On the Hardness of Satisfiability with Bounded Occurrences in the Polynomial-Time Hierarchy

Ishay Haviv, Oded Regev*, Amnon Ta-Shma†

Received: July 28, 2006; published: March 28, 2007.

Abstract: In 1991, Papadimitriou and Yannakakis gave a reduction implying the NP-hardness of approximating the problem 3-SAT with bounded occurrences. Their reduction is based on expander graphs. We present an analogue of this result for the second level of the polynomial-time hierarchy based on superconcentrator graphs. This resolves an open question of Ko and Lin (1995) and should be useful in deriving inapproximability results in the polynomial-time hierarchy.

More precisely, we show that given an instance of ∀∃-3-SAT in which every variable occurs at most B times (for some absolute constant B), it is Π2-hard to distinguish between the following two cases: YES instances, in which for any assignment to the universal variables there exists an assignment to the existential variables that satisfies all the clauses, and NO instances in which there exists an assignment to the universal variables such that any

*Supported by an Alon Fellowship, by the Binational Science Foundation, by the Israel Science Foundation, and by the EU Integrated Project QAP.
†Supported by the Binational Science Foundation, by the Israel Science Foundation, and by the EU Integrated Project QAP.

ACM Classification: F.1.3

AMS Classification: 03D15, 68Q17

Key words and phrases: satisfiability, polynomial-time hierarchy, expander graphs, superconcentrator graphs
assignment to the existential variables satisfies at most a $1 - \varepsilon$ fraction of the clauses. We also generalize this result to any level of the polynomial-time hierarchy.

1 Introduction

In the problem $\forall \exists$-3-SAT, given a 3-CNF formula we have to decide whether for any assignment to a set of universal variables $X$ there exists an assignment to a set of existential variables $Y$, such that the formula is satisfied. Here, by a 3-CNF formula we mean a conjunction of clauses where each clause is a disjunction of at most 3 literals. This problem is a standard $\Pi_2$-complete problem. We denote the corresponding gap problem by $\forall \exists$-3-SAT $[1 - \varepsilon_1, 1 - \varepsilon_2]$ where $0 \leq \varepsilon_2 < \varepsilon_1 \leq 1$. This is the problem of deciding whether for any assignment to the universal variables there exists an assignment to the existential variables such that at least a $1 - \varepsilon_2$ fraction of the clauses are satisfied, or there exists an assignment to the universal variables such that any assignment to the existential variables satisfies at most a $1 - \varepsilon_1$ fraction of the clauses. The one-sided error gap problem $\forall \exists$-3-SAT $[1 - \varepsilon, 1]$ is $\Pi_2$-hard for some $\varepsilon > 0$, as was shown in [6]. This problem has the perfect completeness property, i.e., in YES instances it is possible to satisfy all the clauses.

In this paper we consider a restriction of $\forall \exists$-3-SAT, known as $\forall \exists$-3-SAT-B. Here, each variable appears at most $B$ times where $B$ is some constant. In [7], Ko and Lin showed that $\forall \exists$-3-SAT-B $[1 - \varepsilon_1, 1 - \varepsilon_2]$ is $\Pi_2$-hard for some constants $B$ and $0 < \varepsilon_2 < \varepsilon_1 < 1$. Our main result is that the problem is still $\Pi_2$-hard for some $\varepsilon_1 > 0$ with $\varepsilon_2 = 0$, i.e., with perfect completeness. This solves an open question given in [7].

**Theorem 1.1.** The problem $\forall \exists$-3-SAT-B $[1 - \varepsilon, 1]$ is $\Pi_2$-hard for some constants $B$ and $\varepsilon > 0$. Moreover, this is true even when the number of literals in each clause is exactly 3.

We note that the problem remains $\Pi_2$-hard even if the number of occurrences of universal variables is bounded by 2 and the number of occurrences of existential variables is bounded by 3. As we will explain later, these are the least possible constants for which the problem is still $\Pi_2$-hard unless the polynomial-time hierarchy collapses. We believe that Theorem 1.1 is useful for deriving $\Pi_2$-hardness results, as well as $\Pi_2$ inapproximability results. In fact, Theorem 1.1 was crucial in a recent proof that the covering radius problem on lattices with high norms is $\Pi_2$-hard [5]. Moreover, using Theorem 1.1, one can simplify the proof that the covering radius on codes is $\Pi_2$-hard to approximate [4].

At a very high level, the proof is based on the following ideas. First, one can reduce the number of occurrences of existential variables by an expander construction in much the same way as was done by Papadimitriou and Yannakakis [10]. The main difficulty in the proof is in reducing the number of occurrences of universal variables: If we duplicate universal variables (as is usually done in order to reduce the number of occurrences), we have to deal with inconsistent assignments to the new universal variables (this problem shows up in the completeness proof). The approach taken by Ko and Lin [7] is to duplicate universal variables and to add existential variables on top of the universal variables. Their construction, in a way, enables the existential variables to override inconsistent assignments to the universal variables. Unfortunately, it seems that this technique cannot produce instances with perfect completeness. In our approach we also duplicate the universal variables, but instead of using them
directly in the original clauses, we use a superconcentrator-based gadget, whose purpose is intuitively
to detect the majority among the duplicates of a universal variable. Crucially, this gadget requires only
a constant number of occurrences of each universal variable.

The rest of the paper is organized as follows. Section 2 provides some background about satisfiability
problems in the second level of the polynomial-time hierarchy and about some explicit expanders and
superconcentrators. In Section 3 we prove Theorem 1.1. Section 4 discusses the least possible value of
B for which the problem remains $\Pi_2$-hard. In Section 5 we generalize our main theorem to any level of
the polynomial-time hierarchy.

2 Preliminaries

2.1 $\Pi_2$ satisfiability problem

A $D$-CNF formula over a set of variables is a conjunction of clauses where each clause is a disjunction
of at most $D$ literals. Each literal is either a variable or its negation. A clause is satisfied by a Boolean
assignment to the variables if it contains at least one literal that evaluates to True.

For any reals $0 \leq \alpha < \beta \leq 1$ and positive integer $D > 0$, we define:

**Definition 2.1** ($\forall \exists$-D-SAT $[\alpha, \beta]$). An instance of $\forall \exists$-D-SAT $[\alpha, \beta]$ is a $D$-CNF Boolean formula $\Psi(X,Y)$
over two sets of variables. We refer to variables in $X$ as universal variables and to those in $Y$ as existential
variables. In YES instances, for every assignment to $X$ there exists an assignment to $Y$ such that at least a
$\beta$ fraction of the clauses are satisfied. In NO instances, there exists an assignment to $X$ such that for
every assignment to $Y$ at most an $\alpha$ fraction of the clauses are satisfied.

The problem $\forall \exists$-D-SAT $[\alpha, \beta]$ is the basic approximation problem in the second level of the
polynomial-time hierarchy (see [11, 12] for a recent survey on the topic of completeness and hard-
ness of approximation in the polynomial-time hierarchy). We also define some additional variants of
the above problem. For any $B \geq 1$ the problem $\forall \exists$-D-SAT-$B[\alpha, \beta]$ is defined similarly except that each
variable occurs at most $B$ times in $\Psi$. In the instances of the problem $\forall \exists$-D-SAT-$B[\alpha, \beta]$, the bound $B$
on the number of occurrences applies only to the universal variables (as opposed to all variables).

In [7] it was shown that $\forall \exists$-3-SAT-$B[1-\epsilon_1,1-\epsilon_2]$ is $\Pi_2$-hard for some $B$ and some $0 < \epsilon_2 < \epsilon_1 < 1$.
As already mentioned, in Section 3 we show that it is $\Pi_2$-hard even for some $B, \epsilon_1 > 0$ and $\epsilon_2 = 0$.

2.2 Expanders and superconcentrators

In this subsection, we gather some standard results on explicit constructions of expanders and supercon-
centrators (where by explicit we mean constructible in polynomial time). The first shows the existence
of certain regular expanders.

**Lemma 2.2** ([8, 9]). There exists a universal constant $C_1$ such that for any integer $n$, there is an explicit
14-regular graph $G = (V,E)$ with $n \leq |V| \leq C_1 n$ vertices, such that any nonempty set $S \subseteq V$ satisfies
$|E(S,\bar{S})| > \min(|S|,|\bar{S}|)$.

For the second, we need to define the notion of a superconcentrator.
Definition 2.3 (n-superconcentrator). A directed acyclic graph \( G = (U \cup V \cup W, E) \) where \( U \) denotes a set of \( n \) inputs (i.e., vertices with indegree 0) and \( V \) denotes a set of \( n \) outputs (i.e., vertices with outdegree 0) is an \( n \)-superconcentrator if for any subset \( S \) of \( U \) and any subset \( T \) of \( V \) satisfying \( |S| = |T| \), there are \( |S| \) vertex-disjoint directed paths in \( G \) from \( S \) to \( T \).

The explicit construction of sparse superconcentrators has been extensively studied. Gabber and Galil [3] were the first to give an explicit expander-based construction of \( n \)-superconcentrator with \( O(n) \) edges. Alon and Capalbo [1] presented the most economical known explicit \( n \)-superconcentrators, in which the number of edges is \( 44n + O(1) \). Their construction is based on a modification of the well-known construction of Ramanujan graphs by Lubotzky, Phillips and Sarnak [8] and by Margulis [9]. The following theorem of [1] summarizes some of the properties of their graphs.

Theorem 2.4 ([1]). There exists an absolute constant \( k > 0 \) for which the following holds. For any \( n \) of the form \( k \cdot 2^l \) (\( l \geq 0 \)) there exists an explicit \( n \)-superconcentrator \( H = (U \cup V \cup W, E) \) with \( |E| = 44n + O(1) \) and all of whose vertices have indegree and outdegree at most 11.

In our reduction, we use a slight modification of the superconcentrator in Theorem 2.4. This graph is described in the following claim (see Figure 1 for an illustration of the construction).

![Figure 1: The graph \( G^{(6)} \). All edges are directed downwards. The marked subgraph is a 6-superconcentrator (but not necessarily the one from [1]).](image)

Claim 2.5. There exist absolute constants \( c \) and \( d \) for which the following holds. For any natural \( n \geq 1 \) there exists an explicit directed acyclic graph \( G^{(n)} = (U \cup V \cup W, E) \) with a set \( U \) of \( 2n \) inputs (i.e., vertices with indegree 0) with outdegree 1 and a set \( V \) of \( n \) outputs (i.e., vertices with outdegree 0), such that for any subset \( S \) of \( U \) of size \( |S| = n \) there are \( n \) vertex-disjoint directed paths from \( S \) to \( V \). Moreover, \( |E| \leq cn \) and all indegrees and outdegrees in \( G^{(n)} \) are bounded by \( d \).
ON THE HARDNESS OF SATISFIABILITY WITH BOUNDED OCCURRENCES

Proof. Fix some \( n \geq 1 \). By Theorem 2.4 there exists an explicit \( n_0 \)-superconcentrator \( H' = (U' \cup V' \cup W', E') \) for some \( n + k \leq n_0 < 2(n + k) \) where \( k \) is the constant from Theorem 2.4, such that \( |E'| = 44n_0 + O(1) \) and all its indegrees and outdegrees are bounded by 11. Denote by \( U'' = \{u''_1, \ldots, u''_m\} \) and by \( V = \{v_1, \ldots, v_n\} \) arbitrary subsets of \( U' \) and \( V' \) of size exactly \( n \).

In order to construct the graph \( G(n) \) we add to the graph \( H' \) the \( 2n \) vertices \( U' = \{u_1, \ldots, u_{2n}\} \) and \( 2n \) edges. The input set of the graph \( G(n) \) is \( U' \), and the output set of \( G(n) \) is \( V \). For each \( i \in \{1, \ldots, n\} \) we add the directed edges \( (u_i, u''_i) \) and \( (u_{i+n}, v_i) \). In other words, we add to the graph two matchings of size \( n \): the first between the vertex sets \( \{u_1, \ldots, u_n\} \) and \( U'' \), and the second between \( \{u_{n+1}, \ldots, u_{2n}\} \) and \( V \).

It is easy to see that our graph satisfies the required properties for large enough absolute constants \( c \) and \( d \). Let \( S \subseteq U \) be of size \( n \), and define \( S_1 = S \cap \{u_i : 1 \leq i \leq n\} \) and \( S_2 = S \cap \{u_i : n + 1 \leq i \leq 2n\} \). We show that there exist \( n \) vertex-disjoint paths from \( S \) to \( V \). According to our construction, the vertices of \( S_2 \) have paths of length 1 to their neighbors in \( V \). So it suffices to show that the vertices of \( S_1 \) have vertex-disjoint paths to the \( n - |S_2| = |S_1| \) remaining vertices of \( V \). According to the property of \( H' \), there exist vertex-disjoint paths in \( G(n) \) between the neighbors of \( S_1 \) in \( U'' \) and the \( n - |S_2| \) vertices of \( V \). Combining these paths together with the matching edges between \( S_1 \) and \( U'' \) completes the proof. \( \square \)

3 Hardness of approximation for \( \forall \exists \)-3-SAT-\( B \)

In this section we prove Theorem 1.1. The proof is by reduction from the problem \( \forall \exists \)-3-SAT\([1 - \varepsilon, 1]\), which was shown to be \( \Pi_2 \)-hard for some \( \varepsilon > 0 \) in [6]. The reduction is performed in three steps. The first step is the main one, and it is here that we present our new superconcentrator-based construction. The remaining two steps are standard (see for example [14] and [2]) and we include them mainly for completeness. We remark that these two steps are also used in [7].

Step 1: Here we reduce the number of occurrences of each universal variable to at most some constant \( B \). As a side effect, the size of the clauses grows from being at most 3 to being at most \( D \), where \( D \) is some constant. More precisely, we establish that there exist absolute constants \( B, D \) and \( \varepsilon > 0 \) such that the problem \( \forall \exists \)-\( D \)-SAT-\( B \)\([1 - \varepsilon, 1]\) is \( \Pi_2 \)-hard.

Step 2: Here we reduce the number of occurrences of the existential variables to some constant \( B \). Notice that we must make sure that this does not affect the number of occurrences of the universal variables. More precisely, we show that there exist absolute constants \( B, D \) and \( \varepsilon > 0 \) such that the problem \( \forall \exists \)-\( D \)-SAT-\( B \)\([1 - \varepsilon, 1]\) is \( \Pi_2 \)-hard.

Step 3: Finally, we modify the formula such that the size of the clauses is exactly 3. Clearly, we must make sure that the number of occurrences of each variable remains constant. This would complete the proof of Theorem 1.1.

3.1 Step 1

Before presenting the first step we offer some intuition. In order to make the number of occurrences of the universal variables constant we replace their occurrences by new and distinct existential variables. In detail, assume \( x \) is a universal variable that occurs \( \ell \) times in an instance \( \Psi \) of \( \forall \exists \)-3-SAT\([1 - \varepsilon, 1]\).
For each variable we construct the graph $G^{(ℓ)} = (U \cup V \cup W, E)$ given in Claim 2.5 and identify its $ℓ$ output vertices $V$ with the $ℓ$ new existential variables. In addition, we associate a universal variable with each of the $2ℓ$ vertices of $U$, and an existential variable with each vertex in $W$ and also with each edge in $E$. We add clauses that verify that in the subgraph of $G^{(ℓ)}$ given by the edges with value $\text{True}$, there are $ℓ$ vertex-disjoint paths from $U$ to $V$ (and hence each vertex in $V$ has one incoming path). We also add clauses that verify that if an edge has value $\text{True}$ then both its endpoints must have the same value. This guarantees that each variable in $V$ gets the value of one of the variables in $U$. Completeness follows because for any assignment to $U$, we can assign all the variables in $V$ to the same value by connecting them to those variables in $U$ that get the more popular assignment (recall that $|U| = 2|V|$ and the properties given in Claim 2.5). For the proof of soundness, we show that if all the $U$ variables are assigned the same value, then all the $V$ variables should also be assigned this value.

### 3.1.1 The reduction

The proof is by reduction from the problem $\forall\exists$-$3$-SAT $[1−\varepsilon, 1]$ which is $Π_2$-hard for some constant $\varepsilon > 0$ as shown in [6]. Let $Ψ(X, Y)$ be a $3$-CNF Boolean formula with $m$ clauses over the set of variables $X \cup Y$, where $X = \{x_1, \ldots, x_{|X|}\}$ is the set of universal variables, and $Y = \{y_1, \ldots, y_{|Y|}\}$ is the set of existential variables. The reduction constructs a formula $Ψ'(X', Y')$ over $X' \cup Y'$. The number of occurrences in $Ψ'$ of each universal variable from $X'$ will be bounded by an absolute constant $B$, and the number of literals in each clause will be at most $D$. In fact, these constants are $B = 2$ and $D = d + 1$, where $d$ is given in Claim 2.5.

For each universal variable $x_i \in X$ denote by $ℓ_i$ the number of its occurrences in the formula $Ψ$, and apply Claim 2.5 to obtain the graph $G_i = G^{(ℓ_i)} = (U_i \cup V_i \cup W_i, E_i)$. Recall that the maximum degree (indegree and outdegree) of these graphs is bounded by some constant $d$ and that the number of edges in $G_i$ is bounded by $c \cdot ℓ_i$ for some constant $c$. Denote the vertex sets of $G_i$ by

\[
V_i = \{v_1^{(i)}, \ldots, v_{ℓ_i}^{(i)}\}, \quad U_i = \{u_1^{(i)}, \ldots, u_{2ℓ_i}^{(i)}\}, \quad \text{and} \quad W_i = \{w_1^{(i)}, \ldots, w_{|W_i|}^{(i)}\},
\]

and its edge set by $E_i = \{e_1^{(i)}, \ldots, e_{|E_i|}^{(i)}\}$. The set of existential variables in $Ψ'$ is

\[
Y' = \bigcup_{i=1}^{|X|} (V_i \cup W_i \cup E_i) \cup Y.
\]

The set of universal variables in $Ψ'$ is $X' = \bigcup_{i=1}^{|X|} U_i$.

The clauses of $Ψ'$ are divided into the following five types (see Figure 2).

1. **Major clauses:** These clauses are obtained from clauses of the formula $Ψ$, by replacing the $j$th occurrence of the universal variable $x_i$ with the variable

   \[
v_j^{(i)} \in V_i
   \]

   for $1 \leq i \leq |X|$, $1 \leq j \leq ℓ_i$. The number of clauses of this type is $m$.  

2. **Outdegree clauses:** These clauses verify that among the directed edges leaving a vertex in $G_i$, at most one has value True. For each vertex $w$, we add the clause

$$(\neg e_{j_1}^{(i)} \lor \neg e_{j_2}^{(i)})$$

for each pair of edges $e_{j_1}^{(i)}, e_{j_2}^{(i)}$ leaving $w$. Each such clause is duplicated $d^2$ times. The number of clauses of this type is at most $\ell_i \cdot c \cdot d^2(d^2)$ for each $i$.

3. **Flow clauses:** These clauses verify for any vertex $w_{j}^{(i)} \in W_i$ that if at least one of its outward edges has value True then there exists also an edge entering $w_{j}^{(i)}$ with value True. This is done by adding a clause of the form

$$(\neg e_{j}^{(i)} \lor e_{j_1}^{(i)} \lor \cdots \lor e_{j_{d'}}^{(i)})$$

for each $e_{j}^{(i)}$ leaving $w_{j}^{(i)}$ where $e_{j_1}^{(i)}, \ldots, e_{j_{d'}}^{(i)}$ are all the $0 \leq d' \leq d$ edges entering $w_{j}^{(i)}$. The number of clauses of this type is at most $c \cdot \ell_i$ for each $i$.

4. **$V$-degrees clauses:** These clauses verify that each vertex $v_{j}^{(i)}$ has at least one incident edge with True value. This is done by adding one clause of the form

$$(e_{j_1}^{(i)} \lor \cdots \lor e_{j_{d'}}^{(i)})$$

where $e_{j_1}^{(i)}, \ldots, e_{j_{d'}}^{(i)}$ are the $d' \leq d$ edges incident to $v_{j}^{(i)}$. The number of clauses of this type is $\ell_i$ for each $i$.

5. **Edge consistency clauses:** For each edge $e_{j}^{(i)} \in E_i$ do the following. Let $w_{j_1}^{(i)}, w_{j_2}^{(i)} \in U_i \cup V_i \cup W_i$ be its endpoints. Add the two clauses

$$(\neg e_{j}^{(i)} \lor w_{j_1}^{(i)} \lor \neg w_{j_2}^{(i)})$$

and

$$(\neg e_{j}^{(i)} \lor \neg w_{j_1}^{(i)} \lor w_{j_2}^{(i)})$$

Figure 2: An illustration of the reduction for the case $\ell = 6$. 

---

**Theory of Computing**, Volume 3 (2007), pp. 45–60
Our goal in the completeness proof is to show that if $\Psi$ only in clauses of type $(2)$, then for any assignment to $X$. Recall that $X$ is the union $\bigcup_{i=1}^{X_i} U_i$. We define an assignment $t$ to $X$ based on the majority of the assignments given by $t'$. More formally,

$$t(x_i) = \begin{cases} 
  \text{True,} & \text{if } |\{j : t'(u_j^{(i)}) = \text{True}\}| \geq \ell_i \\
  \text{False,} & \text{otherwise.}
\end{cases}$$

By the assumption on the original formula $\Psi'(X, Y)$, the assignment $t$ can be extended to $X \cup Y$, in a way that satisfies all the clauses in $\Psi'(X, Y)$. Let us extend the assignment $t'$ to the existential variables

$$Y' = \left( \bigcup_{i=1}^{X} (V_i \cup W_i \cup E_i) \right) \cup Y.$$

First, let the assignment $t'$ give the same values as $t$ for the variables in $Y$. For each $i$ denote by $S_i \subseteq U_i$ a set of vertices from $U_i$ of size $|S_i| = \ell_i$ in which every variable has value $t(x_i)$. There exists such a set according to the definition of $t$. By Claim 2.5 there are $\ell_i$ vertex-disjoint directed paths in $G_i$ from $S_i$ to $V_i$. We define $t'(e_j^{(i)})$ to be True if $e_j^{(i)}$ appears in one of these paths and False otherwise. In addition, $t'$ gives the value $t(x_i)$ to all variables in $V_i \cup W_i$.

We now check that the assignment $t'$ satisfies all clauses in $\Psi'$. The assignment to the variables in $V_i$ is $t(x_i)$. Since the variables $Y$ are also assigned according to $t$, all clauses of type $(1)$ are satisfied. The paths given by Claim 2.5 are vertex-disjoint. In particular, every vertex has at most one outward edge assigned to True, so all clauses of type $(2)$ are satisfied too. Moreover, if at least one of the edges leaving a vertex $w \in W_i$ has value True then there exists also a directed edge with value True entering $w$. Therefore, the clauses of type $(3)$ are satisfied. The number of paths in $G_i$ is $\ell_i$, so there is one path reaching every vertex in $V_i$. This means that the clauses of type $(4)$ are satisfied too. Finally, our assignment gives the value $t(x_i)$ to all variables in $S_i \cup V_i \cup W_i$. In particular, each edge assigned to True has both its endpoints with the same value. Thus, the clauses of type $(5)$ are satisfied, as required.

### 3.1.3 Soundness

In the soundness proof we assume $\Psi(X, Y)$ is a NO instance of $\forall\exists$-3-SAT$[1 - \varepsilon, 1]$. We will show the existence of an assignment to $X'$ for which any assignment to $Y'$ satisfies at most $(1 - \varepsilon')m'$ clauses of $\Psi'(X', Y')$ for $\varepsilon' = \varepsilon / C$, and hence the theorem will follow.
Let \( t \) be an assignment to \( X \) such that every extension of \( t \) to \( X \cup Y \) satisfies at most \((1 - \varepsilon)m\) clauses in \( \Psi(X, Y) \). Define an assignment \( t' \) to \( X' \) in which every variable \( u_j^{(i)} \) has the value \( t(x_i) \). Extend \( t' \) to an assignment to \( X' \cup Y' \) in an arbitrary way. Our goal in the following is to show that the number of clauses satisfied by \( t' \) is at most \((1 - \varepsilon')m'\). We start with the following two claims.

**Claim 3.1.** Let \( t' \) be an assignment to \( X' \cup Y' \) as above. Then \( t' \) can be modified to an assignment \( t'' \) that satisfies every clause of type (2) and satisfies at least as many clauses as \( t' \) satisfies.

**Proof.** We obtain \( t'' \) by performing the following modification to \( t' \) for each \( i \): For each variable in \( W_i \), if it has more than one outward edge assigned to True by \( t' \), \( t'' \) assigns False to all its outward edges. Since we only modify variables in \( E_i \), clauses of type (1) are not affected. Moreover, since we only set edges to False, we do not decrease the number of satisfied clauses of type (5). We might, however, reduce the number of satisfied clauses of types (3) and (4) by at most \( d^2 \) for each variable (at most \( d \) for each out-neighbor of the vertex). On the other hand, the corresponding clause of type (2) is satisfied by \( t'' \), and by the duplication, this amounts to at least \( d^2 \) additional satisfied clauses. In total, the number of clauses satisfied by \( t'' \) is at least the number of clauses satisfied by \( t' \), and the claim follows.

**Claim 3.2.** Let \( t' \) be an assignment to \( X' \cup Y' \) that satisfies all clauses of type (2). Denote by \( k \) the number of vertices \( v_j^{(i)} \in \bigcup_i V_i \) satisfying \( t'(v_j^{(i)}) \neq t(x_i) \), where \( t \) is the assignment to \( X \) as above. Then at least \( k \) clauses of types (3), (4) or (5) are unsatisfied by \( t' \).

**Proof.** Fix some \( i \). It suffices to show that to each vertex \( v_j^{(i)} \) satisfying \( t'(v_j^{(i)}) \neq t(x_i) \) we can assign in a one-to-one fashion a clause of type (3), (4) or (5) which is not satisfied by \( t' \). To show this let \( G' \) be the subgraph of \( G_i \) given by the edges assigned to True by \( t' \). Let \( A_j \) be the set of vertices that have a directed path in \( G' \) to \( v_j^{(i)} \). Since clauses of type (2) are all satisfied by \( t' \), the sets \( A_j \) are pairwise disjoint. Fix some \( 1 \leq j \leq \ell_i \) such that \( t'(v_j^{(i)}) \neq t(x_i) \). Since \( G_i \) is acyclic, \( A_j \) contains a vertex \( u \) whose indegree in \( G' \) is 0. If \( u \) is in \( U_i \) then at least one of the clauses of type (5) on the path from \( u \) to \( v_j^{(i)} \) is unsatisfied by \( t' \), because \( t'(u) = t(x_i) \) whereas \( t'(v_j^{(i)}) \neq t(x_i) \). Otherwise at least one of the clauses of types (3) and (4) is unsatisfied by \( t' \). Therefore, we see that the number of clauses of type (3)-(5) unsatisfied by \( t' \) is at least the number of vertices \( v_j^{(i)} \) satisfying \( t'(v_j^{(i)}) \neq t(x_i) \).

Recall that \( t' \) is an assignment to \( X' \cup Y' \) that assigns every variable \( u_j^{(i)} \) to \( t(x_i) \). We have to show that \( t' \) satisfies at most \((1 - \varepsilon')m' \) clauses in \( \Psi' \). By Claim 3.1 we can assume that \( t' \) satisfies all clauses of type (2) in \( \Psi' \).

Now, we define an assignment \( t'' \) to \( X' \cup Y' \) as follows. For each \( i \), let \( S_i \) be an arbitrary subset of \( U_i \) of size \( \ell_i \). We know that there exist \( \ell_i \) directed vertex-disjoint paths from \( S_i \) to \( V_i \) in \( G_i \). The assignment \( t'' \) assigns all the \( e_j^{(i)} \) in these paths to True and all other \( e_j^{(i)} \) to False. Moreover, \( t'' \) gives all variables in \( U_i \cup V_i \cup W_i \) the value \( t(x_i) \). Finally, we define \( t'' \) on \( Y \) to be identical to \( t' \). Notice that in \( t'' \) all clauses of type (2)-(5) are satisfied. Denote by \( k \) the number of the variables \( v_j^{(i)} \) satisfying \( t'(v_j^{(i)}) \neq t(x_i) \). Then the number of type (1) clauses satisfied by \( t'' \) is smaller than that of \( t' \) by at most \( k \). Moreover, \( t' \) satisfies all clauses of type (2), so by Claim 3.2 at least \( k \) clauses of type (3)-(5) are unsatisfied by \( t' \). In total, the number of clauses satisfied by \( t'' \) is at least the number of clauses satisfied by \( t' \).
Finally, by our assumption on \( \Psi \) and on \( t \) we get that at least \( \epsilon m \) clauses of type (1) are not satisfied by \( t'' \). So the number of satisfied clauses is at most \( m' - \epsilon m \leq (1 - \epsilon')m' \), as required.

### 3.2 Step 2

With Step 1 proven, we now apply an idea of [10] to show that there are absolute constants \( B \) and \( \epsilon > 0 \), for which the problem \( \forall \exists \text{-D-SAT-B}[1 - \epsilon, 1] \) is \( \Pi_2 \)-hard. This proof uses the expander graphs from Lemma 2.2.

**The reduction:** Consider the \( \Pi_2 \)-hard problem \( \forall \exists \text{-D-SAT-B}[1 - \epsilon', 1] \) for some \( \epsilon' > 0 \). Let \( \Psi(X, Y) \) be an instance of this problem. For every existential variable \( y_i \in Y \) (1 \( \leq i \leq |Y'| \)) denote by \( n_i \) the number of the occurrences of \( y_i \) in \( \Psi \). Assuming \( n_i \) is large enough, consider the graph \( G_i = (V_i, E_i) \) given by Lemma 2.2 for \( n_i \), with \( n_i \leq |V_i| \leq C_1 n_i \). (If \( n_i \) is not large enough, we do not need to modify this variable.) Label the vertices of \( G_i \) with \( |V_i| \) new distinct existential variables \( Y_i = \{y^{(i)}_1, \ldots, y^{(i)}_{|V_i|}\} \). We construct a new Boolean formula \( \Psi'(X, Y') \) over the universal variables in \( X \) and the existential variables in \( Y' = \bigcup_{1 \leq i \leq |Y|} Y_i \). First, for each \( 1 \leq i \leq |Y| \) replace the occurrences of \( y_i \) by \( n_i \) distinct variables of \( Y_i \). Second, for each edge \( (y^{(i)}_j, y^{(i)}_j) \) in \( G_i \), add to \( \Psi \) the two clauses

\[
(-y^{(i)}_j \lor y^{(i)}_j) \quad \text{and} \quad (y^{(i)}_j \lor \neg y^{(i)}_j)
\]

which are both satisfied if and only if the variables \( y^{(i)}_j, y^{(i)}_j \) have the same value. The number of clauses in \( \Psi' \) is linear in \( \sum n_i \leq Dm \). Notice, that the number of occurrences of each variable in \( \Psi' \) is bounded by a constant.

**Correctness:** Let \( \Psi(X, Y) \), an \( m \) clause formula, be a YES instance, i.e., for every assignment to \( X \) there exists an assignment to \( Y \) such that every clause in \( \Psi \) is satisfied. Clearly, for any assignment to \( X \) there exists an assignment to \( Y' \) which satisfies all the clauses in \( \Psi' \), because we can set the \( Y_i \) variables the value of \( y_i \) in \( \Psi \). Now, assume \( \Psi \) is a NO instance, so there is an assignment \( t \) to \( X \) such that for any assignment to \( Y \) at least \( \epsilon' m \) clauses are unsatisfied in \( \Psi \). Let \( t' \) be an arbitrary extension of \( t \) to \( X \cup Y' \). If for some \( 1 \leq i \leq |Y| \), \( t' \) does not assign to all the \( Y_i \) variables the same value for some \( 1 \leq i \leq |Y'| \), it is possible to improve the number of satisfied clauses by setting all the \( Y_i \) variables to the majority vote of \( t' \) on \( Y_i \). Indeed, denote by \( S_i \) the set of variables in \( Y_i \) that were assigned by \( t' \) to True. This modification reduces the number of satisfied clauses by at most \( \min(|S_i|, |\overline{S}_i|) \), but satisfies at least \( |E(S_i, \overline{S}_i)| \) unsatisfied consistency clauses. Lemma 2.2 states that \( |E(S_i, \overline{S}_i)| > \min(|S_i|, |\overline{S}_i|) \), so this modification improves the number of satisfied clauses. Hence, we can assume that for each \( 1 \leq i \leq |Y| \), \( t' \) assigns to all the \( Y_i \) variables the same value for each \( 1 \leq i \leq |Y'| \). Thus, by the assumption on \( \Psi \) we conclude that \( t' \) does not satisfy at least \( \epsilon' m \) clauses, meaning at least an \( \epsilon'/D \) fraction of the clauses is not satisfied. Defining \( \epsilon = \epsilon'/D \) completes the proof.

### 3.3 Step 3

This subsection completes the proof of Theorem 1.1 by showing a reduction that modifies the size of the clauses to exactly 3.
The reduction: Let $\Psi(X,Y)$ be an instance of $\forall\exists$-D-SAT-$B[1-\varepsilon', 1]$ with $m$ clauses. We transform $\Psi$ into a formula $\Psi'(X',Y')$, whose clauses are of size exactly 3, as follows. For each clause of size 1, like $(a)$, we add a new universal variable $z$ and replace it by $(a \lor z \lor z)$. Similarly, for each clause of size 2, like $(a \lor b)$, we add a new universal variable $z$ and replace it by $(a \lor b \lor z)$. Now consider a clause $C = (u_1 \lor u_2 \lor \cdots \lor u_r)$ of size $r > 3$, where the $u_i$ are literals. For each such clause introduce $r - 3$ new and distinct existential variables $z_1, \ldots, z_{r-3}$ and replace $C$ in the formula $\Psi$ by the clauses of $C'$,

$$C' = (u_1 \lor u_2 \lor z_1) \land (\neg z_1 \lor u_3 \lor z_2) \land \cdots \land (\neg z_{r-4} \lor u_{r-2} \lor z_{r-3}) \land (\neg z_{r-3} \lor u_{r-1} \lor u_r).$$

The number of the clauses in $\Psi'$ is at most $Dm$. Obviously, the number of occurrences of each variable remains the same, and the newly added variables appear either once or twice.

Correctness: It is easy to see that if $\Psi$ is a YES instance then so is $\Psi'$ and that if $\Psi$ is a NO instance, then there exists an assignment to $X'$ such for any assignment $Y'$, at least $\varepsilon'm$ of the clauses of $\Psi'(X',Y')$ are unsatisfied. So for $\varepsilon = \varepsilon' / D$ we get the desired result.

4 On the number of occurrences

The output of the reduction of Section 3 is a formula in which every universal variable occurs at most twice and every existential variable occurs at most $B$ times for some constant $B$. By performing a transformation similar to the one in Step 2 with the graphs of Lemma 2.2 replaced by directed cycles, the number of occurrences of each existential variable can be made at most 3 (see for example Theorem 10.2, Part 1 in [2]). This implies that if we allow each universal variable to occur at most twice and each existential variable to occur at most 3 times, the problem remains $\Pi_2$-hard. Here, we show that 2 and 3 are the best possible constants (unless the polynomial-time hierarchy collapses).

First note that whenever a universal variable occurs only once in a formula, we can remove it without affecting the formula. Hence, if each universal variable occurs at most once, the problem is in $\text{NP}$ and thus is not $\Pi_2$-hard, unless the polynomial-time hierarchy collapses.

Moreover, if we allow every existential variable to occur at most twice, the problem lies in $\text{coNP}$ and is thus unlikely to be $\Pi_2$-hard. Given an assignment to the universal variables $X$, the formula $\Psi(X,Y)$ becomes a SAT formula in which each variable appears at most twice. Checking satisfiability of such formulas can be done in polynomial time [13]. Indeed, variables that appear only once and those that appear twice with the same sign can be removed from the formula together with the clauses that contain them. This means that we are left with a SAT formula in which each variable appears once as a positive literal and once as a negative one. So consider the bipartite graph $H = (A \cup B, E)$ in which $A$ is the set of clauses of $\Psi$ and $B$ is the set of its existential variables. We connect by an edge a clause in $A$ to a variable in $B$ if the clause contains the variable. Notice that there exists a matching in $H$ that saturates $A$ if and only if the formula is satisfiable. The existence of such a matching can be checked easily in polynomial time. Therefore $\forall\exists$-SAT restricted to instances in which every existential variable occurs at most twice is in $\text{coNP}$.
5 Extension to higher levels of the hierarchy

As one might expect, Theorem 1.1 can be generalized to any level of the polynomial-time hierarchy. In this section, we describe in some detail how this can be done. Our aim is to prove the following theorem (the problems below are the natural extension of \( \forall \exists \)-3-SAT to higher levels of the hierarchy; see [6]).

**Theorem 5.1.** For any \( r \geq 1 \) there exists an \( \varepsilon > 0 \) such that \((\forall \exists)^r\)-3-SAT-B[1–\( \varepsilon \), 1] is \( \Pi_{2r} \)-complete and \( \exists(\forall \exists)^r\)-3-SAT-B[1–\( \varepsilon \), 1] is \( \Sigma_{2r+1} \)-complete (where \( B \) is some absolute constant). Moreover, this is true even when the number of literals in each clause is exactly 3.

For convenience, we present the proof only for the even levels of the hierarchy (\( \Pi_{2r} \)). The case of odd levels is almost identical.

Our starting point is a result of [6], which says that for any \( r \geq 1 \) there exists an \( \varepsilon > 0 \) such that \((\forall \exists)^r\)-3-SAT[1–\( \varepsilon \), 1] is \( \Pi_{2r} \)-complete. As in Section 3, the proof proceeds in three steps. In the first we reduce the number of occurrences of universal variables. In the second we reduce the number of occurrences of existential variables. Finally, in the third step we modify the formula such that the size of each clause is exactly 3.

5.1 Step 1

In this step we show that for any \( \varepsilon > 0 \) there exists an \( \varepsilon ' > 0 \) such that \((\forall \exists)^r\)-3-SAT[1–\( \varepsilon \), 1] reduces to \((\forall \exists)^{-1}\)-D-SAT-B[1–\( \varepsilon ' \), 1] for some absolute constants \( D, B \) (where the latter problem is a restriction of the former to instances in which each universal variable appears at most \( B \) times). In more detail, given a 3-CNF formula \( \Psi \) on variable set \( X_1 \cup Y_1 \cup \cdots \cup X_r \cup Y_r \), we show how to construct a D-CNF formula \( \Psi ' \) on variable set \( X_1 ' \cup Y_1 ' \cup \cdots \cup X_r ' \cup Y_r ' \) in which each universal variable appears at most \( B \) times, and whose size is linear in the size of \( \Psi \), such that

\[
\max_{t_{X_1}, t_{Y_1}} \min_{t_{X_2}, t_{Y_2}} \cdots \min_{t_{X_r}, t_{Y_r}} \text{SAT}(\Psi, t_{X_1}, t_{Y_1}, \ldots, t_{X_r}, t_{Y_r})
\]

\[
= \max_{t_{X_1}', t_{Y_1}'} \min_{t_{X_2}', t_{Y_2}'} \cdots \min_{t_{X_r}', t_{Y_r}'} \text{SAT}(\Psi', t_{X_1}', t_{Y_1}', \ldots, t_{X_r}', t_{Y_r}'),
\]  

(5.1)

where \( \text{SAT} \) denotes the number of unsatisfied clauses in a formula for a given assignment. It is easy to see that this is sufficient to establish the correctness of the reduction. Moreover, it can be verified that in Step 1, Section 3 we proved Equation (5.1) for the case \( r = 1 \).

Before describing the reduction, we note that in Step 1, Section 3, the only property of the original formula that we used is that flipping the value of an occurrence of a variable can change the number of satisfied clauses by at most one. This leads us to the following lemma, whose proof was essentially given already in Step 1, Section 3.

**Lemma 5.2.** For any \( \ell \geq 1 \) there exists a \( k \geq \ell \) and a D-SAT formula \( \Phi(x_1, \ldots, x_2\ell, y_1, \ldots, y_k) \) (for some absolute constant \( D \)) on \( 2\ell + k \) variables of size \( O(\ell) \) in which each of the first \( 2\ell \) variables appears at most twice such that the following holds. For any integer-valued function \( f \) on \( \ell \) Boolean variables with the property that flipping any one variable changes the value of \( f \) by at most one, we have that

\[
\max_x f(x, \ldots, x) = \max_{x_1, \ldots, x_{2\ell}, y_1, \ldots, y_k} \min_{y_1, \ldots, y_{2\ell}} \left(f(y_1, \ldots, y_{2\ell}) + \text{SAT}(\Phi, x_1, \ldots, x_{2\ell}, y_1, \ldots, y_k) \right),
\]
where \( x, x_1, \ldots, x_2, y_1, \ldots, y_k \) are Boolean variables.

Using this lemma we can now describe our reduction. We are given a 3-CNF formula \( \Psi \) on variable set \( X_1 \cup Y_1 \cup \cdots \cup X_r \cup Y_r \). We perform the following modifications for each universal variable \( x \). Let \( i \) be such that \( x \in X_i \) and \( \ell \) be the number of times \( x \) occurs in \( \Psi \). Let \( k \) and \( \Phi \) be as given by Lemma 5.2. First, we replace \( x \in X_i \) with \( 2\ell \) new variables \( x_1, \ldots, x_2 \in X_i \) and add \( k \) new variables \( y_1, \ldots, y_k \) to \( Y_i \). Next, we replace the \( \ell \) occurrences of \( x \) with \( y_1, \ldots, y_\ell \). Finally, we append \( \Phi(x_1, \ldots, x_2, y_1, \ldots, y_k) \) to the formula. Let \( \Psi' \) be the resulting formula and \( X'_1 \cup Y'_1 \cup \cdots \cup X'_r \cup Y'_r \) be the resulting variable set. This completes the description of the reduction.

Clearly, each universal variable in \( \Psi' \) appears at most twice, and moreover, the size of \( \Psi' \) is linear in that of \( \Psi \). Therefore it remains to prove Equation (5.1). We do this by showing that for each universal variable, the modifications we perform leave the expression in Equation (5.1) unchanged. So let \( \Psi \) be an arbitrary formula on some variable set \( X_1 \cup Y_1 \cup \cdots \cup X_r \cup Y_r \), and let \( x \in X_i \) be a universal variable with \( \ell \) occurrences. It can be seen that our goal is to show that\(^1\)

\[
\max_{t_1} \min_{t_1} \cdots \max_{t_X} \min_{t_Y} \cdots \max_{t_X} \min_{t_Y} g(t_X, t_1, \ldots, t_{X \setminus \{x\}}, x, \ldots, y_1, \ldots, y_k) = \max_{t_1} \min_{t_1} \cdots \max_{t_X} \min_{t_Y} \cdots \max_{t_X} \min_{t_Y} \left(g(t_X, t_1, \ldots, t_{X \setminus \{x\}}, x, \ldots, y_1, \ldots, y_k) + \overline{\text{SAT}}(\Phi, x_1, \ldots, x_2, y_1, \ldots, y_k)\right),
\]

where \( g \) denotes the number of unsatisfied clauses in \( \Psi \) under the given assignment to all variables except \( x \) and to all occurrences of \( x \), and \( k \) and \( \Phi \) are as in Lemma 5.2. Clearly it suffices to prove this equality for any fixed setting to the variables quantified before \( x \), i.e.,

\[
\max_{t_X} \min_{t_Y} \max_{t_X} \min_{t_Y} g(t_X, t_1, \ldots, t_{X \setminus \{x\}}, x, \ldots, y_1, \ldots, y_k) = \max_{t_X} \min_{t_Y} \max_{t_X} \min_{t_Y} \left(g(t_X, t_1, \ldots, t_{X \setminus \{x\}}, x, \ldots, y_1, \ldots, y_k) + \overline{\text{SAT}}(\Phi, x_1, \ldots, x_2, y_1, \ldots, y_k)\right),
\]

but this follows from Lemma 5.2.

We conclude that \((\forall \exists)^{\ell} \text{-D-SAT-B}_\mathcal{V}[1 - \epsilon, 1] \) is \( \Pi_{2r} \)-hard for some \( \epsilon > 0 \).

### 5.2 Step 2

In this step we show that for any \( \epsilon > 0 \) there exists an \( \epsilon' > 0 \) such that \((\forall \exists)^{\ell} \text{-D-SAT-B}_\mathcal{V}[1 - \epsilon, 1] \) reduces to \((\forall \exists)^{\ell} \text{-D-SAT-B}[1 - \epsilon', 1] \) for some absolute constants \( D, B \). The following lemma is the analogue of Lemma 5.2 for existential variables, and its proof essentially appeared already in Step 2, Section 3.

**Lemma 5.3.** For any large enough \( \ell \) there exists a 2-SAT formula \( \Phi(y_1, \ldots, y_\ell) \) on \( \ell \) variables of size \( O(\ell) \) in which each variable appears at most \( B \) times (for some absolute constant \( B \)) such that the

\(^1\)We remark that the fact that we write \( \max_{t_X \setminus \{x\}} \max_x \) as opposed to \( \max_x \max_{t_X \setminus \{x\}} \) will be crucial when we apply Lemma 5.2, as this prevents an additional quantifier alternation.
following holds. For any integer-valued function \( f \) on \( \ell \) Boolean variables with the property that flipping any one variable changes the value of \( f \) by at most one, we have that

\[
\min_y f(y, \ldots, y) = \min_{y_1, \ldots, y_\ell} (f(y_1, \ldots, y_\ell) + \text{SAT}(\Phi, y_1, \ldots, y_\ell)),
\]

where \( y, y_1, \ldots, y_\ell \) are Boolean variables.

The reduction is as follows. We are given a D-CNF formula \( \Psi \) on variable set \( X_1 \cup Y_1 \cup \cdots \cup X_r \cup Y_r \). We perform the following modifications for each existential variable \( y \). Let \( i \) be such that \( y \in Y_i \) and \( \ell \) be the number of times \( y \) occurs in \( \Psi \). Let \( \Phi \) be as given by Lemma 5.3. First, we replace \( y \in Y_i \) with \( \ell \) variables \( y_1, \ldots, y_\ell \in Y_i \). Next, we replace the \( \ell \) occurrences of \( y \) with \( y_1, \ldots, y_\ell \). Finally, we append \( \Phi(y_1, \ldots, y_\ell) \) to the formula. This completes the description of the reduction. The proof of correctness is similar to the previous one and uses Lemma 5.3.

### 5.3 Step 3

To complete the proof of Theorem 5.1 we now modify the formula so that the number of literals in each clause is exactly 3. Given a formula \( \Psi \) on variable set \( X_1 \cup Y_1 \cup \cdots \cup X_r \cup Y_r \) we apply the modification of Step 3, Section 3. We add the new existential variables to \( Y_r \) and the new universal variables to \( X_r \). The proof of correctness is easy and is omitted.

### Acknowledgements

We thank Ker-I Ko for sending us a copy of [7]. Some of the early ideas that eventually led us to the construction of Section 3 were obtained while the second author was working on [4] together with Daniele Micciancio and Venkatesan Guruswami. We also thank two anonymous referees for their helpful comments.

### References

[1] * N. Alon and M. Capalbo*: Smaller explicit superconcentrators. *Internet Math.*, 1(2):151–163, 2004. [InternetMath:1(2):151–163]. 2.2, 2.4, 1

[2] * S. Arora and C. Lund*: Hardness of approximation. In Dorit S. Hochbaum, editor, *Approximation algorithms for NP-hard problems*. PWS, Boston, 1996. 3, 4

[3] * O. Gabber and Z. Galil*: Explicit constructions of linear-sized superconcentrators. *Journal of Computer and System Sciences*, 22(3):407–420, June 1981. [JCSS:10.1016/0022-0000(81)90040-4]. 2.2

[4] * V. Guruswami, D. Micciancio, and O. Regev*: The complexity of the covering radius problem on lattices and codes. *Computational Complexity*, 14(2):90–121, 2005. Preliminary version in CCC’04. [doi:10.1007/s00037-005-0193-y]. 1, 5.3
ON THE HARDNESS OF SATISFIABILITY WITH BOUNDED OCCURRENCES

[5] * I. HAVIV AND O. REGEV: Hardness of the covering radius problem on lattices. In Proc. of 21st Ann. Conf. on Computational Complexity (CCC’06), pp. 145–158. IEEE Computer Society Press, 2006. [CCC:10.1109/CCC.2006.23]. 1

[6] * K.-I. KO AND C.-L. LIN: Non-approximability in the polynomial-time hierarchy. Technical Report 94-2, Dept. of Computer Science, SUNY at Stony Brook, 1994. 1, 3, 3.1.1, 5, 5

[7] * K.-I. KO AND C.-L. LIN: On the longest circuit in an alterable digraph. J. Global Optimization, 7(3):279–295, 1995. [Springer:k74448w88p1483ww]. 1, 1, 2.1, 3, 5.3

[8] * A. LUBOTZKY, R. PHILLIPS, AND P. SARNAK: Ramanujan graphs. Combinatorica, 8(3):261–277, 1988. [Springer:k285687344657q53]. 2.2, 2.2

[9] * G. A. MARGULIS: Explicit group-theoretic constructions of combinatorial schemes and their applications in the construction of expanders and concentrators. Problemy Peredachi Informatsii, 24(1):51–60, 1988. 2.2, 2.2

[10] * C. H. PAPADIMITRIOU AND M. YANNAKAKIS: Optimization, approximation, and complexity classes. Journal of Computer and System Sciences, 43(3):425–440, 1991. [JCSS:10.1016/0022-0000(91)90023-X]. 1, 3.2

[11] * M. SCHAEFER AND C. UMANS: Completeness in the polynomial-time hierarchy: A compendium. SIGACT News, 33(3):32–49, September 2002. Guest column in Complexity Theory Column. [SIGACT:10.1145/582475.582484]. 2.1

[12] * M. SCHAEFER AND C. UMANS: Completeness in the polynomial-time hierarchy: Part II. SIGACT News, 33(4):22–36, December 2002. Guest column in Complexity Theory Column. [SIGACT:10.1145/601819.601829]. 2.1

[13] * C.A. TOVEY: A simplified NP-complete satisfiability problem. Discrete Applied Mathematics, 8(1):85–89, 1984. [Elsevier:10.1016/0166-218X(84)90081-7]. 4

[14] * V.V. VAZIRANI: Approximation algorithms. Springer-Verlag, Berlin, 2001. 3

AUTHORS

Ishay Haviv [About the author]
School of Computer Science
Tel Aviv University, Tel Aviv, Israel
I. Haviv, O. Regev, and A. Ta-Shma

Oded Regev [About the author]
Assistant professor
School of Computer Science
Tel Aviv University, Tel Aviv, Israel
http://www.cs.tau.ac.il/~odedr

Amnon Ta-Shma [About the author]
Assistant professor
School of Computer Science
Tel Aviv University, Tel Aviv, Israel
http://www.cs.tau.ac.il/~amnon

ABOUT THE AUTHORS

Ishay Haviv is a graduate student at the School of Computer Science, Tel Aviv University, under the supervision of Oded Regev. He is interested in theoretical computer science, especially the complexity of lattice problems. He also loves animals and acts for their rights.

Oded Regev graduated from Tel Aviv University in 2001 under the supervision of Yossi Azar. Before joining Tel Aviv University, he spent two years as a postdoctoral fellow at the Institute for Advanced Study, Princeton, and one year at the University of California, Berkeley. His research interests include quantum computation, computational aspects of lattices, and other topics in theoretical computer science. He also enjoys hiking, running, and photography.

Amnon Ta-Shma graduated from the Hebrew University in 1996 under the supervision of Noam Nisan. Before joining Tel Aviv University, he spent three years as a postdoctoral fellow at the International Computer Science Institute, Berkeley and the University of California, Berkeley. His research interests include the role of randomness in computation, quantum computation, and other topics in theoretical computer science. He also enjoys his family and thanks them for their love.