q' = \{(I, k) : \max(I) \geq f(|I|) + \epsilon|I|\}$ is NP-hard. But in general, the constant $c_0 \geq 1$ and so this observation cannot be generalized.

We can extend Theorem 4 to minimization problems. For a minimization problem $Q = \{\mathscr{I}, S, V, \min\}$, we need the tight lower bound $f : \mathbb{N} \to \mathbb{N}$ and tight upper bound $g : \mathbb{N} \to \mathbb{N}$ to be increasing functions and satisfy the following conditions

P3 For all $a, b \in \mathbb{N}$, $g(a + b) \leq g(a) + g(b) + c^*$, where c^* is a constant,
P4 There exists $r \in \mathbb{Q}^+$ such that $g(n) - f(n) > rn$ for all $n \geq n_0$ for some $n_0 \in \mathbb{N}$.

Define $\mathscr{R}$ to be the set

$$\mathscr{R} = \{r \in \mathbb{Q}^+ : g(n) - f(n) > rn \text{ for all } n \geq n_0\},$$

and $p = \sup \mathscr{R}$. For $0 < \epsilon < p$, define $Q(\epsilon)$ as follows

$$Q(\epsilon) = \{(I, k) : I \in \mathscr{I} \text{ and } \min(I) \leq g(|I|) - \epsilon|I| - k\}.$$

For minimization problems, we have the following

Theorem 5. *Let $Q = \{\mathscr{I}, S, V, min\}$ be a polynomially bounded NP-minimization problem such that the following conditions hold.*

1. *Q is partially additive.*

2. *Q has a tight lower bound (*TLB*) f such that the infinite family of instances $\mathscr{I}'$ witnessing the tight lower bound is dense poly-time uniform with respect to the condition* $\min(I) = f(|I|)$.
3. *Q has a tight upper bound (*TUB*) g which is increasing, satisfies condition $P3$, and with f satisfies $P4$. The infinite family of instances $\mathscr{I}'$ witnessing the tight upper bound is dense poly-time uniform with respect to the condition* $\min(I) = g(|I|)$.
4. *The underlying decision problem $\tilde{Q}$ of Q is* NP*-hard.*

For $0 < \epsilon < p$, define $Q(\epsilon)$ to be the following parameterized problem

$$Q(\epsilon) = \{(I, k) : \ I \in \mathscr{I} \text{ and } \min(I) \leq g(|I|) - \epsilon|I| - k\}$$

where $p = \sup \mathscr{R}$. If $Q(\epsilon)$ is FPT for any $0 < \epsilon < p$, then P = NP.

The proof of this is similar to that of Theorem 4 and is omitted.

5 Conclusion

We have shown that every problem in MAX SNP has a lower bound on the optimal solution size that is unbounded and that the above guarantee question with respect to that lower bound is in FPT. We have also shown that the $\text{TLB}(I) + \epsilon \cdot |I| + k$ question is hard for a general class that includes a number of NP-maximization problems. However we do not know the parameterized complexity of tight lower bound $+ \, k$ questions for most NPO problems. In particular, apart from MAX SAT, MAX c-SAT and LINEAR ARRANGEMENT, this question is open for the rest of the problems stated in Theorem 3. It would be interesting to explore the parameterized complexity of these problems and above guarantee problems in general.

References

1. L. CAI AND J. CHEN. *On Fixed-Parameter Tractability and Approximability of NP Optimization Problems*. Jour. Comput. Sys. Sci. 54(3): 465-474 (1997).
2. G. GUTIN, A. RAFIEY, S. SZEIDER AND A. YEO. *The Linear Arrangement Problem Parameterized Above Guaranteed Value*. Available at: http://arxiv.org/abs/cs.DS/0511030
3. J. HÅSTAD AND S. VENKATESH. *On the Advantage Over a Random Assignment*. Proceedings of the 34th Annual ACM Symposium on Theory of Computing, 2002, pages 43-52.
4. S. KHANNA, R. MOTWANI, M. SUDAN AND U. VAZIRANI. *On Syntactic Versus Computational Views of Approximability*. SIAM Jour. Computing. Vol. 28, No. 1, pp 164-191.
5. M. MAHAJAN AND V. RAMAN. *Parameterizing above Guaranteed Values: MaxSat and MaxCut*, Journal of Algorithms **31**, 335-354 (1999).
6. R. MOTWANI AND P. RAGHAVAN. *Randomized Algorithms*. Cambridge University Press, 1995.
7. C. H. PAPADIMITRIOU AND M. YANNAKAKIS. *Optimization, Approximation, and Complexity Classes*, JCSS **43**, 425-440 (1991).
8. S. POLJAK AND D. TURZÍK. *A Polynomial Algorithm for Constructing a Large Bipartite Subgraph with an Application to a Satisfiability Problem*. Canadian Jour. Math., Vol. 34, No. 3, 1982, pp. 519-524.
9. P. ROSSMANITH AND R. NIEDERMEIER. *New Upper Bounds for Maximum Satisfiability*. Journal of Algorithms, 36: 63-88, 2000.