Electronic Colloquium on Computational Complexity - Reports Series 1996 - available via:
ECCC
TR96-016
FTP: ftp.eccc.uni-trier.de:/pub/eccc/
WWW: http://www.eccc.uni-trier.de/eccc/
Email: ftpmail@ftp.eccc.uni-trier.de with subject 'help eccc'

Improved Non-approximability Results for Minimum Vertex Cover with Density Constraints*

Andrea E. F. Clementi† and Luca Trevisan‡
Dipartimento di Scienze dell'Informazione
Università di Roma "La Sapienza"
Via Salaria 113, 00198 Roma, Italy
Email: {clementi,trevisan}@dsi.uniroma1.it

January 1996

Abstract

We provide new non-approximability results for the restrictions of the Min Vertex Cover problem to bounded-degree, sparse and dense graphs. We show that for a sufficiently large B, the recent 16/15 lower bound proved by Bellare et al. [5] extends with negligible loss to graphs with bounded degree B. Then, we consider sparse graphs with no dense components (i.e. everywhere sparse graphs), and we show a similar result but with a better trade-off between non-approximability and sparsity. Finally we observe that the Min Vertex Cover problem remains APX-complete when restricted to dense graph and thus recent techniques developed for several Max SNP problems restricted to "dense" instances introduced by Arora et al. [2] cannot be applied.

1 Introduction

Given the common belief that NP-hard optimization problems cannot be solved exactly in polynomial time, much research has been devoted in the past twenty years to derive efficient *approximation algorithms*, i.e. algorithms that deliver solutions whose value is guarantee to be within some multiplicative factor from the optimum.

In order to evaluate the performance guarantees of such approximation algorithms, it is important to understand how far we can go, i.e. to prove, for any approximable problem, which is the best approximation achievable in polynomial time.

Until 1991, only a very few non-approximability results were known, usually with *ad hoc* techniques that did not generalize to other problems. In 1991, Feige et al. [14] showed that results about *Probabilistic Checking of Proofs* (PCP in short - this terminology has been introduced later by Arora and Safra [4]) for NP languages imply non-approximability results for the Max Clique problem.

*Part of this work was done when both authors were visiting the Centre Universitaire d'Infomatique, University of Geneva.

†Partially supported by the Swiss National Science Foundation, grant 21-43309.95.

‡Partially supported by the Human Capital and Mobility project SCOOP - Solving Combinatorial Optimization Problems in Parallel - of the European Union.

Roughly speaking, the key ingredient of a proof checking system is a probabilistic polynomial-time oracle Turing machine (commonly called *verifier*) which, given a language L and an instance x, efficiently checks the correctness of any "proof" π (i.e. the oracle) for the "Theorem" $x \in L$. Feige et al. established a rather surprising connection between the efficiency of the verifier for the language SAT and the hardness of approximating the MAX CLIQUE problem. Such a relation is sometimes called the *FGLSS reduction* after the names of its discoverers.

Using this new approach, in a short while, a lot of increasingly strong non-approximability results were given for several problems. The verifier developed by Arora et al. [3] yielded, for several constant-factor approximable problems (namely, all the MAX SNP-hard problems [29]), a lower bound on their approximability. Lund and Yannakakis successively gave other explicit lower bounds on the approximability of the MIN NODE COLORING and the MIN SET COVER problems.

In the last three years, the search for further non-approximability results has became a growing field of computational complexity theory, and too many results have been proved to be listed here; however, we can remark that two major sources of improvement have played a key role in virtually all the recent non-approximability results.

On the one hand, there have been several improvements in the efficiency of verifiers and in the way of measuring such efficiency [6, 15, 7, 5]. The last achievement in this direction, due to Hastad [20], has been a verifier for SAT implying that MAX CLIQUE is not $n^{1/2-\epsilon}$-approximable for any $\epsilon > 0$.

On the other hand, much recent work has been devoted to improve the reductions from verifiers to optimization problems and those between problems themselves. Improved reductions yielded several recent breakthrough in approximability theory. For example, making heavy use of Raz's Parallel Repetition Theorem [30], and tightening a previous reduction given by Lund and Yannakakis [26], Feige [13] has recently shown a tight approximation lower bound for MIN SET COVER. Bellare et al. [5] use local reductions to prove hardness results for the weighted versions of MAX SAT and MAX CUT. Sorkin et al. [32] show that their local reduction for MAX SAT is the best possible and give new (and optimal) local reductions for MAX CUT and MAX DIRECTED CUT, thus giving improved non-approximability results for those problems. Crescenzi et al. [11] give a tight reduction between the weighted and unweighted versions of MAX SAT, MAX CUT, and MAX DIRECTED CUT, thus extending such non-approximability results to the unweighted case.

This paper follows the latter approach to investigate the approximability of the MIN VERTEX COVER problem with density constraints.

The MIN VERTEX COVER problem is a fundamental graph problem and was proved to be NP-hard in the original Karp's paper [22]. It is known to be NP-hard even when restricted to graphs with bounded degree [17], and this gives a clear motivation in the study of its approximability in both the general and the restricted case.

In the general case, a very simple 2-approximate algorithm has been known for twenty years [19], and no better approximation algorithm has been found until now. Slightly better approximation guarantees are achievable over bounded-degree graphs [27]. On the negative side, the MIN VERTEX COVER problem has been shown to be MAX SNP-hard even when restricted to graphs with maximum degree 3 by Papadimitriou and Yannakakis [29]. Their reduction is from MAX 3-SAT and uses explicit construction of expander graphs [16]. Combining this reduction, the non-approximability results by Bellare et al. [5] and the best known explicit construction of expanders [25], one can show that MIN VERTEX COVER is not 1.00036-approximable on bounded degree graphs.

Bellare et al. [5] give a 1.0688 lower bound for the general MIN VERTEX COVER problem by using a different technique, namely, they reduce directly from the computation of a verifier using a somehow "complementary" version of the FLGSS reduction [14]. However, their method does not apply when classes of graphs in which a fixed bound on the maximum degree or some other density constraints are considered.

Since better approximation algorithms are known to exist for the bounded degree case, and since there is such a huge gap (i.e. 1.0688 *vs* 1.00036) between the lower bound for the general case and the lower bound for the bounded-degree case, one may be tempted to conjecture that indeed the bounded-degree version is strictly easier to approximate.

We provide a new characterization of the graphs resulting from the reduction from PCP verifiers to Min Vertex Cover [5], and we show that such graphs can be seen as the union of bipartite complete graphs. We then give a construction of a particular kind of expanders (denoted as *switchers*). This technical result permits us to "sparsify" the bipartite complete graphs still preserving the connectivity property required by the reduction. This allows us to show the following hardness result for Min Vertex Cover over bounded degree graphs by directly reducing from PCP verifiers: if P $\neq$ NP then the Min Vertex Cover problem is not $(1.0688 - \epsilon)$-approximable even when restricted to graphs with maximum degree $O(1/\epsilon^3)$.

Actually, our result is fairly more general. We show that any lower bound for Min Vertex Cover proved using current techniques can be extended with negligible loss to the bounded-degree case, and we provide a trade-off between the degree of the resulting graphs and the hardness result.

It is worth noting that the best current non-approximability result for Max 3-Sat is about 1.038 [5], while we can prove the Min Vertex Cover problem to be hard to 1.068-approximate over bounded-degree graphs. It should be then clear that our result cannot be proved using a reduction from Max 3-Sat (such as Papadimitriou and Yannakakis' reduction) and, consequently, it is necessary to follow our approach of reducing directly from the verifier computations.

A better tradeoff can be achieved when a class of sparse graphs, slightly larger than that of bounded degree graphs, is considered. In particular, using a better (but probabilistic) construction of "sparse" switchers, we improve the above result for the class of *everywhere sparse graphs* i.e. graphs in which the sparsity condition is satisfied by any induced subgraph (a formal definition will be given in Section 2): If the polynomial hierarchy does not collapse, then the Min Vertex Cover problem is not $(16/15 - \epsilon)$-approximable even when restricted to everywhere $O(1/\epsilon \log 1/\epsilon)$-sparse graphs.

We have to use the hypothesis that the polynomial hierarchy does not collapse (actually, that NP $\not\subseteq$ P/poly) because we use a *non-uniform reduction.*

We also note that the reduction appeared in [5] can be slightly modified in order to show that the Min Vertex Cover problem is APX-complete even when restricted to dense graphs, and in particular to graphs with large minimum degree (thus, the "dense" restriction does not admit approximation schemes). This contrasts with the fact that several other graph problems (such as the Max Cut problem) admit an approximation scheme when restricted to dense instances [2]. The rest of the paper is organized as follows. In Section 2, we give some preliminary definitions and some previous results. Section 3 is devoted to both the probabilistic and the deterministic constructions of switchers. In Section 4, we use these graphs to derive the hardness results for Min Vertex Cover with density constraints. Finally, in Section 5, we discuss the consequences of our results for the degree of approximation of some other important optimization problems.

2 Preliminaries

Given a graph $G(V, E)$, the Min Vertex Cover problem is to find a cover C of G (i.e. a subset $C \subseteq V$ such that C contains at least an endpoint of any edge in E) whose size (i.e. $|C|$) is as small as possible. As usual, we will use n and m to denote the size of V and the size of E, respectively.

Furthermore, given a vertex $v \in V$, the degree of v will be denoted as $d(v)$. We study the complexity of approximating the MIN VERTEX COVER problem with respect to the density of the input graphs. In particular, we will make use of the following definitions.

1) *Bounded degree graphs.* A B-bounded degree graph $G(V,E)$ ($B > 0$) is a graph such that, for any $v \in V$, $d(v) \leq B$.

2) *Everywhere sparse graphs.* An everywhere k-sparse graph $G(V,E)$ is a graph such that for any subset $W \subseteq V$, the graph induced by W has a number of edges which is not greater than $k|W|$.

Given an instance x of an optimization problem and a feasible solution y of x, we let $m(x,y)$ be the *measure* (or *cost*) of the solution[1]. We also denote by $\mathsf{opt}(x)$ the measure of an optimum solution. The *performance ratio of y with respect to x* is defined as

$$R(x,y) = \max\left\{\frac{m(x,y)}{\mathsf{opt}(x)}, \frac{\mathsf{opt}(x)}{m(x,y)}\right\}.$$

Note that the performance ratio is always a number no smaller than one, and is as close to one as the solution is close to the optimum.

Definition 1 (Approximation algorithm) *Let $r > 1$ be any real; a polynomial-time algorithm is said to be* r-approximate *for an optimization problem Π if, for any instance x of Π, it returns a solution y feasible for x whose performance ratio is not greater than r.*

Definition 2 (Approximation scheme) *An algorithm is said to be an* approximation scheme *for an optimization problem Π, if, for any instance x of Π and a rational $r > 1$, it returns a solution y feasible for x whose performance ratio is not greater than r. Furthermore, for any fixed r, the running time of the algorithm is polynomial in the size of x.*

The class of optimization problems that admit an r-approximate algorithm for some $r > 1$ is denoted by APX, while the class of optimization problems that admit an approximation scheme is denoted by PTAS. It is possible to define PTAS-preserving reductions among APX problems and show natural completeness results [10, 12, 24]. In particular, the MIN VERTEX COVER problem is APX-complete even when restricted to bounded-degree graphs [29, 24].

In which follows, we summarize the main definitions from the theory of probabilistically checkable proofs and its connections with the MIN VERTEX COVER problem. Our exposition follows [5].

A *verifier* is an oracle probabilistic polynomial-time Turing machine V. During its computation, V tosses random coins, reads its input and has oracle access to a string π called *proof.* In particular, let a be the sequence of oracle answers received by V during the course of its computation on input x and random string R. If V accepts in that particular circumstance, then we say that (x,R,a) is an accepting configuration for V. Let now x be an input and π be a proof. We denote by $\mathsf{ACC}[V^{\pi}(x)]$ the probability over its random tosses that V accepts x using π as an oracle. We also denote by $\mathsf{ACC}[V(x)]$ the maximum of $\mathsf{ACC}[V^{\pi}(x)]$ over all proofs π.

We are interested in several parameters that determine the efficiency of the proof checking.

Definition 3 (PCP parameters) *Let x be a language, and let V be a verifier for L. Then we say that*

[1] In the MIN VERTEX COVER problem, instances are graphs and solutions are covers.

- V uses $r(n)$ random bits *(where* $r : \mathbf{Z}^+ \to \mathbf{Z}^+$ *is an integer function) if for any input* x *and for any proof* π*,* V *tosses at most* $r(|x|)$ *random coins;*

- V has query complexity q *(where* q *is an integer) if for any input* x*, any random string* R*, and any proof* π*,* V *reads at most* q *bits from* π*;*

- V has free bit complexity f *(where* f *is a real) if for any input* x *and any random string* R*, there are at most* 2^f *set of answers* a *such that* (x, R, a) *is an accepting configuration for* V*;*

- V has soundness s *(where* $s \in [0, 1]$ *is a real) if, for any* $x \notin L$*,* $\mathsf{ACC}[V(x)] \leq s$*;*

- V has completeness c *(where* $c \in [0, 1]$ *is a real) if, for any* $x \in L$*,* $\mathsf{ACC}[V(x)] \geq c$*.*

Definition 4 (PCP with few free bits) *Let* L *be a language, let* $0 < s < c \leq 1$ *be any constants, let* $f > 0$ *be a real,* q *be a positive integer and* $r : \mathbf{Z}^+ \to \mathbf{Z}^+$*, then we say that* $L \in \mathrm{FPCP}_{c,s}[r, f, q]$ *if a verifier* V *exists for* L *that uses* $O(r(n))$ *random bits, has query complexity* q*, free bit complexity* f*, soundness* s *and completeness* c*.*

The following theorem shows that the existence of efficient verifiers for any NP problem implies a non-approximability result for Min Vertex Cover.

Theorem 5 (Non-approximability of Min Vertex Cover **[14, 5])** *Let us assume that* $\mathrm{NP} \subseteq \mathrm{FPCP}_{c,s}[\log, f, q]$*. Then, for any* $\epsilon > 0$*, it is* NP*-hard to find* $(1 - \epsilon + (c - s)/(2^f - c))$*-approximate solutions for the* Min Vertex Cover *problem.*

Sketch of the proof. Let ϕ be an instance of the Sat problem, and let us consider the behavior of the verifier claimed in the theorem with input ϕ and a proof π. Let $\mathbf{r} = 2^{O(\log n)}$ be the total (polynomial) number of possible random sequences accessed by the verifier. For any of these sequences R, there are at most 2^f different accepting configurations (x, R, a). We say that two configurations (x, R, a) and (x, R', a') are *consistent* if a proof π exists such that a (respectively, a') is the set of answers received during the computation $V^\pi(x, R)$ (respectively, $V^\pi(x, R')$). We construct a graph G_ϕ with a node for each accepting configurations (adding dummy configurations, we make sure that there are exactly $2^f\mathbf{r}$ nodes). Then we put an edge between u and v if and only if u and v are not consistent. It is possible to show (see [14]) that there is an independent set in G_ϕ with at least k nodes if and only if there exists a proof for ϕ that makes the verifier accept at least k times over $\mathbf{r}$ (i.e. with probability $k/\mathbf{r}$). Observe that a graph G_ϕ with n nodes has an independent set with k nodes if and only if it has a vertex cover with $n - k$ nodes. It follows that if ϕ is satisfiable then there exists a vertex cover in G_ϕ with at most $\mathbf{r}(2^f - c)$ nodes; otherwise any vertex cover in G_ϕ will have at least $\mathbf{r}(2^f - s)$ nodes. Thus, any approximation factor better than $(2^f - s)/(2^f - c)$ would be sufficient to decide the satisfiability of ϕ. □

In the following, the graphs G_ϕ arising from the above described construction will be called *FGLSS graphs.*

The best current non-approximability result for Min Vertex Cover is achieved by showing that $\mathrm{NP} \subseteq \mathrm{FPCP}_{1,0.794}[\log, 2, q]$ for a certain constant q [5]. This implies that it is NP-hard to 1.068-approximate Min Vertex Cover.

3 Switchers

As described in the Introduction, our technical goal is to replace complete bipartite graphs with sparse bipartite graphs which preserve a sufficiently good "connectivity" property. In which follows we will define this particular kind of graphs and we will show its existence and how to generate them deterministically.

Definition 6 (Switcher) *Let ϵ be a positive number. A bipartite graph $G = (V_1, V_2, E)$ is an (n_1, n_2, ϵ)-switcher if the following holds:*

1. $|V_1| = n_1$, $|V_2| = n_2$;
2. *for any vertex cover C of G, either $|V_1 - C| \le \epsilon|C|$ or $|V_2 - C| \le \epsilon|C|$.*

Roughly speaking, a switcher is such that any of its vertex covers has to choose almost all the nodes in at least one component. It is worth noting that a bipartite complete graph over components of size n_1 and n_2 is an $(n_1, n_2, 0)$-switcher. As will be shown later, bipartite complete graphs are used in the proof of Theorem 5 because of their perfect switching properties. In the next section we shall show that, essentially, constant-degree switchers suffice.

In order to construct switchers, it is useful to restate property (2) in a different way. Let I be any independent set in G, let $A = V_1 \cap I$ and $B = V_2 \cap I$. Then property (2) is equivalent to asking that either

$$|A| \le \epsilon(|V_1| + |V_2| - |A| - |B|) \ ,$$

or

$$|B| \le \epsilon(|V_1| + |V_2| - |A| - |B|) \ .$$

If we consider the counterpositive version of the latter statement, we have that property (2) holds if and only if for any subset $A \subseteq V_1$ and for any subset $B \subseteq V_2$ such that

$$|A|, |B| > \epsilon(n_1 + n_2 - (|A| + |B|))$$

there is at least one edge in E joining a node in A with a node in B. This turns out to be an expansion property: switchers are indeed a generalization of *OR dispersers*.

Definition 7 (OR disperser [31]) *An (n_1, n_2, ϵ)-disperser is a bipartite graph $G = (V_1, V_2, E)$ such that $|V_1| = n_1$, $|V_2| = n_2$, and for any subsets $A \subseteq V_1$, $B \subseteq V_2$ such that $|A| \ge \epsilon n_1$ and $|B| \ge \epsilon n_2$, there is at least one edge having an endpoint in A and an endpoint in B.*

Proposition 8 *An $(n_1, n_2, \epsilon/(1+\epsilon))$-OR disperser is also an (n_1, n_2, ϵ)-switcher.*

Proof. Let $G = (V_1, V_2, E)$ be an $(n_1, n_2, \epsilon/(1+\epsilon))$-OR disperser, and let $A \subseteq V_1$ and $B \subseteq V_2$ be such that

$$\begin{aligned} |A| &> \epsilon(n_1 + n_2 - (|A| + |B|)) \\ |B| &> \epsilon(n_1 + n_2 - (|A| + |B|)) \end{aligned}$$

Since $|B| \le n_2$, it follows that $|A| \ge \epsilon(n_1 - |A|)$, that is,

$$|A| \geq \frac{\epsilon}{1+\epsilon} n_1 \ .$$

Similarly, we can show that

$$|B| \geq \frac{\epsilon}{1+\epsilon} n_2 \ .$$

Consequently, G is an (n_1, n_2, ϵ)-switcher. □

Lemma 9 (Randomized construction of switchers) *A constant $c > 0$ exists such that for any $\epsilon > 0$, for any $k > c(1/\epsilon)\log(1/\epsilon)$ and for any n_1, n_2, a $2k$-everywhere sparse (n_1, n_2, ϵ)-switcher with at most $k(n_1 + n_2)$ edges exists.*

Proof. It sufficient to show the existence of a (n_1, n_2, γ)-OR disperser where $\gamma = \epsilon/(1+\epsilon) \geq \epsilon/2$. We randomly construct a bipartite graph in the following way. Consider two vertex sets V_1 and V_2 where $|V_1| = n_1$, $|V_2| = n_2$, and $n_1 \geq n_2$. Then for any vertex u of V_1 we choose at random $\lceil (k-1)(n_1+n_2)/n_1 \rceil$ distinct elements of V_2 and we connect u to them. This construction ensures that

$$|E| \leq (k-1)(n_1+n_2) + n_1 < k(n_1+n_2) \ .$$

For any vertex pair $(v_1, v_2) \in V_1 \times V_2$, we have that

$$\mathbf{Pr}\left[(v_1, v_2) \in E\right] \geq (k-1)\frac{n_1+n_2}{n_1 n_2} \ .$$

We now provide an upper bound on the probability that the random graph $G(V_1, V_2, E)$ is not an (n_1, n_2, γ)-OR disperser. This probability will be denoted as $\mathbf{Pr}$[no-disperser]. It is not hard to prove that

$$\mathbf{Pr}[\text{no-disperser}] \ \leq \ \binom{n_1}{\gamma n_1}\binom{n_2}{\gamma n_2}\left(1 - \frac{(k-1)(n_1+n_2)}{n_1 n_2}\right)^{\gamma n_1 \gamma n_2} \ .$$

Using Stirling's approximation for the binomial coefficient and the inequality $(1-x) \leq e^{-x}$ $(x \geq 0)$, we obtain the following inequalities

$$\begin{aligned}
\mathbf{Pr}[\text{no-disperser}] \ &\leq \ \left(\frac{en_1}{\gamma n_1}\right)^{\gamma n_1}\left(\frac{en_2}{\gamma n_2}\right)^{\gamma n_2}\left(1 - \frac{(k-1)(n_1+n_2)}{n_1 n_2}\right)^{\gamma n_1 \gamma n_2} \\
&\leq \ \left(\frac{e}{\gamma}\right)^{\gamma(n_1+n_2)} e^{-\gamma^2(k-1)(n_1+n_2)} \\
&= \ e^{(1-\log\gamma)\gamma(n_1+n_2) - \gamma^2(k-1)(n_1+n_2)} \ .
\end{aligned}$$

If we introduce the condition $\mathbf{Pr}$[no-disperser] $\leq e^{-b} < 1$, where b is a fixed positive constant then we have that

$$\gamma(n_1+n_2)(1 - \log\gamma - \gamma(k-1)) \leq -b \ .$$

That is,

$$k \geq 1 + \frac{1}{\gamma}\left(\log\frac{e}{\gamma} + \frac{b}{\gamma^2(n_1+n_2)}\right) = O\left(\frac{1}{\gamma}\log\frac{1}{\gamma}\right) = O\left(\frac{1}{\epsilon}\log\frac{1}{\epsilon}\right) .$$

To show that the resulting graphs are 2k-everywhere sparse, we note that the nodes in the component V_1 have degree at most $k(n_1+n_2)/n_1 \leq 2k$. Thus, given any set $W = W_1 \cup W_2$ of nodes (where $W_i = V_i \cap W$ for $i = 1, 2$), it follows that the number of edges in the subgraph induced by W is at most $2k|W_1| < 2k|W|$. □

We shall now consider a deterministic construction that makes use of Ramanujan graphs [25, 28]. This will be used to prove non-approximability results for graphs with bounded degree under the assumption that P $\neq$ NP.

Lemma 10 (Deterministic construction of switchers) *A constant $c > 0$ exists such that, for any $\epsilon > 0$ and any n_1, n_2 such that $n_1 \geq n_2$, an (n_1, n_2, ϵ)-switcher with maximum degree $k \leq c(n_1+n_2)/n_2\epsilon^2$ exists and is constructable in polynomial time.*

Proof. It is sufficient to show how to construct an (n_1, n_2, γ)-OR disperser where $\gamma = \epsilon/(1+\epsilon) \geq \epsilon/2$. Let G be a d-regular Ramanujan expander with n nodes, where $n_1 + n_2 \leq n \leq 4(n_1+n_2)$, and $128(n_1+n_2)/n_2\gamma^2 < d \leq 512(n_1+n_2)/n_2\gamma^2$ (such a graph exists and is constructable in polynomial time [25]). The second largest eigenvalue of G is at most $2\sqrt{d-1}$. It is well known (see e.g. [28], exercise 6.27) that in a d-regular graph, if we let λ be the second largest eigenvalue of its adjacency matrix, then the number of edges connecting A and B is at least

$$d\frac{|A||B|}{n} - \lambda\sqrt{|A||B|} .$$

Let us now consider any two sets A, B such that $|A| \geq \gamma n_1$ and $|B| \geq \gamma n_2$.

We have that

$$\begin{aligned} d\frac{|A||B|}{n} - \lambda\sqrt{|A||B|} &= \sqrt{|A||B|d}\left(\frac{\sqrt{|A||B|d}}{n} - \frac{\lambda}{\sqrt{d}}\right) \\ &> \sqrt{|A||B|d}\left(\frac{\sqrt{\gamma n_1 \gamma 128(n_1+n_2)/\gamma^2}}{n} - 2\right) \\ &\geq \sqrt{|A||B|d}\left(\frac{\sqrt{64(n_1+n_2)^2}}{n} - 2\right) \\ &\geq 0 . \end{aligned}$$

Let now V_1 and V_2 be two disjoint sets of nodes of G such that $|V_1| = n_1$ and $|V_2| = n_2$. Let G' be the bipartite subgraph of G induced by the components V_1 and V_2: clearly, G' is an (n_1, n_2, γ)-OR disperser (and thus an (n_1, n_2, ϵ)-switcher) and its maximum degree is $O((n_1+n_2)/n_2\epsilon^2)$. □

4 Hardness results

Theorem 11 (Non-approximability of Min Vertex Cover-*B* **)** *Let us assume that* NP $\subseteq$ FPCP$_{c,s}$[log, f, q]. *Then for any $\epsilon > 0$ a constant $B = O(q^4/\epsilon^3)$ exists such that it is* NP*-hard to $(1-\epsilon+(c-s)/(2^f-c))$-approximate the* Min Vertex Cover *problem on graphs with maximum degree B.*

Proof. Let ϕ be an instance of SAT, and let us consider the FGLSS graph $G_\phi = (V_\phi, E_\phi)$. This graph has the following characterization. Let l be the length of the proof accessed by the verifier; for any $i = 1, \ldots, l$, let $\pi[i]$ be the i-th bit of the proof π, and let $U[i]$ (respectively, $Z[i]$) be the set of nodes of the graph corresponding to accepting configurations in which $\pi[i] = 1$ (respectively, $\pi[i] = 0$). Let also $\mathbf{u}[i] = |U[i]|$ and $\mathbf{z}[i] = |Z[i]|$. Finally, let $u_i^{(j)}$ (respectively, $z_i^{(j)}$) be the j-th element of $U[i]$ (respectively, of $Z[i]$) in lexicographic order. Given two configurations (vertices) u and v, there is an edge between u and v if and only if they are inconsistent, that is, if and only if for some i they read the same bit $\pi[i]$ and expect it to have different values. Then, we can characterize the edge set of G_ϕ as

$$E_\phi = \bigcup_{i=1}^{l} \{(u_i^{(j)}, z_i^{(k)}) : (j,k) \in K_{\mathbf{u}[i],\mathbf{z}[i]}\} \ ,$$

where, for any n_1 and n_2, K_{n_1,n_2} is the edge set of the bipartite complete graph with vertex components $\{1, \ldots, n_1\}$ and $\{1, \ldots, n_2\}$. Moreover, any node u of V_ϕ belongs to at most q sets $U[i], Z[i]$. We can thus see E_ϕ as the union of bipartite complete graphs, i.e. graphs with the best possible switching properties. We shall now show that indeed constant degree switchers are sufficient. Without loss of generality, we assume that for any $i = 1, \ldots, l$, $\mathbf{u}[i] \geq \mathbf{z}[i]$ (otherwise, we can invert the value of the i-th bit of the proof in any configuration and then swap the values of $\mathbf{u}[i]$ and $\mathbf{z}[i]$). Let γ be a constant to be fixed later such that $1/\gamma = O(q/\epsilon)$. Let I be the set of bits i such that $\mathbf{z}[i] \geq \gamma(\mathbf{z}[i] + \mathbf{u}[i])$. For any n_1 and for any n_2, let S_{n_1,n_2} be the set of edges of an (n_1, n_2, γ)-switcher (we assume that the vertex sets are $\{1, \ldots, n_1\}$ and $\{1, \ldots, n_2\}$). We define a graph $G'_\phi = (V_\phi, E'_\phi)$ with the same vertex set of G_ϕ and with edge set

$$E'_\phi = \bigcup_{i \in I} \{(u_i^{(j)}, z_i^{(k)}) : (j,k) \in S_{\mathbf{u}[i],\mathbf{z}[i]}\} \ .$$

Lemma 10 implies that, for any $i \in I$, we can construct a $(\mathbf{u}[i], \mathbf{z}[i], \gamma)$-switcher with degree bounded by

$$O((\mathbf{u}[i] + \mathbf{z}[i])/\mathbf{z}[i]\gamma^2) = O(1/\gamma^3) \ .$$

Since any node belongs to at most q sets $U[i]$, $Z[i]$, and we assumed that $\gamma = O(\epsilon/q)$, it follows that G'_ϕ has degree bounded by $O(q^4/\epsilon^3)$.

We shall now show how to convert any vertex cover for G'_ϕ into a "slightly larger" vertex cover for G_ϕ. We claim that from any vertex cover C' in G'_ϕ we can recover a vertex cover C in G_ϕ such that $|C| \leq |C'|(1 + q\gamma) + q\gamma n$. Indeed, let $C'' = \bigcup_{i \notin I} Z[i]$: from the definition of I, it follows that

$$|C''| \leq \sum_{i \notin I} \gamma(\mathbf{z}[i] + \mathbf{u}[i]) \leq q\gamma n \ .$$

Moreover, for any $i \in I$, let $C'[i] = C' \cap (U[i] \cup Z[i])$ be the set of nodes of C' that are used to cover the configurations where the i-th bit has been read. By adding at most $\gamma C'[i]$ nodes, we have a cover that comprises either all nodes of $U[i]$ or all nodes of $Z[i]$. Let $C[i]$ be this new set. Clearly, $C = C'' \cup \bigcup_{i=1}^{l} C[i]$ is a vertex cover for G_ϕ, and we have that

$$|C| - |C'| \leq q\gamma n + \sum_i |C[i]| - |C'[i]| \leq q\gamma n + q\gamma |C'|$$

where $n = \mathbf{r}2^f$. If ϕ is not satisfiable, then

$$\mathsf{opt}(G'_\phi) \geq \frac{1}{1+q\gamma}\mathsf{opt}(G_\phi) - \gamma 2^f q\mathbf{r} \geq \mathbf{r}(2^f - s)\left(\frac{1}{1+q\gamma} - q\gamma\frac{2^f}{2^f - s}\right) .$$

Furthermore, G'_ϕ is an edge-subgraph of G_ϕ, thus any vertex cover for G_ϕ is also a vertex cover for G'_ϕ. It follows that if ϕ is satisfiable, then

$$\mathsf{opt}(G'_\phi) \leq \mathsf{opt}(G_\phi) \leq \mathbf{r}(2^f - c) .$$

By letting γ be sufficiently small (but such that $1/\gamma = O(q/\epsilon)$), the theorem follows. □

Using the same technique applied in the proof of Theorem 11 we can prove the following result. The main difference with respect to the proof of Theorem 11 is that this time we use sparse switchers whose existence is guaranteed by Lemma 9.

Theorem 12 *Let us assume that* $\mathrm{NP} \subseteq \mathrm{FPCP}_{c,s}[\log, f, q]$. *Then for any* $\epsilon > 0$ *a constant* $k = O((q^2/\epsilon)\log q/\epsilon)$ *exists such that the* Min Vertex Cover *problem restricted to everywhere* k*-sparse graphs is not* $(1 - \epsilon + (c-s)/(2^f - c))$*-approximable unless* $\mathrm{NP} \subseteq \mathrm{P/poly}$.

Proof. The proof is very similar to that of Theorem 11, the only difference being the use of better switchers whose existence is proved in Lemma 9. In particular, for any i, we use an everywhere k/q-sparse $(\mathbf{u}[i], \mathbf{z}[i], \gamma)$-switcher which exists provided that $k/q = O((1/\gamma)\log 1/\gamma)$. It follows that G'_ϕ is everywhere k-sparse where $k = O(q(1/\gamma)\log 1/\gamma)$, that is, $k = O((q^2/\epsilon)\log q/\epsilon)$. We can now repeat the same analysis as in the proof of Theorem 11. Since we are not able to explicitly construct such switchers, we assume that the reduction receives them as *polynomial size advice*. Thus, instead of a polynomial-time reduction we use a P/poly reduction, and this allows us to prove hardness results under the hypothesis that $\mathrm{NP} \not\subseteq \mathrm{P/poly}$ (recall that $\mathrm{NP} \subseteq \mathrm{P/poly}$ mplies the collapse of the polynomial hierarchy [23]). □

Our techniques also yield results regarding the approximability of the Min Vertex Cover problem on graphs having a non-linear number of edges.

An interesting consequence of Theorem 11 is the fact that any lower bound proved with the PCP technique for the Min Vertex Cover problem on general graphs extends *without any loss* to graphs with maximum degree bounded by *any* (thus even very slow) increasing function.

Corollary 13 (of Theorem 11) *Let* $h : \mathbf{Z}^+ \to \mathbf{Z}^+$ *be a computable function such that* $\lim_n h(n) = \infty$, *let* $\mathrm{NP} \subseteq \mathrm{FPCP}_{c,s}[\log, f, q]$. *Then for any* $\epsilon > 0$ *the* Min Vertex Cover *problem restricted to graphs with maximum degree* $h(n)$ *is* NP*-hard to approximate within* $1 - \epsilon + (c-s)/(2^f - c)$.

Proof. For any $\epsilon > 0$, Theorem 11 implies that a constant B_ϵ exists such that the Min Vertex Cover problem is NP-hard to approximate within $1 - \epsilon + (c-s)/(2^f - c)$ when restricted to graph with maximum degree B_ϵ. It is clear that the Min Vertex Cover problem restricted to graphs with maximum degree B_ϵ is reducible to the Min Vertex Cover problem restricted to graphs with maximum degree $h(n)$. Indeed, the two problems just differ on a finite number of instances. □

The restriction to dense instances (i.e. graphs with $\Omega(n^2)$ edges) of optimization graph problems often admits an efficient approximation scheme [2] (see also [1] on parallel approximation) even if the general problem is hard to approximate. We note, however, that this is not the case of Min Vertex Cover.

Theorem 14 *The* MIN VERTEX COVER *problem restricted to dense graphs is* APX*-complete. In particular, for any $\epsilon > 0$ there exists a constant $r > 1$ (depending on ϵ) such that it is* NP*-hard to r-approximate the* MIN VERTEX COVER *problem restricted to graphs such that any node has degree at least $\epsilon|V|$.*

Proof. For any $\epsilon > 0$, let $\alpha = \epsilon/(1-\epsilon)$. Consider the graph G'_ϕ obtained by adding a clique with $\alpha|V_\phi|$ nodes to the FGLSS graph G_ϕ and then connecting any node of the clique to any node of $G_\phi = (V_\phi, E_\phi)$. Let $n = (\alpha+1)|V_\phi|$ be the number of vertices in G'_ϕ, It is easy to see that G'_ϕ has minimum degree at least ϵn. Moreover,

$$\mathsf{opt}(G_\phi) + \alpha|V_\phi| - 1 \leq \mathsf{opt}(G'_\phi) \leq \mathsf{opt}(G_\phi) + \alpha|V_\phi| \ .$$

The results of [5] imply that a constant $r > 1$ exists such that it is NP-hard to r-approximate MIN VERTEX COVER on graphs with minimum degree at least ϵn. Furthermore, Theorem 5 in [24] implies that the MIN VERTEX COVER problem is APX-complete with respect to the AP-reducibility even when restricted to FGLSS graphs. It is easy to see that the above described reduction from FGLSS graphs to dense graphs is approximation-preserving, and, in particular, is an L-reduction [29], and thus also an E-reduction [24] and an AP-reduction (see [9]). The APX-completeness of the MIN VERTEX COVER problem restricted to dense graphs follows. □

Note that inserting a large clique provides a non-approximability result only because FGLSS graphs are such that the minimum vertex cover always has $\Omega(n)$ nodes, which is not true in general. In particular, the same technique does not provide an approximation-preserving reduction from the general MIN VERTEX COVER problem to its restriction over dense graphs.

5 Conclusions

In this paper, we have provided new hardness results on the approximation of MIN VERTEX COVER when some density constraints on the input graphs are considered. A further motivation in determining whether or not the presence of a bound on the number of edges (or on the maximum degree) yields a more "tractable" restriction of the general problem is due to the fact that the MIN VERTEX COVER problem restricted to bounded maximum-degree graphs or to sparse ones (observe that we have considered a "strong" concept of sparse graphs) has been used as the starting problem in several reductions to other important problems such as the restriction of the MIN STEINER TREE problem to metric spaces [8] and the LONGEST COMMON SUBSEQUENCE problem over alphabet with small size [21] (a problem related to DNA sequencing). For example, the reduction from MIN VERTEX COVER to MIN STEINER TREE shown in [8] implies a non-approximability result for MIN STEINER TREE that depends on the non-approximability ratio that one can prove for vertex cover on sparse graphs and on the sparsity of such graphs (and the additional condition that the sparse graphs are such that the minimum cover is guaranteed to be a constant fraction of the number of nodes). We computed the non-approximability result for MIN STEINER TREE that arises from [29, 8, 25, 5], and it is about $1 + 1/5600$. More generally, there is a linear relation between the hardness ratio that one can prove for the MAX 3-SAT problem and the consequent hardness ratio implied for the MIN STEINER TREE problem. On the other hand, our present results, combined with the best currently available verifier [5], give a worse hardness ratio for the MIN STEINER TREE problem, but the relation between the efficiency of the verifier and the hardness for MIN STEINER TREE is superlinear, and thus better verifiers will imply a larger improvement for the hardness implied by our reduction than for that implied by Papadimitriou and Yannakakis' reduction. More formally, applying Theorem 12 with $\epsilon = (c-s)/2^{f+1}$ it follows that if NP $\subseteq$ FPCP$_{c,s}[\log n, f, q]$, then it is hard to approximate MIN STEINER TREE to within a factor

$$1 + \frac{c-s}{O\left(2^f \cdot \frac{q^2 \cdot 2^{f+1}}{c-s} \cdot \log \frac{q \cdot 2^{f+1}}{c-s}\right)} ,$$

where the constant hidden in the O notation is about 4. Instead, using Papadimitriou and Yannakakis' reduction, we have that if NP $\subseteq$ FPCP$_{c,s}[\log n, f, q]$, then it is hard to approximate MIN STEINER TREE to within a factor

$$1 + \frac{c-s}{200 \cdot 2^{q-1}} .$$

Observe also that our results are related to the free-bit complexity of the verifier, and improvements on this query complexity measure do not imply any improvement for Papadimitriou and Yannakakis' reduction.

Acknowledgements

We wish to thank Pierluigi Crescenzi, José Rolim and Madhu Sudan for several helpful discussions.

References

[1] A. Andreev, A. Clementi, and J. Rolim. Constructing the highest degree subgraph for dense graphs is in NCAS. To appear on *Theoretical Computer Science*, 1995.

[2] S. Arora, D. Karger, and M. Karpinski. Polynomial time approximation schemes for dense instances of graph problems. In *Proceedings of the 27th ACM Symposium on Theory of Computing*, pages 284–293, 1995.

[3] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and hardness of approximation problems. In *Proceedings of the 33rd IEEE Symposium on Foundations of Computer Science*, pages 14–23, 1992.

[4] S. Arora and S. Safra. Probabilistic checking of proofs; a new characterization of NP. In *Proceedings of the 33rd IEEE Symposium on Foundations of Computer Science*, pages 2–13, 1992.

[5] M. Bellare, O. Goldreich, and M. Sudan. Free bits, PCP's and non-approximability – towards tight results (3rd version). Technical Report TR95-24, Electronic Colloquium on Computational Complexity, 1995. Extended abstract in *Proc. of FOCS'95.*

[6] M Bellare, S. Goldwasser, C. Lund, and A. Russell. Efficient probabilistically checkable proofs and applications to approximation. In *Proceedings of the 25th ACM Symposium on Theory of Computing*, pages 294–304, 1993. See also the errata sheet in *Proc of STOC'94*.

[7] M. Bellare and M. Sudan. Improved non-approximability results. In *Proceedings of the 26th ACM Symposium on Theory of Computing*, pages 184–193, 1994.

[8] M. Bern and P. Plassmann. The steiner tree problem with edge lengths 1 and 2. *Information Processing Letters*, 32:171–176, 1989.

[9] P. Crescenzi, V. Kann, R. Silvestri, and L. Trevisan. Structure in approximation classes. In *Proceedings of the 1st Combinatorics and Computing Conference*, pages 539–548. LNCS 959, Springer Verlag, 1995.

[10] P. Crescenzi and A. Panconesi. Completeness in approximation classes. *Information and Computation*, 93:241–262, 1991. Preliminary version in *Proc. of FCT'89.*

[11] P. Crescenzi, R. Silvestri, and L. Trevisan. To weight or not to weight: Where is the question? Manuscript, 1996.

[12] P. Crescenzi and L. Trevisan. On approximation scheme preserving reducibility and its applications. In *Proceedings of 14th Conference on Foundations of Software Technology and Theoretical Computer Science*, pages 330–341. LNCS 880, Springer Verlag, 1994.

[13] U. Feige. A threshold of $\ln n$ for approximating set cover. In *Proceedings of the 28th ACM Symposium on Theory of Computing*, 1996. To appear.

[14] U. Feige, S. Goldwasser, L. Lovasz, S. Safra, and M. Szegedy. Approximating clique is almost NP-complete. In *Proceedings of the 32nd IEEE Symposium on Foundations of Computer Science*, pages 2–12, 1991.

[15] U. Feige and J. Kilian. Two prover protocols - low error at affordable rates. In *Proceedings of the 26th ACM Symposium on Theory of Computing*, pages 172–183, 1994.

[16] O. Gabber and Z. Galil. Explicit construction of linear sized superconcentrators. *Journal of Computer and System Sciences*, 22:407–425, 1981.

[17] M. R. Garey, D. S. Johnson, and L. Stockmeyer. Some simplified NP-complete graph problems. *Theoretical Computer Science*, 1:237–267, 1976.

[18] M.R. Garey and D.S. Johnson. *Computers and Intractability: a Guide to the Theory of NP-Completeness.* Freeman, 1979.

[19] F. Gavril. Manuscript cited in [18], 1974.

[20] J. Håstad. Testing of the long code and hardness for clique. In *Proceedings of the 28th ACM Symposium on Theory of Computing*, 1996. To appear.

[21] T. Jiang and M. Li. On the approximation of shortest common supersequences and longest common subsequences. In *Proceedings of 21th International Colloquium on Automata, Languages and Programming*, pages 191–202. Lecture Notes in Computer Science 820, 1994.

[22] R.M. Karp. Reducibility among combinatorial problems. In R.E. Miller and J.W. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972.

[23] R.M. Karp and R.J. Lipton. Some connections between nonuniform and uniform complexity classes. In *Proceedings of the 12th ACM Symposium on Theory of Computing*, pages 302–309, 1980.

[24] S. Khanna, R. Motwani, M. Sudan, and U. Vazirani. On syntactic versus computational views of approximability. In *Proceedings of the 35th IEEE Symposium on Foundations of Computer Science*, pages 819–830, 1994.

[25] A. Lubotzky, R. Phillips, and P. Sarnak. Ramanujam graphs. *Combinatorica*, 8:261–277, 1988.

[26] C. Lund and M. Yannakakis. On the hardness of approximating minimization problems. *Journal of the ACM*, 41:960–981, 1994. Preliminary version in *Proc. of STOC'93*.

[27] B. Monien and E. Speckenmeyer. Some further approximation algorithms for the vertex cover problem. In *Proceedings of CAAP83*, pages 341–349. LNCS 159, Springer Verlag, 1983.

[28] R. Motwani and P. Raghavan. *Randomized Algorithms.* Cambridge University Press, 1995.

[29] C. H. Papadimitriou and M. Yannakakis. Optimization, approximation, and complexity classes. *Journal of Computer and System Sciences*, 43:425–440, 1991. Preliminary version in *Proc. of STOC'88*.

[30] R. Raz. A parallel repetition theorem. In *Proceedings of the 27th ACM Symposium on Theory of Computing*, pages 447–456, 1995.

[31] M. Sipser. Expanders, randomness or time vs. space. In *Proceedings of the 1st Conference on Structure in Complexity Theory.* Springer Verlag, 1986.

[32] G. Sorkin, M. Sudan, L. Trevisan, and D.P. Williamson. Gadgets, approximation and linear programming. In preparation, 1996.