THE HARDNESS OF APPROXIMATION : GAP LOCATION

Erez Petrank

Abstract. We refine the complexity analysis of approximation problems by relating it to a new parameter called *gap location*. Many of the results obtained so far for approximations yield satisfactory analysis with respect to this refined parameter, but some known results (e.g., MAX-k-COLORABILITY, MAX 3-DIMENSIONAL MATCHING and MAX NOT-ALL-EQUAL 3SAT) fall short of doing so. As a second contribution, our work fills the gap in these cases by presenting new reductions.

Next, we present definitions and hardness results of new approximation versions of some NP-complete optimization problems. The problems we treat are VERTEX COVER (for which we define a different optimization problem from the one treated in Papadimitriou & Yannakakis 1991), k-EDGE COLORING, and SET SPLITTING.

Key words. Hardness of approximation; approximation algorithms; computational complexity; NP-hardness.

Subject classifications. 03D15, 68Q25, 68Q15, 68R10.

1. Introduction

We begin by presenting a partition of the class of optimization problems into two categories. We later introduce the notion of gap location which refers to the second category.

1.1. Two categories of optimization problems. Generally, an optimization problem consists of a set of instances and a function $f : \{0,1\}^* \times \{0,1\}^* \rightarrow \mathbb{R}$ that assigns, for each instance I and each candidate solution σ, a real number $f(I, \sigma)$ called the value of the solution σ. The optimization task is to find a solution σ to a problem instance I such that $f(I, \sigma)$ is the largest possible over all $\sigma \in \{0,1\}^*$. We say that an algorithm A *approximates* a maximization (resp. minimization) problem Π to within $1-\epsilon$ (resp. $1+\epsilon$) if, for every instance I of Π whose optimal solution has value $OPT(I)$, the output of A on I satisfies $(1-\epsilon)OPT(I) \leq A(I) \leq OPT(I)$ (resp. $OPT(I) \leq A(I) \leq (1+\epsilon)OPT(I)$). Most natural optimization problems are associated with a decision problem in

NP. We have a relation $R \subseteq \{0,1\}^* \times \{0,1\}^*$ which is checkable in polynomial time (i.e., given (I,σ) it is possible to check in time polynomial in $|I|$ whether $(I,\sigma) \in R$), and we call σ a *valid solution* to an input I if $(I,\sigma) \in R$. The decision problem is whether or not there exists a valid solution to the input I. Two natural categories of corresponding optimization problems follow.

1. THE LARGEST SOLUTION.
   Here, we associate valid solutions with some natural "size" which we would like to maximize/minimize. More formally, we are trying to maximize the function

$$f(I,\sigma) = \begin{cases} \text{size}(\sigma) & \text{if } (I,\sigma) \in R \\ -\infty & \text{otherwise} \end{cases}$$

   where size$(\cdot)$ is a function which depends on the problem and can be efficiently extracted from σ (usually, size(σ) is the number of elements encoded in the solution σ). We replace $-\infty$ by $+\infty$ when a minimization problem is involved. Two examples in this category are MAX CLIQUE, in which we are looking for the size of the largest clique in the input graph, and MIN COLORING, in which we would like to find the minimum number of colors required to color the input graph such that no two adjacent vertices have the same color.

2. THE QUALITY OF THE SOLUTION.
   Here, we assume that the condition $(I,\sigma) \in R$ contains a large (yet polynomially bounded) number of natural "sub-conditions" and our task is to find the maximum number of sub-conditions that can be satisfied by a single (i.e., the best) solution. In this category, we have MAX-SAT, in which we are trying to find the maximum number of clauses that can be satisfied by an assignment to the input formula, and MAX 3-COLORABILITY, in which we are trying to find the maximum number of consistent edges in a 3-coloring of the input graph. (We call an edge in a graph G *consistent* with respect to a 3-coloring of G if its two adjacent vertices are assigned different colors.) Note that, in this setting, the best solution of an instance I is not necessarily a valid solution, since there are instances that do not have a valid solution. Also, A solution σ is a valid solution of an instance I ($(I,\sigma) \in R$) iff σ satisfies all sub-conditions implied by I.

Approximation problems can be partitioned in the same manner and so can results concerning the difficulty of approximations. The first category contains problems such as approximating the size of the biggest clique in the input graph (the hardness of this problem was shown by Feige *et al.* 1991, Arora & Safra 1992, and Arora *et al.* 1992) or the minimum number of colors needed to properly color the input graph (Lund & Yannakakis 1992). The second category includes problems such as approximating the maximum number of clauses that can be simultaneously satisfied in an input formula (Arora *et al.* 1992), or approximating the maximum number of consistent edges in a best 3-coloring of the input graph (Papadimitriou & Yannakakis 1991).

The gap location parameter is a natural measure which arises when analyzing the hardness of approximation problems in the second category.

1.2. The gap-location parameter. Practically all researchers in the area noticed the connection between the hardness of approximating a problem and the existence of a "gap" that is hard to differentiate. For example, the hardness of approximating MAX-SAT was shown by proving that there exists a constant $\epsilon_0 > 0$ such that, unless P=NP, one can not distinguish in polynomial time between formulae for which all clauses can be satisfied and formulae for which only a fraction $1 - \epsilon_0$ of clauses can be satisfied (Arora *et al.* 1992). The L-reductions that were used by Papadimitriou & Yannakakis (1991) preserve the existence of such gaps and thus, polynomial time inseparable gaps appear in all MAX-SNP-hard problems. This implies that these problems have no polynomial time approximation schemes. Loosely speaking, a hard gap for an optimization problem Π consists of two reals $0 \le \alpha_0, \epsilon_0 \le 1$ such that, given an instance I of Π, it is NP-hard to tell whether I has a solution that satisfies at least a fraction α_0 of the sub-conditions posed by I or whether any solution of I satisfies at most a fraction $\alpha_0 - \epsilon_0$ of the sub-conditions posed by I. Such a hard gap is said to have location α_0. For example, the hard gap proven for MAX-SAT has location 1. Let us first argue that the "location" of the hard gap shown for MAX-SAT is the "right" one and then explain why there are some problems whose proofs of hardness are weaker in this respect.

Our main interest lies in the original question of satisfiability, i.e., in telling whether a formula φ has an assignment that satisfies all its clauses or not. It is therefore interesting to see that we cannot solve even the easier question of whether all the clauses of φ can be satisfied (simultaneously) or whether any assignment to φ satisfies at most a fraction $1 - \epsilon_0$ of its clauses. It would be somewhat artificial (and clearly of lesser interest) to show that it is impossible to tell whether a formula φ has an assignment that satisfies more than 2/3 of its clauses or whether no assignment to φ satisfies more than $2/3 - \epsilon_0$ of its clauses, although showing this would still imply the hardness of approximating

MAX-SAT (i.e., approximating the maximum number of simultaneously satisfied clauses). Furthermore, the intuition about the "right" location of the hard gap coincides with the power of such a result. Namely, for MAX-SAT, the existence of a hard gap at location 1 implies the existence of a hard gap at any location $1/2 < \alpha \leq 1$. That is, the fact that one cannot tell in polynomial time whether a formula has an assignment that satisfies all its clauses or whether all assignments to φ satisfy at most a fraction $1 - \epsilon_0$ of its clauses implies the fact that one cannot tell in polynomial time whether a formula has a satisfying assignment that satisfies more than a fraction α of its clauses or whether any assignment to φ satisfies at most a fraction $\alpha - \epsilon_\alpha$ of its clauses, where ϵ_α is a constant that depends only on ϵ_0 and α. (This can be shown by a padding argument. Namely, use enough new variables $y_1, \ldots, y_l$ and add the clauses (y_i) and $(\overline{y_i})$, $1 \leq i \leq l$, to the original formula.) There are no hard gaps at locations $0 \leq \alpha \leq 1/2$ since any formula has an assignment that satisfies at least half of its clauses. We conclude that, in this respect, the proof of the hardness of approximating MAX-SAT is the strongest possible.

We conjecture a generalization of this example. Suppose we have a "natural" optimization problem that seeks the quality of the best solution. We conjecture that showing a hard gap at location $\alpha_0 = 1$ can be used to prove the existence of hard gaps in all other locations (in which they exist).

In previous works, hard gaps are not always shown at location $\alpha_0 = 1$. For example, recall that in MAX k-COLORABILITY, we look for the maximum number of consistent edges in a k-coloring of the input graph. The interesting original problem (for $k \geq 3$) is to tell whether a graph $G(V, E)$ is k-colorable or not. Relaxing the precision requirement, we would like to know if it is easier to tell whether G has a k-coloring, for which all $|E|$ edges in G are consistent, or whether for all k-colorings of G at most $(1 - \epsilon)|E|$ edges are consistent. Can this relaxation be determined in polynomial time for any constant ϵ? This question was not considered before. Instead, following the gap-preserving L-reductions used by Papadimitriou & Yannakakis (1991), we get that there are constants $0 < \epsilon_0, \alpha_0 < 1$ such that, unless P=NP, it is not possible to tell in polynomial time whether a given graph $G(V, E)$ has a k-coloring with more than $\alpha_0|E|$ consistent edges, or whether for any k-coloring of G at most $(\alpha_0 - \epsilon_0)|E|$ edges are consistent. Using this hard gap we can indeed say that there is no polynomial time approximation scheme for MAX k-COLORABILITY unless P=NP, but the hardness of the interesting gap (i.e., $\alpha_0 = 1$) remains open.

The importance of the gap at location $\alpha_0 = 1$ can also be expressed in terms of the analogous search problem. The implication of the result of Papadimitriou & Yannakakis (1991) is that given a graph G, we cannot find a k-coloring of G

that is as close as desired to the optimal solution (unless P=NP). But, suppose we are given a k-colorable graph and we would like to color it such that as many edges as possible are consistent. Papadimitriou and Yannakakis (1991) give no evidence that we cannot achieve this task in polynomial time such that the number of consistent edges is greater than $(1-\epsilon)|E|$ for any constant ϵ.

1.3. Summary of results. The hardness of finding a k-coloring that has "almost" as many consistent edges as possible, for k-colorable graphs, is implied by our showing a hard gap at location 1 for MAX k-COLORABILITY (for all $k \geq 3$). We thus settle the problem raised in the previous subsection. In our proof, we use a different reduction than the one in Papadimitriou & Yannakakis (1991). Two other problems that were previously shown hard to approximate using a gap location different from 1 are 3-DIMENSIONAL MATCHING (Kann 1991) and NOT-ALL-EQUAL-3SAT (Papadimitriou and Yannakakis 1991). We strengthen these results by showing that these problems indeed possess a hard gap at location 1.

Last, we define new approximation versions of some known NP-complete problems, in the spirit of approximating the quality of the best solution. In particular, we define approximation versions of:

- VERTEX COVER (Karp 1972). Note that the version that was treated by Papadimitriou & Yannakakis (1991) is in the spirit of the largest solution. We treat the version that seeks the quality of the best solution.

- k-EDGE COLORABILITY (CHROMATIC INDEX) (Hoyler 1981 and Leven & Galil 1983). In this case, it is not hard to approximate the size of the smallest solution since there is a polynomial time algorithm that colors the edges of a graph of degree k with $k+1$ colors (Vizing's Theorem, see Berge 1973). Again, we treat the version that seeks the quality of the best solution (i.e., a best k edge-coloring of a k-degree graph).

- SET SPLITTING (Lovász 1973).

We show that all these problems possess a hard gap at gap-location 1.

1.4. Summary of the motivation for proving hardness at gap-location 1. To complete the discussion, we summarize the arguments that motivate showing a hard gap at location 1. We provide four motivating arguments, as follows.

1. **The relation to the decision problem.** Recall that the original decision problem was to distinguish between instances for which all subconditions can be satisfied and instances for which not all subconditions can

be satisfied. Showing a hard gap at location 1 implies the hardness of the relaxation of the original problem in which we have to distinguish between instances for which all subconditions can be satisfied and instances for which all solutions are "far from" satisfying all subconditions. This implication on the relaxation of the original problem does not follow from hard gaps in other locations. We believe that this implication is fundamental: the approximation task is initially meant to relax the hardness of the original task, therefore, we would expect that the hardness of approximation should imply the hardness of the relaxation of the original task.

2. **The relation to the search problem.** Suppose we are given instances which satisfy the original decision problem (i.e., there exists a solution that satisfies all their subconditions), and we would like to find a solution that satisfies as many subconditions as possible. For example, given a 3-colorable graph G, find a 3-coloring of G with as many as possible consistent edges. As discussed in Section 1.2, a hard gap at location 1 implies the NP-hardness of finding a solution which is even "close" to the optimal solution. A hard gap at any location other than 1 does not imply the hardness of this search task.

3. **The hardness of the complementary minimization problem.** Consider the complementary minimization problem in which we try to find the minimum (over all possible solutions) of the number of subconditions that are not satisfied. A hard gap at location 1 for our maximization problem implies that it is NP-hard to approximate this complementary minimization problem to within any ratio. This implication follows from the fact that it is NP-hard to tell between instances having a minimum of 0 subconditions not satisfied and instances having a minimum of a constant fraction of the subconditions not satisfied. This strong implication on the hardness of the complementary problem does not follow from the existence of a hard hard gap at any other location. (Note that this implication follows also from the NP-hardness of the decision problem, however, it is not implied by a hard gap at any location other than 1.)

4. **Strength of the result (given our conjecture).** We conjecture that for natural problems, showing a hard gap at location 1 implies a hard gap at all other possible locations. As will be explained in Section 2, the converse is not generally true. Namely, there exist problems having a hard gap at a location other than 1 which do not have a hard gap at

location 1 at all. Therefore, given our conjecture, showing a hard gap at location 1 is the strongest possible result in this respect.

1.5. Related results.

Quadratic equations over the rationals. A related problem possessing a hard gap at location 1 is MAX-QER: given a set of quadratic equations with rational coefficients, find the maximum number of equations satisfied, over all rational assignments to the variables. We include a simple proof of this fact which is due to Bellare & Petrank (1992), Theorem 5.7. The same problem when the field is $GF(p)$ (p a prime) was considered by Håstad, Phillips, and Safra (1993); they showed that if P $\neq$ NP, then MAX-QE(p) cannot be approximated to within a factor of $\frac{1}{p-(1/poly)}$.

k-coloring. Most problems shown as having a hard gap in this paper have related positive results asserting that it is possible to approximate them to within some constant ratio in polynomial time. A different kind of positive result was given by Alon *et al.* (1992). They showed that for any $\epsilon > 0$, there is a polynomial time algorithm that, given k-colorable graphs having $|E| = \Omega(|V|^2)$, finds a k-coloring in which the number of consistent edges exceeds $(1-\epsilon)|E|$. This does not contradict our impossibility result for the general case and it is interesting to note that the graphs output by our reduction (which shows a hard gap for MAX k-COLORABILITY) have their number of edges linear in their number of vertices. The existence of hard gaps (or approximation schemes) for graphs having $|E|$ neither quadratic nor linear in $|V|$ remains open.

2. The Gap Location Parameter

In this section, we introduce the definitions concerning hard gaps. We follow the definitions with an informal discussion. Let us make a distinction between optimization problems that seek the quality of the best solution and optimization problems that seek the size of the largest solution. We concentrate on the first class of problems, for which the gap-location parameter is more interesting. However, it should be noted that a straightforward modification of the definition below makes it suitable for all optimization problems. Denote by $N(I)$ the number of sub-conditions posed by the instance I. (To make this notation suitable for all optimization problems, one may interpret $N(I)$ as a polynomial time computable bound on the optimum value of I.) $OPT(I)$ denotes the number of sub-conditions which are satisfied by the best solution of I (i.e., the solution that maximizes the number of satisfied sub-conditions).

DEFINITION 2.1 (THE PARAMETERS OF A HARD GAP). *Consider an optimization problem in which we are looking for the quality of the best solution. Suppose that it is NP-hard to tell whether* $OPT(I) \leq \alpha_1 \cdot N(I)$ *or* $OPT(I) \geq \alpha_0 \cdot N(I)$. *Then we say that there exists a* hard gap *for the problem P, at* location α_0, *with* width $\alpha_0 - \alpha_1$.

In this work, we are only concerned with gaps of constant width so we omit the width parameter in what follows. Throughout the paper, we show that problems have hard gaps. Let us clearly state the implication of hard gaps on the hardness of approximations.

REMARK 2.2. *If a maximization (or a minimization) problem* P *has a hard gap at any location* $0 < \alpha \leq 1$, *then there exists a constant* $\epsilon > 0$ *such that, unless P=NP, there is no polynomial time algorithm that approximates* P *to within* $1 - \epsilon$ *(or* $1 + \epsilon$*), respectively.*

The instances of an optimization problem always satisfy $0 \leq OPT(I) \leq N(I)$, but sometimes the solution interval is more restricted.

DEFINITION 2.3 (THE SOLUTION INTERVAL). *Consider an optimization problem* P *in which we seek the quality of the best solution. We say that* P *has solution interval* $[\beta_1, \beta_2]$ *if* $\liminf_{|I|\to\infty} \frac{OPT(I)}{N(I)} = \beta_1$ *and* $\limsup_{|I|\to\infty} \frac{OPT(I)}{N(I)} = \beta_2$.

Namely, neglecting a finite number of instances, $OPT(I)$ satisfies $\beta_1 \cdot N(I) \leq OPT(I) \leq \beta_2 \cdot N(I)$. Usually, $\beta_2 = 1$ in natural problems.

Informal discussion. Let us restrict the discussion to natural problems that seek the quality of the best solution and that are hard to approximate (i.e., have a hard gap somewhere in their solution interval). Also, let the solution interval of P be $[\beta_P, 1]$. We would like to state two beliefs we have concerning the issue of gap-location.

The first belief is that if there exists a hard gap at any location in the interval solution, then there exists hard gaps at all points in the open interval $(\beta_P, 1)$. Note that this implies a difference between proving a hard gap at location 1 and at any other location. A hard gap at location 1 implies (assuming this conjecture) a hard gap at all locations in the interval $(\beta_P, 1]$, whereas a hard gap at any other location does not imply a hard gap at location 1 (even assuming the conjecture). In fact, there exist optimization problems for which there are hard gaps at all locations in the open interval $(\beta_P, 1)$ but not at location 1. We discuss such problems in what follows, but first, let us state our conjecture.

Informal Conjecture 2.4. *Let P be some* natural *optimization problem in which we are trying to determine the quality of the best solution. Suppose P has a solution interval $[\beta, 1]$, for some $0 \leq \beta < 1$. Then a hard gap at any location in $(\beta, 1]$ implies hard gaps at all locations in the open interval $(\beta, 1)$.*

In what follows, we assume the validity of Conjecture 2.4. Conjecture 2.5 (below) makes a distinction between problems which possess hard gaps at all locations in the interval $(\beta_P, 1]$ and problems that have hard gaps only in the open interval $(\beta_P, 1)$. Consider the following decision problem related to an approximation problem P. Given an instance I, determine whether $OPT(I) = N(I)$. For example, the decision problem related to MAX-SAT is SAT (i.e., given a formula φ, determine whether all its clauses can be satisfied). We partition the optimization problems into two classes according to the difficulty of their related decision problems. The first class contains optimization problems for which it is NP-hard to determine whether an input I has $OPT(I) = N(I)$. This class contains problems as MAX-SAT (proven hard to approximate by Arora *et al.* 1992) and MAX k-COLORABILITY (for $k \geq 3$) (Papadimitriou & Yannakakis 1991 and Section 3). The other class contains problems for which it is easy to decide whether an input I has $OPT(I) = N(I)$. This class contains problems such as MAX-CUT (which can be also viewed as MAX 2-COLORABILITY), and MAX-2SAT (both shown hard to approximate by Papadimitriou & Yannakakis 1991). (While doing this partition, we do not claim that all decision problems are either NP-hard or easy, but practically all known interesting problems of this form do fall into one of these two categories, and we are interested only in these natural problems here.) Our second belief is that any natural problem of the first class has hard gaps at all locations $\alpha \in (\beta_P, 1]$ and that any natural problem of the second class has hard gaps at all locations $\alpha \in (\beta_P, 1)$. Note that we can never have a hard gap at location β_P, since we know that all instances have $OPT(I) \geq \beta_P \cdot N(I)$ and none has $OPT(I) \leq (1 - \epsilon)\beta_P \cdot N(I)$, for any $\epsilon > 0$. Conjecture 2.5 states that a problem P has a hard gap at location 1 if and only if the corresponding decision problem is NP-hard. The validity of the two conjectures implies our second belief.

Informal Conjecture 2.5. *Let P be some* natural *optimization problem in which we seek the quality of the best solution. Suppose that P has a hard gap at some location in its solution interval. Then P has a hard gap at gap-location 1 if and only if it is NP-hard to determine whether $OPT(I) = N(I)$ (i.e., the original decision problem is NP-hard).*

Conjecture 2.4 can be proven for many known optimization problems using a padding argument. In these cases, one may use a padding argument to "transfer" the hard gap to any location in the open interval $(\beta, 1)$. This can be demonstrated on the problem MAX-SAT (as was done in the introduction) and on MAX k-COLORABILITY in the following way. To lower the gap, add a large enough clique to the graph and do not connect it to any original vertex. To move the gap up, add a large enough bipartite graph disconnected from the original nodes of the graph. Note that the solution interval here is $[1-\frac{1}{k}, 1]$ since any graph has a k-coloring for which $\left(1-\frac{1}{k}\right) \cdot |E|$ edges are consistent. Generally, note that this padding method cannot be used to transfer a hard gap from a location other than 1 to a hard gap at location 1.

Let us define a class of optimization languages for which the first conjecture can be proven. This class contains many natural optimization problems such as MAX-SAT, MAX 3-DIMENSIONAL MATCHING, etc. We define the class to contain problems which possess a "padding property", which will enable us to prove the conjecture. We will have two requirements to make of problems in this class. First, we require the existence of an efficiently computable operation which joins two instances into one. This operation, which we denote by $\circ$, will have a specific way of combining the optimum values of both instances. (We state this formally in Definition 2.6 below). The second requirement we make is that the problem has a "substantial" number of instances whose optimum lies at the endpoints of the solution interval. Formally, we have the following.

DEFINITION 2.6. *The class* PADDABLE *of optimization problems contains all optimization problems which satisfy the following two requirements:*

1. **Padding property:** *There exists a polynomial time computable padding operation, denoted by $\circ$, which operates on any two instances I_1 and I_2 of P and has the property that for all I_1, I_2, $N(I_1 \circ I_2) = N(I_1) + N(I_2)$ and $OPT(I_1 \circ I_2) = OPT(I_1) + OPT(I_2)$.*

2. **Endpoints property:** *Let $(\beta, 1]$ be the interval solution of P. There exists a constant $c > 0$ such that for any $n \in \mathsf{N}$ it is possible to find in polynomial time two instances I_1, I_2 such that $n \leq N(I_1), N(I_2) \leq n + c$, and $\frac{OPT(I_1)}{N(I_1)} = \beta$, $\frac{OPT(I_2)}{N(I_2)} = 1$.*

For this class the conjecture can be stated as a theorem.

THEOREM 2.7. *Let $P \in$ PADDABLE be some optimization problem in which we are try2ing to determine the quality of the best solution. Suppose P has a solution interval $[\beta, 1]$ for some $0 \leq \beta < 1$. Then, a hard gap at any location in $(\beta, 1]$ implies hard gaps at all locations in the open interval $(\beta, 1)$.*

Sketch of proof. Suppose that P has a hard gap at location $\alpha_0 \in (\beta, 1]$ and consider any location $\alpha_1 \in (\beta, 1)$. Suppose w.l.o.g. that $\alpha_1 < \alpha_0$. We show that if, for any $\epsilon' > 0$, there exists a polynomial time algorithm that distinguishes between the case $OPT(I)/N(I) \leq \alpha_1 - \epsilon'$ and the case $OPT(I)/N(I) \geq \alpha_1$, then there exists a polynomial time algorithm that distinguishes (for any instance I) between the case $OPT(I)/N(I) \leq \alpha_0 - \epsilon$ and the case $OPT(I)/N(I) \geq \alpha_0$. Thus, the NP-hardness of the gap at location α_1 is implied by the NP-hardness of the gap at location α_0.

Suppose we have an instance I for which we would like to determine whether $OPT(I)/N(I) \leq \alpha_0 - \epsilon$ or $OPT(I)/N(I) \geq \alpha_0$. We "pad" this instance with an instance I' such that the following two relations hold:

$$\frac{OPT(I)}{N(I)} \geq \alpha_0 \implies \frac{OPT(I \circ I')}{N(I \circ I')} \geq \alpha_1$$

$$\frac{OPT(I)}{N(I)} \leq \alpha_0 - \epsilon \implies \frac{OPT(I \circ I')}{N(I \circ I')} \leq \alpha_1 - \epsilon'$$

for the constant $\epsilon' = \epsilon/2$. The possibility to find such an instance I' in polynomial time follows from the second requirement in the definition of PADDABLE. Now, if for any constant $\epsilon' > 0$ there is a polynomial time algorithm which distinguishes between the cases $OPT(I \circ I')/N(I \circ I') \geq \alpha_1$ and $OPT(I \circ I')/N(I \circ I') \leq \alpha_1 - \epsilon'$, then we get a polynomial time algorithm that distinguishes between the cases $OPT(I)/N(I) \geq \alpha_0$ and $OPT(I)/N(I) \leq \alpha_0 - \epsilon$, and we are done. □

The definition of the class PADDABLE is semantic. We consider it an interesting open problem to find a syntactic definition of a class of optimization problems for which the first conjecture can be proven.

3. A hard gap for k-COLORABILITY at gap-location 1

Consider the problem of finding a k-coloring of a given graph G such that as many edges as possible are adjacent to two vertices of different colors.

Definition 3.1 (A consistent edge). *Consider a graph $G(V, E)$ and a coloring of its vertices $\sigma : V \to \{1, 2, ..., k\}$. We say that an edge $e = (v_i, v_j)$ is* consistent *regarding σ if $\sigma(v_i) \neq \sigma(v_j)$.*

Definition 3.2 (The problem max k-colorability).
Input: *A graph $G(V, E)$.*

Problem: *Find the maximum number of consistent edges in G, over all k-colorings of the vertices in G.*

For $k \geq 3$, it is NP-hard to tell whether a graph is k-colorable or not. We show that for any $k \geq 3$, there exists a constant $\epsilon_k > 0$ such that unless P=NP, there is no polynomial time algorithm which can determine whether an input graph $G(V,E)$ is k-colorable, or whether any k-coloring of G has at most $(1-\epsilon_k)|E|$ consistent edges.

THEOREM 3.3. *For any $k \geq 3$,* MAX k-COLORABILITY *possesses a hard gap at location 1.*

PROOF. We use a reduction from MAX 3SAT-B to MAX 3-COLORABILITY. Next, we can use techniques from Papadimitriou & Yannakakis (1991) to further reduce MAX 3-COLORABILITY to MAX k-COLORABILITY for any $k > 3$. In the reduction, we use a bipartite expander, which helps us to preserve the hard gap. The use of expanders in preserving gaps was first noticed by Papadimitriou & Yannakakis (1991). A bipartite graph on $2 \times n$ nodes is called a bipartite *expander* with degree d and expansion factor $1+\gamma$ if every subset S of at most $n/2$ nodes of one side of the graph is adjacent to at least $(1+\gamma)|S|$ nodes on the other side. Bipartite expanders on $2 \times n$ nodes can be efficiently constructed for any $n \in \mathbb{N}$ (Margulis 1975, Gabber & Galil 1981, and Ajtai 1987). Let us first show the reduction, and then show that if the instance φ of MAX 3SAT-B is satisfiable, then the output of the reduction, $G^\varphi(V,E)$, is 3-colorable (Lemma 3.4 below), while if any assignment to φ satisfies at most a fraction $1-\epsilon$ of the clauses in φ, then any 3-coloring of $G(V,E)$ induces at least $\epsilon\frac{\gamma}{2B}m \geq c|E^\varphi|$ inconsistent edges for some constant $c > 0$ (Lemma 3.5 below).

The reduction. We are given an instance of MAX 3SAT-B, i.e., a 3-CNF formula φ with n variables and m clauses, such that each variable appears at most B times in the formula φ. We use an extension of the standard reduction from 3SAT to 3-COLORABILITY (Stockmeyer 1973 and Garey *et al.* 1976). Let us shortly describe the original reduction, which uses a gadget with nine vertices and 10 edges (Figure 3.1).

We call the top vertex g_4 the gadget head, and the three bottom vertices g_1, g_2, g_3 the gadget legs. The other vertices in the gadget are called the gadget body. The useful property of this gadget (which will be denoted "$\mathcal{P}$") is that if the three legs have the same color, then any consistent 3-coloring of the gadget assigns the gadget head the same color too, while if the three legs do not have the same color, then for any color assigned to the head, we can complete the coloring of the body consistently.

Figure 3.1: The gadget.

The reduction outputs $2n$ vertices (the literals vertices) labeled $x_1, \overline{x}_1, x_2, \overline{x}_2, \ldots, x_n, \overline{x}_n$, two vertices named GROUND and B, and m gadgets, one for each clause. The edges of the output graph connect x_i to $\overline{x}_i$, x_i to GROUND, $\overline{x}_i$ to GROUND for $1 \leq i \leq n$, all the gadget heads to B, and B to GROUND. Last, it identifies the three legs of the gadget of C_i with the vertices that correspond to the literals of C_i.

Our extension proceeds as follows. We duplicate the vertex B m times to get $B_1, B_2, \ldots, B_m$, which are all connected to GROUND, and we connect them to the m gadget heads using a bipartite expander. Namely, one side of the expander (which we call *the upper side*) contains $B_1, B_2, \ldots, B_m$, and the other (*the lower side*) contains the gadget heads. The resulting graph is illustrated in Figure 3.2.

Note that, for simplicity, we have drawn the vertex GROUND twice in the figure. Note also that the number of edges in the output graph is $O(m)$.

LEMMA 3.4. *If φ is satisfiable, then G^φ is 3-colorable.*

PROOF. We use the colors T,F,G. Color the vertex GROUND with the color G. Color each literal-vertex with T if the corresponding literal is assigned TRUE by the satisfying assignment τ of φ, or with F otherwise. Color $B_1, B_2, \ldots, B_m$ with F and the gadget heads with T. It remains to color the body vertices of the clauses-gadgets. Recall that each gadget head is colored T and that since the truth assignment τ satisfies φ, then at least one gadget leg must be colored T (This is the literal-vertex that corresponds to the literal that is assigned TRUE by τ). By property $\mathcal{P}$ of the gadget, we get that it is possible to complete the coloring of all the gadget-bodies consistently. $\square$

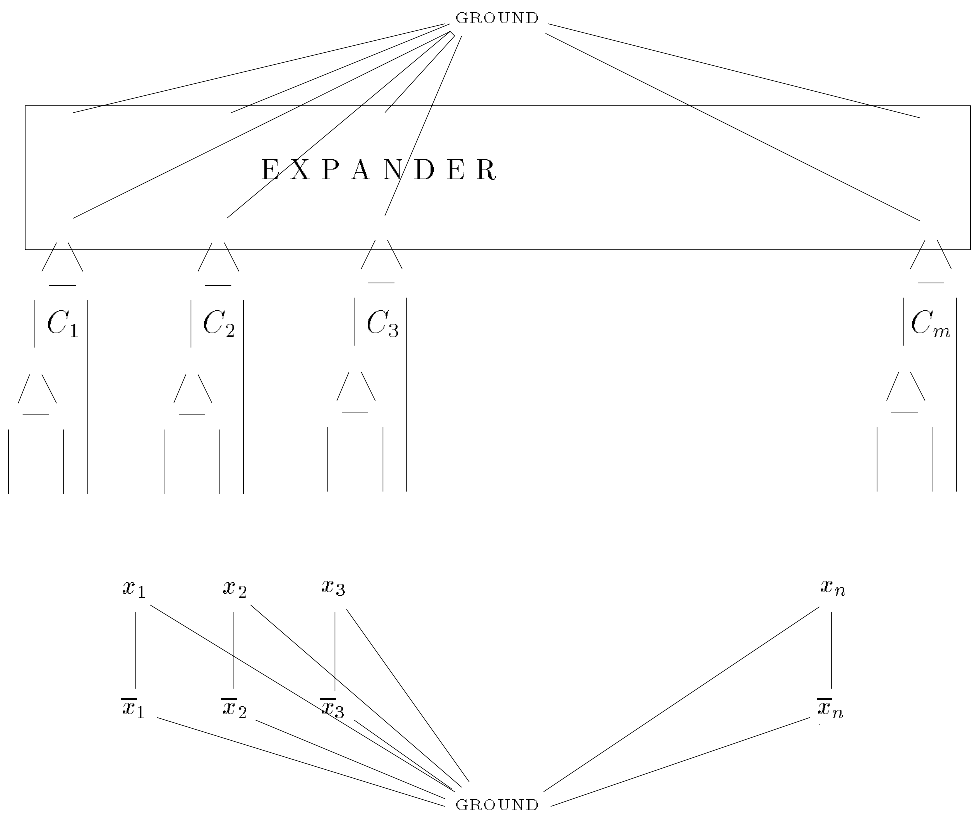

Figure 3.2: The reduction from 3-SAT to MAX-3-COLORABILITY

LEMMA 3.5. *If there exists a 3-coloring of G^{φ} which induces δm inconsistent edges, then there is an assignment to φ that satisfies at least $\left(1 - \frac{2B\delta}{\gamma}\right) m$ clauses in φ, where $1 + \gamma$ is the expansion rate of the polynomial time constructible expander that is used in the construction of G^{φ}.*

PROOF. The lemma is trivially valid for $\delta \geq 1/4$, since for any formula there exists an assignment that satisfies at least half of its clauses. Therefore, we restrict ourselves to $\delta < 1/4$. Given a 3-coloring of G^{φ}, we first select names for the 3 colors and define an assignment to the variables of φ. Denote the color of the vertex GROUND by G. The majority color in the heads of all gadgets is denoted T (if there are two candidate colors appearing the same number of times, select one of them arbitrarily). Note that the color T is different from G since all the m gadget-heads are connected to the GROUND vertex and we have less than $m/4$ inconsistent edges. The third color is denoted F. Now, to fix the assignment for the variable x_i, $1 \leq i \leq n$, we consider the vertex labeled x_i. If it is colored T, we assign the value TRUE to x_i. Otherwise, the assignment for x_i is FALSE. We claim that this assignment satisfies at least $\left(1 - \frac{2B\delta}{\gamma}\right) m$ clauses.

By the property of the gadget, there is no consistent 3-coloring of the vertices of the gadget such that the head of the gadget is colored T and its three bottom vertices (the three literal vertices) are colored F. Therefore, if the gadget of the clause C_j, $1 \leq j \leq m$, has all its edges consistent and its head colored T, then one of its literals must be colored T or G. In other words, if a clause $C_j = (l_{j_1} \vee l_{j_2} \vee l_{j_3})$, $1 \leq j \leq m$, is not satisfied by our assignment, i.e., all its literals are assigned FALSE by our assignment, then one of the following conditions must be met:

1. One of the edges in the gadget of C_j is inconsistent.
2. The gadget head is not colored T.
3. One of the literals in the clause is assigned FALSE by our assignment, and its vertex is not colored F.

We show that these three conditions cannot be met "too many times" in the graph G^φ by showing that fulfillment of these conditions implies inconsistent edges. Denote by K_1, K_2 and K_3 the number of times that conditions 1, 2, and 3 are met in the graph G^φ. Clearly, the number of clauses that are not satisfied by our assignment is at most $K_1 + K_2 + K_3$. We claim that

$$K_1 + \frac{\gamma}{2} K_2 + \frac{1}{B} K_3 \;<\; \delta m. \tag{3.1}$$

Assuming this (the proof follows) we get that

$$K_1 + K_2 + K_3 \;\leq\; \frac{2B}{\gamma} \delta m$$

and therefore, the number of clauses that are not satisfied by our assignment is at most $\frac{2B\delta}{\gamma} m$, as needed.

It remains to prove Equation (3.1), i.e., to show that there are at least $K_1 + \frac{\gamma}{2} K_2 + \frac{1}{B} K_3$ inconsistent edges in G^φ. We partition the edges in the graph into three disjoint subsets, and give a lower bound on the number of inconsistent edges in each subset (given K_1, K_2, and K_3). Since the subsets are disjoint, we can sum these lower bounds into a single lower bound on the number of inconsistent edges in the graph. The first subset consists of the edges inside the gadgets. By the definition of K_1, we have at least K_1 inconsistent edges in this subset. Next, we consider all the edges that connect the literal-vertices to each other and to the vertex GROUND. Recall that each literal appears in at most B clauses and therefore, if we have K_3 clauses connected to literals having the property of condition 3, then there are at least $\frac{1}{B}K_3$ literals that are assigned FALSE by our assignment and whose vertices are not colored F. We would like to show that for each such literal, there is a unique inconsistent edge. If the vertex l_i or the vertex $\overline{l_i}$ is colored G, then there exists an inconsistent edge between that vertex and the vertex GROUND and we are done for that literal. So, assume this is not the case. Since the vertex l_i is not colored F and not colored G, then it is colored T, and since we assigned FALSE to the literal l_i, the vertex $\overline{l_i}$ must be colored T also and we get an inconsistent edge $(l_i, \overline{l_i})$. Note that for each vertex that satisfies the above condition we have a different inconsistent edge. Therefore, we have at least $\frac{1}{B}K_3$ inconsistent edges in the second subset.

The remaining edges consist of the edges of the expander and the edges that connect the expander to the vertex GROUND. We claim that if K_2 vertices of the lower side of the expander are not colored T, then at least $\frac{\gamma}{2}K_2$ edges in this set are inconsistent. Denote by l_1, l_2, and l_3 the number of vertices in the lower side of the expander that are colored T, F, and G correspondingly. Recall that $K_2 = l_2 + l_3$ is the number of expander lower vertices (which are gadget-head vertices) that are not colored T. We show that the number of inconsistent edges in this last set of edges is at least

$$\max(l_3, \gamma l_2) \;\geq\; \frac{\gamma}{2}K_3.$$

Clearly, there are at least l_3 inconsistent edges in this set because there are l_3 different edges that connect the vertex GROUND, which is colored G, to gadget-head vertices with the same color. On the other hand, consider the l_2 expander lower vertices that are colored F. By definition of the color names, l_2 is smaller than half the number of the vertices in the lower side of the expander. Using the expansion property of the expander, these nodes have a set of $(1+\gamma)l_2$ neighbors in the upper side of the expander. Denote this set of neighbors by S. We cannot use the expansion property again on the set S since we are not sure that it is small enough. However, we know that S is adjacent to at least $|S|$ vertices on the lower side. Furthermore, we can associate with each vertex in S a unique neighbor on the lower side. (This holds for all known expander constructions. Yet, this property can be simply achieved (without foiling the expander) by adding m edges that connect each vertex i on one side to vertex i on the other side.) Now, only l_2 of the vertices in S can have their associated neighbors in the lower side colored F. The other γl_2 members of S have their γl_2 twin vertices on the lower side colored G or T. Putting it all together, we have at least γl_2 vertices in S which, on one hand, are connected to a vertex colored F (by the definition of the set S), and on the other hand, are connected to unique lower vertices that are not colored F. Consider such a vertex in S. If it is colored F, then we have a unique inconsistent edge between this vertex and an F-colored lower vertex. If it is colored G, then we get a unique inconsistent edge between this vertex and the vertex GROUND. The last possibility is that this vertex is colored T. If its twin vertex is also colored T, then we have an inconsistent edge between them. Otherwise, the unique neighbor is colored G (since we know that it is not F), and we also end up with a unique inconsistent edge between the lower neighbor and the vertex GROUND. This completes the proof of Lemma 3.5, and of Theorem 3.3. □

4. The hardness of MAX NAE 3SAT and MAX 3DM

Two more problems were shown to have a hard gap at locations less than 1 although it is NP-hard to decide if an instance has its best solution at location 1. These are MAX NOT-ALL-EQUAL 3SAT (Papadimitriou & Yannakakis 1991) and MAX 3-DIMENSIONAL MATCHING (Kann 1991). In this section, we show that these problems have a hard gap at the gap location 1. Let us begin with defining the problems.

DEFINITION 4.1 (THE PROBLEM MAX NOT-ALL-EQUAL 3SAT).
Input: *A 3-CNF formula* φ.
Problem: *Find the maximum number of clauses that contain at least one true literal and at least one false literal, over all truth assignments to the variables of* φ.

DEFINITION 4.2 (THE PROBLEM MAX 3-DIMENSIONAL MATCHING-B).
Input: *A set* $M \subseteq W \times Y \times Z$*, where* W, Y*, and* Z *are disjoint finite sets and each element in* $W \cup Y \cup Z$ *appears in the triplets of* M *at most* B *times.*
Problem: *Find the maximum number of elements in* $W \cup Y \cup Z$ *which appear exactly once in the triplets of* M'*, over all* $M' \subseteq M$.

THEOREM 4.3. *The problem* MAX NOT-ALL-EQUAL 3SAT *possesses a hard gap at location 1.*

PROOF. We first reduce 3SAT to NOT-ALL-EQUAL 4SAT, and then reduce NOT-ALL-EQUAL 4SAT to NOT-ALL-EQUAL 3SAT. In the first reduction, we simply add a new variable to all clauses. Note that for the NOT-ALL-EQUAL problem, the number of "satisfied" clauses does not change if we use an assignment σ or its complement $\overline{\sigma}$ (the complement of an assignment is an assignment that gives the opposite value to each of the variables in the formula). Therefore, we can fix the new variable to always be assigned FALSE without changing the solution to this optimization problem, and the maximum number of clauses that can be satisfied in the input formula (to the reduction) is exactly the number of clauses that can have both a false and a TRUE literal simultaneously in the output formula. In the second reduction, we treat each clause $C_j = (l_{j_1} \vee l_{j_2} \vee l_{j_3} \vee l_{j_4})$ that contains four literals by adding a new variable y_j and replacing C_j with the two clauses: $(l_{j_1} \vee l_{j_2} \vee y_j) \wedge (\overline{y_j} \vee l_{j_3} \vee l_{j_4})$. It is easy to verify that there exists an assignment to the variables of the original formula, in which at least one of the literals $l_{j_1}, l_{j_2}, l_{j_3}, l_{j_4}$ is TRUE and at least one is false iff there exists an assignment to the variables of the new formula such that both sets of literals $\{l_{j_1}, l_{j_2}, y_j\}$ and $\{\overline{y_j}, l_{j_3}, l_{j_4}\}$ contain at least one TRUE literal and at least one false literal. Hence, the hard gap at location 1 is preserved. Note that the width of the new hard gap is at least half the width of the original gap. This is because the number of clauses is at most twice the number of clauses of the original formula, and the number of clauses that can not be satisfied does not decrease. □

THEOREM 4.4. *For any $B \geq 3$,* MAX 3-DIMENSIONAL MATCHING-B *possesses a hard gap at location 1.*

PROOF. A slight modification of the original reduction from SAT to 3-DIMENSIONAL MATCHING (Karp 1972) works as a gap preserving reduction from MAX-SAT-B to MAX 3-DIMENSIONAL MATCHING-3. We shortly describe the reduction, following the presentation of Garey & Johnson (1979). Given a formula φ with n variables and m clauses in which each variable appears at most B times, we construct three disjoint sets $W^\varphi, Y^\varphi, Z^\varphi$ and a set of triplets $M^\varphi \subseteq W^\varphi \times Y^\varphi \times Z^\varphi$. The triplets in M^φ consist of n truth-setting components (one for each variable), m satisfaction-testing components (one for each clause), and a "garbage collection" mechanism.

Let x_i be a variable that appears d_i times in the formula. The truth-setting component of x_i involves "internal" elements $a_i[k] \in W^\varphi$, $b_i[k] \in Y^\varphi$ and "external" elements $x_i[k], \overline{x}_i[k] \in Z^\varphi$, for $1 \leq k \leq d_i$. We call the $a_i[k]$'s and the $b_i[k]$'s internal because they appear only inside their truth-setting component.

The external $x_i[k]$'s and $\overline{x}_i[k]$'s appear in their truth-setting components as well as in other components (which we describe later). The triplets making up the truth-setting component can be divided into two sets:

$$T_i^t = \{(\overline{x}_i[k], a_i[k], b_i[k]) \,:\, 1 \le k \le d_i\}\,,$$

$$T_i^f = \{(x_i[k], a_i[k+1], b_i[k]) \,:\, 1 \le k < d_i\} \cup \{(x_i[d_i], a_i[1], b_i[d_i])\}\,.$$

The property of this component is that any matching that covers all internal elements $a_i[k], b_i[k]$, $1 \le k \le d_i$, exactly once contains either exactly all triplets in T_i^t or exactly all triplets in T_i^f. Thus, we get that either all elements $x_i[k]$ are covered and all elements $\overline{x}_i[k]$ are not (we associate this with assigning FALSE to x_i), or all elements $\overline{x}_i[k]$ are covered and all elements $x_i[k]$ are not (this is associated with assigning TRUE to x_i). Note that the number of elements produced so far is $O(n)$ since each variable appears at most B times in φ. (We remark that the original reduction produced $O(nm)$ elements, having m external elements for each variable).

The satisfaction-testing component that stands for the clause C_j ($1 \le j \le m$) contains two internal elements $s_1[j] \in W^\varphi$ and $s_2[j] \in Y^\varphi$, and at most three external elements from the truth-setting components that correspond to the literals in C_j. If C_j contains the k-th appearance of the variable x_i then we add the triplet $(x_i[k], s_1[j], s_2[j])$ if x_i appears positively in C_j or the triplet $(\overline{x}_i[k], s_1[j], s_2[j])$ if x_i appears negated in C_j. These components add $2m$ elements to the output. So far, each internal element appears in at most three triplets of M^φ and each external element appears at most twice.

Note that a matching that covers all internal elements exactly once and which does not use an external element more than once corresponds to a truth assignment that satisfies φ. Given an assignment τ that satisfies φ, we select the triplets of T_i^t to be in the matching if x_i is assigned TRUE by τ or the triplets of T_i^f otherwise. This leaves all elements $x_i[k]$ ($1 \le k \le d_i$) uncovered if x_i is assigned TRUE by τ or all elements $\overline{x}_i[k]$ uncovered otherwise. For each clause C_j, we select a literal that is assigned TRUE (such a literal must exist since τ satisfies φ). The element that corresponds to the appearance of this literal in C_j is not covered, since its literal is assigned TRUE. Thus, we may choose the triplet that contains this element to cover $s_1[j]$ and $s_2[j]$.

In order to cover the remaining uncovered external elements, we use a garbage collection mechanism. The original mechanism is too big (it contains $O(nm)$ elements) and does not meet the demand that each element appears in at most three triplets. We present an appropriate garbage collecting mechanism later.

Consider the other direction, in which we are given a good matching and we would like to build a satisfying assignment to φ. In order to show a hard gap in MAX 3-DIMENSIONAL MATCHING-B, we shall show that if there are "only few" violations in the given matching, then there is a truth assignment that satisfies "almost all" clauses in φ. More formally, if there exists a matching M' for which the number of internal elements which appear more than once or none at all plus the number of external elements that appear more then once is at most δm, then there exists an assignment that satisfies more than $(1-\delta B)m$ clauses.

To fix an assignment for x_i, consider the truth-setting component of x_i. We assign TRUE to x_i if the M' contains a triplet in T_i^t. Otherwise, x_i is assigned FALSE. Note that if this truth assignment does not satisfy a clause C_j, then one of the following conditions must be met.

1. The internal elements $s_1[j]$ and $s_2[j]$ do not appear in the matching M'.

2. The elements $s_1[j]$ and $s_2[j]$ appear in a triplet that contains a literal which was assigned FALSE by our assignment.

Suppose condition (1) is met K_1 times and condition (2) is met K_2 times in M'. The number of clauses in φ that are not satisfied by our assignment is at most $K_1 + K_2$. To conclude, we show that the number of violations in the matching M' is at least $2K_1 + \frac{1}{B}K_2$. Clearly, each time condition (1) is met, we have two unique internal elements that are not covered by M'. If condition (2) is met K_2 times in M', then there are at least $\frac{1}{B}K_2$ literals that are assigned FALSE and that have an associated external element covered by a satisfaction-testing component. We claim that the truth-setting component of this literal has either an internal element that does not appear uniquely in the matching M', or an external element that appears more than once. Suppose all internal elements appear exactly once. By the property of the truth-setting component, we know that all external elements associated with the literal that was assigned FALSE by our assignment are covered by the truth-setting component. Thus, the external element appears in M' at least twice.

It remains to show how to do garbage collection with $O(m)$ elements such that no element appears in more than three triplets. Note that the existence of a hard gap is already proven, but we must show a garbage collection mechanism in order to place the hard gap at location 1. An elegant way to solve both problems of the original garbage collection mechanism is due to Garey and Johnson (private communications). Create three independent copies of the construction described, with the roles of W^φ, Y^φ, and Z^φ cyclically permuted between the three copies. Now, for each external element $x_i[k]$ and $\overline{x}_i[k]$ (currently included in just two triplets), add a single triplet including the three copies (one from each copy of the overall construction). This method shrinks the width of the gap by a constant factor (at location 1), and therefore is sufficient to prove the theorem. □

5. Some more hard problems

In the introduction, we discussed the difference between optimization problems in which we seek the value of the largest solution and optimization problems in which we seek the quality of the best solution. In this section, we consider three optimization problems that seek the quality of the best solution, and which were considered before only in the largest solution version. We believe that these new optimization problems are interesting and so we give their definitions and investigate their hardness properties. Specifically, we treat approximation versions of VERTEX COVER, SET SPLITTING, and EDGE COLORING (CHROMATIC INDEX).

Let us start with the definitions. We say that an edge (v_i, v_j) in a graph $G(V, E)$ is *covered* by a subset $V' \subseteq V$ if $v_i \in V'$ or $v_j \in V'$.

DEFINITION 5.1 (THE PROBLEM MAX VERTEX COVER-B).
Input: *A graph $G(V, E)$ with degree at most B and an integer K.*
Problem: *Find the maximum number of edges in G that V' covers, over all subsets $V' \subseteq V$ of cardinality K.*

Note the difference between this problem and the optimization problem MIN VERTEX COVER treated by Papadimitriou & Yannakakis (1991) which was to minimize the size of the vertex cover (which covers all edges). In MAX VERTEX COVER, we are given the size of the cover, K, in the input, and we are trying to maximize the number of edges that are covered.

DEFINITION 5.2 (THE PROBLEM MAX SET SPLITTING).
Input: *A collection C of subsets of a finite set S.*
Problem: *Find the maximum number of subsets in C that are not entirely contained in either S_1 or S_2, over all partitions of S into two subsets S_1 and S_2.*

We say that a vertex $v \in V$ is consistent regarding an edge-coloring of a graph $G(V, E)$ if no two edges of the same color are adjacent to v.

DEFINITION 5.3 (THE PROBLEM MAX k-EDGE COLORABILITY).
Input: *A graph $G(V, E)$ of degree k.*

Problem: *Find the maximum number of consistent vertices, over all edge-colorings of G with k colors.*

THEOREM 5.4. *The following problems possess a hard gap at location 1:*

1. MAX VERTEX COVER-B
2. MAX k-EDGE COLORABILITY (CHROMATIC INDEX)
3. MAX SET SPLITTING

PROOF:

1. We use the identity reduction from MIN VERTEX COVER-B (a problem that was shown hard by Papadimitriou & Yannakakis 1991). If there is a vertex cover of G of cardinality K that covers at least $(1-\epsilon)|E|$ edges in G, then there is a cover of all edges with at most $K+\epsilon|E|$ vertices. Note that $|E| \leq |V| \cdot B/2$, and recall that the hard gap for MIN VERTEX COVER-B was shown hard for $K = \Theta(|V|)$.

2. Use the original reduction of MAX-SAT to k-EDGE COLORABILITY (Hoyler 1981 and Leven & Galil 1983), only let the domain of the reduction be MAX 3SAT-B (shown hard by Papadimitriou & Yannakakis 1991).Clearly, if there is a satisfying assignment to the input formula, then there is an edge-coloring of the output graph with k colors such that all the vertices are consistent. It is left to show that if there is a k-coloring of the edges with a small number of inconsistent vertices, then there is an assignment that satisfies almost all clauses in the input formula. This can be done using an accountancy similar to the one of Lemma 3.5.

3. The trivial reduction from MAX NOT-ALL-EQUAL 3SAT works. □

REMARK 5.5. *The approximation of* MAX SET SPLITTING *remains hard even if all subsets in C are of cardinality less than or equal to* 3 *(see the proof).*

We follow by defining the problem MAX-QER.

DEFINITION 5.6 (THE PROBLEM MAX-QER: MAX QUADRATIC EQUATIONS OVER THE RATIONALS).
Input: *A set of quadratic equations over the rational field in the variables $x_1, \ldots, x_n$.*
Problem: *Find the maximum number of equations that are satisfied, over all*

assignments of rational numbers to $x_1, \ldots, x_n$.

THEOREM 5.7 (BELLARE & PETRANK 1992). MAX-QER *has a hard gap at location 1.*

PROOF. Reduce MAX-3SAT-B to MAX-QER in the following way. For each variable x, add the equation $x^2 = x$. For each clause $C_i = (x \vee y \vee z)$ add a new variable c_i and produce two equations. First, write $c_i = xy$, and then use it to reduce the degree of the equation $(1-x)(1-y)(1-z) = 0$ to 2, by substituting each appearance of xy with c_i. □

Note that the same proof is valid over the field of real numbers as well.

Acknowledgements

I am greatly indebted to Oded Goldreich for many insightful discussions; without him this work would have not been possible. I am grateful to David S. Johnson for his quick and profound answer to a query I made. I am also grateful to Amos Beimel, Benny Chor, and Seffi Naor for helpful discussions. And last, I am very grateful to the anonymous referees for their deep and helpful remarks.

A preliminary version of this work has appeared in Petrank (1993).

This research was supported by grant no. 89-00312 from the United States—Israel Binational Science Foundation, Jerusalem, Israel.

References

M. AJTAI, Recursive Construction for 3-Regular Expanders. In *Proc. 28th Ann. Symp. Found. Comput. Sci.*, 1987, 295–304.

N. ALON, R.A. DUKE, H. LEFMANN, V. RÖDL, AND R. YUSTER, The Algorithmic Aspects of the Regularity Lemma. In *Proc. 33th Ann. Symp. Found. Comput. Sci.*, 1992, 473–482.

S. ARORA AND S. SAFRA, Probabilistic Checking of Proofs: A New Characterization of NP. In *Proc. 33th Ann. Symp. Found. Comput. Sci.*, 1992, 1–13.

S. ARORA, C. LUND, R. MOTWANI, M. SUDAN, AND M. SZEGEDY, Proof Verification and Intractability of Approximation Problems. In *Proc. 33th Ann. Symp. Found. Comput. Sci.*, 1992, 14–23.

M. BELLARE AND E. PETRANK, private communication, 1992.

C. Berge, *Graph and Hypergraphs.* North-Holland, Amsterdam, 1973.

M. Bern and P. Plassmann, The Steiner problem with edge lengths 1 and 2. *Inform. Process. Lett.* **32** (1989), 171–176.

A. Blum, T. Jiang, M. Li, J. Tromp, and M. Yannakakis, Linear Approximation of Shortest Superstrings. In *Proc. 31th Ann. Symp. Found. Comput. Sci.*, 1990, 554–562.

E. Dahlhaus, D. S. Johnson, C. H. Papadimitriou, P. D. Seymour, and M. Yannakakis, The Complexity of Multiway Cuts. In *Proc. Twenty-forth Ann. ACM Symp. Theor. Comput.*, 1992, 241–251.

U. Feige, S. Goldwasser, L. Lovász, S. Safra, and M. Szegedy, Approximating clique is almost NP-complete. In *Proc. 32th Ann. Symp. Found. Comput. Sci.*, 1991, 2–12.

O. Gabber and Z. Galil, Explicit Construction of linear sized superconcentrators. *J. Comput. System Sci.* **22** (1981), 407–420.

M. R. Garey and D. S. Johnson, The complexity of near-optimal graph coloring. *J. Assoc. Comput. Mach.* **23** (1976), 43–49.

M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman, 1979.

M. R. Garey, D. S. Johnson, and L. J. Stockmeyer, Some Simplified NP-Complete Graph Problems. *Theoret. Comput. Sci.* **1** (1976), 237–267.

J. Hastad, S. Phillips, and S. Safra, A well Characterized Approximation Problem. In *Proceedings of the 2nd Israel Symposium on Theory of Computing and Systems*, 1993, 261–265.

I. Hoyler, The NP-Competeness of Edge Coloring. *SIAM J. Comput.* **10** (1981), 718–720.

V. Kann, Maximum Bounded 3-Dimensional Matching is MAX SNP-Complete. *Inform. Process. Lett.* **37** (1991), 27–35.

R. M. Karp, Reducibility among combinatorial problems. In *Complexity of Computer Computations*, ed. Raymond E. Miller and James W. Thatcher, 85–103. Plenum Press, 1972.

D. Leven and Z. Galil, NP-completness of finding the chromatic index of regular graphs . *J. Algorithms* **4** (1983), 35–44.

L. Lovász, Coverings and Colorings of Hypergraphs. In *Proc. 4-th Southern Conference on Combinatorics, Graph Theory, and Computing*. Utilitas Mathematica Publishing, 1973, 3–12.

C. Lund and M. Yannakakis, On the Hardness of Approximating Minimization Problems. In *Proc. Twenty-fifth Ann. ACM Symp. Theor. Comput.*, 1993, 286–293.

G. A. Margulis, Explicit Constructions of Concentrators. *Comm. ACM* **9** (1973), 71–80. (English translation in *Problems Inform. Transmission*, 1975, 325–332.).

R. Motwani, Lecture Notes on Approximation Algorithms. Technical report, Dept. of Computer Science, Stanford University, 1992.

C. H. Papadimitriou and M. Yannakakis, Optimization, Approximation, and Complexity Classes. *J. Comput. System Sci.* **43** (1991), 425–440.

C. H. Papadimitriou and M. Yannakakis, The Traveling Salesman Problem with Distances One and Two. *Mathematics of Operations Research* (to appear).

S. Sahni and T. Gonzalez, P-complete approximation problems. *J. Assoc. Comput. Mach.* **23** (1976), 555–565.

L. J. Stockmeyer, Planar 3-Colorability is NP-Complete. *SIGACT News* **5**(3) (1973), 19–25.

Manuscript received 11 May 1993

Erez Petrank
Department of Computer Science
Technion - Israel Institute of Technology
Technion city
Haifa, Israel 32000
erez@cs.technion.ac.il