Theoretical Computer Science 237 (2000) 123–134
Theoretical Computer Science
www.elsevier.com/locate/tcs

Some APX-completeness results for cubic graphs

Paola Alimonti [a,*,1], Viggo Kann [b,2]

[a] *Dipartimento di Informatica e Sistemistica, University of Rome "la Sapienza", Italy*
[b] *Numerical Analysis and Computing Science, Royal Institute of Technology, Stockholm, Sweden*

Received April 1997; received in revised form February 1998
Communicated by G. Ausiello

Abstract

Four fundamental graph problems, Minimum vertex cover, Maximum independent set, Minimum dominating set and Maximum cut, are shown to be APX-complete even for cubic graphs. Therefore, unless P = NP, these problems do not admit any polynomial time approximation scheme on input graphs of degree bounded by three. © 2000 Elsevier Science B.V. All rights reserved.

Keywords: Computational complexity; NP-hard optimization problems; approximation; APX-completeness; cubic graphs

1. Introduction

Among combinatorial optimization problems that are computationally hard to solve, NP-hard optimization problems on graphs have a great relevance both from the theoretical and practical point of view.

Despite the apparent simplicity of cubic and at-most cubic graphs, several NP-hard graph problems remain NP-hard even if restricted to these classes of graphs, but become polynomial time solvable for graphs of degree 2 [12, 14].

Since it is widely conjectured that NP-hard problems cannot be efficiently solved, one has to restrict oneself to compute approximate solutions. Therefore, it would be desirable to identify if and how much boundedness of the graph degree is helpful in approximation.

* Corresponding author.
E-mail address: alimon@dis.uniroma1.it (P. Alimonti).
[1] Supported by: the CEE project ALCOM-IT ESPRIT LTR, project no. 20244, "Algorithms and Complexity in Information Technology"; the Italian Project "Algoritmi, Modelli di Calcolo e Strutture Informative", Ministero dell'Università e della Ricerca Scientifica e Tecnologica. Parts of this work were done when the author was visiting the Royal Institute of Technology.
[2] Supported by TFR.

PII: S0304-3975(98)00158-3

It is well known that the variation of NP-hard graph problems in which the degree of the graph is bounded by a constant often allows to achieve different results with respect to the approximation properties. Namely, problems that for general graphs are not approximable within a constant ratio (e.g. Maximum independent set, Minimum dominating set and Minimum independent dominating set) have been shown to be in APX (i.e. approximable within *some* constant) for bounded degree graphs. For some NP-hard optimization problems that are approximable for general graphs (e.g. Minimum vertex cover) better approximation ratios have been achieved for graphs of low degree [5–7, 15, 16, 19, 20, 22, 24].

Nevertheless, many graph problems are APX-hard even if the degree of the graph is bounded by some constant, and therefore they can be approximated within some constant factor of the optimum, but cannot be approximated within *any* constant (PTAS) [18–20, 24].

Some problems are known to be APX-hard even for cubic or at-most-cubic graphs (e.g. Maximum three-dimensional matching and Maximum independent dominating set [18, 19]). For several other graph problems it is just known that they are APX-hard for graphs of some bounded degree greater than 3 [24].

In this work we show APX-hardness results for several optimization problems on cubic or at-most-cubic graphs, namely for Minimum vertex cover (MIN VERTEX COVER), Maximum independent set (MAX IND SET), Minimum dominating set (MIN DOM SET), and Maximum cut (MAX CUT).

Surprisingly, simple reductions are used for three out of four of our results, but for showing the APX-completeness of MAX CUT on cubic graphs we need a quite complicated structure consisting of a chain of expander graphs. Expander graphs have been used in different ways in approximation preserving reductions [2, 3, 11, 24], and seems to be very useful. For a description of expander graphs and an algorithm constructing expander graphs we refer to [1].

The remainder of the paper is organized as follows. In Section 2, we state basic definitions and notations. In Section 3, we show the APX-completeness of MIN VERTEX COVER, MAX IND SET and MIN DOM SET on cubic graphs. In Section 4, we prove the APX-completeness of MAX CUT on cubic graphs.

2. Definitions

The degree of a vertex v in an undirected graph is denoted $d(v)$. The *degree of a graph* is the maximum degree of any vertex in the graph. A graph is *cubic* if every vertex has degree 3, and *at-most-cubic* if every vertex has degree at most 3.

Several notions of approximation scheme preserving reducibilities have been proposed with the aim of establishing hardness and completeness results in APX and of deriving proofs of intractability of arbitrary approximation for NP-hard optimization problems [10, 23, 24].

Among them, the AP-reducibility introduced in [9] is perhaps the strictest one appearing in the literature that allows to obtain natural APX-completeness results. Namely, using AP-reducibility one can show that a problem F does not admit polynomial time approximation scheme by proving that F is APX-hard w.r.t. AP-reducibility, i.e., every approximable problem can be AP-reduced to F. If F is both approximable within c for some c and APX-hard w.r.t. AP-reducibility then it is APX-complete w.r.t. AP-reducibility.

A more restricted kind of reducibility that has been very popular, since it has been introduced in [24], is the L-reducibility.

Given two NP optimization problems F and G and a polynomial time transformation f from instances of F to instances of G, we say that f is an *L-reduction* if there are positive constants α and β such that for every instance x of F

1. $\mathrm{opt}_G(f(x)) \leqslant \alpha \cdot \mathrm{opt}_F(x)$,
2. for every feasible solution y of $f(x)$ with objective value $m_G(f(x), y) = c_2$ we can in polynomial time find a solution y' of x with $m_F(x, y') = c_1$ such that $|\mathrm{opt}_F(x) - c_1| \leqslant \beta\, |\mathrm{opt}_G(f(x)) - c_2|$.

If a problem F is APX-complete w.r.t. L-reducibility then it is also APX-complete w.r.t. AP-reducibility. This enables us to prove APX-completeness by means of the easier-to-use L-reducibility, which therefore will be used in the paper.

2.1. Definitions of problems

Here the problems considered in the paper are defined, and known approximation results are given. A much larger list of optimization problems and their approximability can be found in [8, 4].

MAX CUT–B
Instance: Graph $G = (V, E)$ of degree bounded by B.
Solution: A partition of V into two parts: a red part P_R and a green part P_G.
Measure: Cardinality of the set of edges that are cut, i.e., edges with one end point in P_R and one end point in P_G.
Approximability: Approximable within 1.139 [13] for every B, and APX-complete for some (large) constant B [24].

MAX IND SET–B
Instance: Graph $G = (V, E)$ of degree bounded by B.
Solution: An independent set for G, i.e., a subset $V' \subseteq V$ such that no two vertices in V' are joined by an edge in E.
Measure: Cardinality of the independent set, i.e., $|V'|$.
Approximability: Approximable within 1.201 for $B = 3$ [6, 7], and APX-complete for $B \geqslant 4$ [24].

MIN DOM SET–B
Instance: Graph $G = (V, E)$ of degree bounded by B.

Solution: A dominating set for G, i.e., a subset $V' \subseteq V$ such that for all $u \in V - V'$ there is a $v \in V'$ for which $(u,v) \in E$.
Measure: Cardinality of the dominating set, i.e., $|V'|$.
Approximability: Approximable within 1.75 for $B = 3$ [16], and APX-complete for $B \geqslant 8$ [24].

MIN VERTEX COVER–B
Instance: Graph $G = (V,E)$ of degree bounded by B.
Solution: A vertex cover for G, i.e., a subset $V' \subseteq V$ such that for all $(u,v) \in E$ at least one of u and v is included in V'.
Measure: Cardinality of the vertex cover, i.e., $|V'|$.
Approximability: Approximable within 1.167 [6] for $B = 3$, and APX-complete for $B \geqslant 4$ [24].

MAX 3-SAT–B
Instance: Set of variables X, set of disjunctive clauses C over the variables X, where each clause consists of at most three variables, and each variable occurs in at most B clauses.
Solution: Truth assignment of X.
Measure: Cardinality of the set of clauses from C that are satisfied by the truth assignment.
Approximability: Approximable within 1.143 for every B [21], and APX-complete for $B \geqslant 3$ [4, 25]. This hardness result is used to show the hardness results for all listed problems above.

3. APX-completeness of some problems on cubic graphs

It is known that MIN VERTEX COVER$-B$, MAX IND SET$-B$ and MIN DOM SET–B are included in APX and APX-complete for some bounded degree B (see the problem list above). In the following, we will show that these problems remain APX-complete even if the degree of the graphs is bounded by 3.

Theorem 3.1. MIN VERTEX COVER-3 *is* APX-*complete.*

Proof. Since MIN VERTEX COVER$-3 \in$ APX we just have to show that it is APX-hard. Let f be the following L-reduction from MIN VERTEX COVER-4 to MIN VERTEX COVER-3.

Given a graph $G = (V,E)$ of bounded degree 4 construct an at-most-cubic graph $G' = (V',E')$ by splitting every degree 4 vertex as shown in Fig. 1.

It is easy to see that from every vertex cover $C \subseteq V$ of G we can construct a vertex cover $C' \subseteq V'$ of $G' = f(G)$ of size exactly $|C| + s$ where s is the number of vertices of degree 4 in G. In C' we include every vertex in C that has degree smaller than 4, and

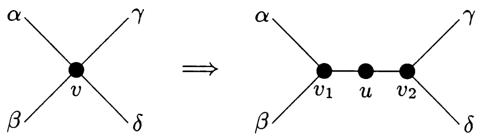

Fig. 1. The transformation of a degree 4 vertex used both in the reduction from MIN VERTEX COVER−4 to MIN VERTEX COVER−3 and the reduction from MAX IND SET−4 to MAX IND SET−3.

for each vertex $v \in V$ of degree 4 we do as follows. If $v \in C$ then $v_1, v_2 \in C'$, if $v \notin C$ then $u \in C'$. Since G has bounded degree 4 we have $4|C| \geqslant \sum_{v \in C} d(v) \geqslant |E| \geqslant |V| \geqslant s$. We see that $|C'| = |C| + s \leqslant 5 \cdot |C|$. Hence, the first property of an L-reduction is satisfied with $\alpha = 5$.

Conversely, given a vertex cover $C' \subseteq V'$ of $G' = f(G)$ we transform it back to a vertex cover $C \subseteq V$ of G in the following manner. Note that for each triple of vertices $v_1, v_2, u \in V'$ coming from a vertex $v \in V$ of degree 4: either $u \in C'$, or at least two vertices belong to C'. Include in C any vertex of degree less than 4 that belongs to C' and any vertex $v \in V$ of degree 4 such that at least two of the vertices v_1, v_2 and u belong to C'. Observe that C is a vertex cover of G and $|C| \leqslant |C'| - s$. Together with the observations from the preceding paragraph this shows that f is an L-reduction with $\beta = 1$. □

Theorem 3.2. MAX IND SET−3 *is* APX-*complete.*

This result was recently and independently proved by Berman and Fujito [6], and Halldórsson and Yoshihara [17] using complex reductions from MAX 3-SAT−B. We can give a much simpler proof of this result using the same reduction as in the proof of Theorem 3.1. Analogously, the reductions by Berman et al. could be used to show that MIN VERTEX COVER−3 is APX-complete.

Proof (*Outline*). Since the complement of any vertex cover is an independent set the same transformation as above can be used to prove the theorem. This is an L-reduction from MAX IND SET−4 to MAX IND SET−3 with $\alpha = 5$ and $\beta = 1$. □

Theorem 3.3. MIN DOM SET−3 *is* APX-*complete.*

Proof. We will use the fact that L-reductions compose [24], and first give an L-reduction f_1 from MIN VERTEX COVER−3 to MIN DOM SET−6 and then an L-reduction f_2 from MIN DOM SET−6 to MIN DOM SET−3. Since isolated vertices are trivially included in any dominating set we can, without loss of generality, assume that the graph contains no isolated vertices in the hardness proof.

Given an at-most-cubic graph $G = (V, E)$ construct a graph $G' = (V', E')$ of bounded degree 6 in the following way. For each edge (u, v) in the former graph insert an extra vertex w and edges (u, w), (v, w), see Fig. 2.

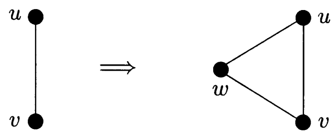

Fig. 2. The transformation of an edge in the reduction from MIN VERTEX COVER−3 to MIN DOM SET−6.

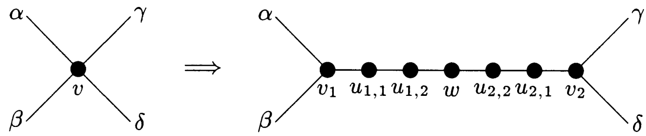

Fig. 3. The transformation of a degree 4 vertex in the reduction from MIN DOM SET−6 to MIN DOM SET−3.

It is easy to see that every dominating set $D \subseteq V'$ of $G' = f_1(G)$ can be transformed into an equally good or better vertex cover $C \subseteq V$ of G by including in C the following vertices. For each vertex $v \in D$ such that $v \in V$, include v in C. For each vertex $w \in D$ such that $w \notin V$, choose a vertex v such that $(v, w) \in E'$ and include v in C.

Now, consider a vertex cover $C \subseteq V$ of G. We can construct a dominating set $D \subseteq V'$ of G' of the same size by including in D exactly the same vertices. Thus, f_1 is an L-reduction with $\alpha = \beta = 1$.

Now, we describe the L-reduction f_2 from MIN DOM SET−6 to MIN DOM SET−3. Given a graph $G = (V, E)$ of bounded degree 6 construct an at-most-cubic graph $G' = (V', E')$ in the following manner. Each vertex $v \in V$ of degree 4 is split and transformed as shown in Fig. 3.

For each vertex $v \in V$ of degree 5 or 6 we do in a similar way except for the fact that we split v into three vertices, instead of two and extend the previous construction with a third leg from the center vertex w. More precisely, in addition to the construction in Fig. 3 we construct vertices v_3, $u_{3,1}$, $u_{3,2}$, and edges $(v_3, u_{3,1})$, $(u_{3,1}, u_{3,2})$, $(u_{3,2}, w)$. v_3 is then connected to the 5th (and, if $d(v) = 6$, 6th) neighbour of v. Now, every vertex has degree at most 3.

It is easy to see that any dominating set $D' \subseteq V'$ of $G' = f_2(G)$ can be transformed back to a dominating set $D \subseteq V$ of G as follows. For each vertex $v \in V$ of degree less than 4 we include v in D iff $v \in D'$. Otherwise, let $V(v)$ be the set of vertices that v was transformed into, and let $K(v) = |V(v) \cap D'|$. If $d(v) = 4$ we include v in D iff $K(v) \geqslant 3$, and if $d(v) \geqslant 5$ we include v in D iff $K(v) \geqslant 4$. It is clear that D is a dominating set of size $|D| \leqslant |D'| - 2 \cdot s_1 - 3 \cdot s_2$, where s_1 and s_2 are the number of vertices of degree 4 and greater than 4 in V, respectively.

Finally, given a dominating set $D \subseteq V$ of G we can construct a dominating set $D' \subseteq V'$ of $G' = f_2(G)$ such that $|D'| = |D| + 2 \cdot s_1 + 3 \cdot s_2$. Since G has bounded degree 6, we have $|D| \geqslant |V|/7$. Therefore $|D'| \leqslant |D| + 3 \cdot (s_1 + s_2) \leqslant 22 \cdot |D|$.

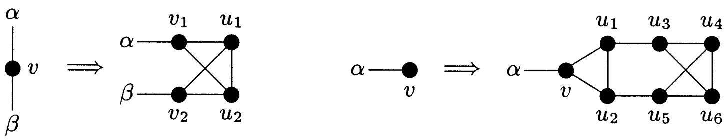

Fig. 4. The transformation of vertices of degree 2 and 1 in the reduction from at-most-cubic graphs to cubic graphs.

Thus, $\text{opt}(f_2(G)) \leqslant 22 \cdot \text{opt}(G)$ and we have shown that f_2 is an L-reduction with $\alpha = 22$ and $\beta = 1$. □

The above results are still valid for cubic graphs. We show this for MIN VERTEX COVER−3.

We can assume that the graph does not contain any isolated vertices (since they would not contribute to the solution in any case). Transform each vertex v of degree two or one as shown in Fig. 4.

From every solution of size c in the at-most-cubic graph we can construct a solution in the cubic graph of size exactly $4s_1 + 2s_2 + c$, where s_1 and s_2 are the number of vertices of degree 1 and 2 in the at-most-cubic graph, respectively.

4. APX-completeness of MAX CUT on cubic graphs

In this section we will show that MAX CUT−3 is APX-complete in two steps. First, we will show that MAX CUT is APX-hard for multigraphs of degree 6, and then for simple graphs of degree 3.

Theorem 4.1. MAX CUT−6 *for multigraphs is* APX-*complete.*

Proof. We will construct an L-reduction from MAX 3-SAT−3 to MAX CUT−6 for multigraphs. Suppose we are given an instance of MAX 3-SAT−3 with n variables and m clauses. Without loss of generality, we can assume that every variable occurs positively in at least one clause and negatively in at least one clause, and that every clause consists of two or three literals. MAX 3-SAT−3 with instances satisfying these restrictions is known to be APX-complete [4].

Construct a multigraph with a vertex set consisting of two vertices named x_i and $\overline{x}_i$ (the *variable vertices*) for each variable, and four vertices named y_j, $\overline{y}_j$, b_{2j-1} and b_{2j} for each clause. For each clause we construct edges as shown in Fig. 5. We also, for every variable x_i, include two parallel edges between the vertices x_i and $\overline{x}_i$. The degree of the graph is 6.

Consider a solution where all the b_i vertices in the graph are placed in the same part, say the red part.

If we look at the subgraphs in Fig. 5 we can see (in both cases) that if all l_k vertices are in the red part, then at most 4 edges can be cut, but if at least one of the

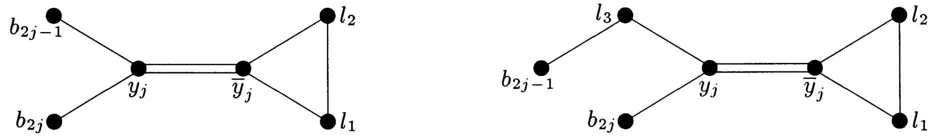

Fig. 5. The constructed edges from clauses $l_1 \vee l_2$ and $l_1 \vee l_2 \vee l_3$, respectively.

l_k vertices is in the green part, then it is always possible to choose parts for y_j and $\overline{y}_j$ so that 6 edges are cut.

If x_i and $\overline{x}_i$ are placed in the same part we can move one of them to the other part without decreasing the number of cut edges. This is because at least one of x_i and $\overline{x}_i$ occurs just in one clause, and has therefore just two edges incident to it except the two edges incident on it connecting x_i and $\overline{x}_i$.

Now, we have a correspondence between the values of the variables and the partition of the variable vertices – if the variable x_i is true then the vertex x_i is green and $\overline{x}_i$ is red, and if the variable x_i is false then the vertex x_i is red and $\overline{x}_i$ is green. Thus, the number of cut edges will be $2n + 4m + 2s$ where s is the number of satisfied clauses in the corresponding MAX 3-SAT−3 problem instance. This shows that the second property of the L-reduction is satisfied with $\beta = \frac{1}{2}$.

Recall that the above reasoning is valid only if all the b_j vertices are in the same part. In order to obtain this we construct a bipartite cubic expander between the b_j vertices and an equivalently large set of new vertices, called $c_{1,j}$. We then construct a chain of bipartite cubic expanders between $\{c_{i,j}\}$ and $\{c_{i+1,j}\}$ for $1 \leqslant i < k$, where k is a constant to be decided later. We thus have $k + 1$ layers of vertices that are connected by expanders. The degree of each vertex is at most 6. Let N be the number of b_j vertices (which means that N is the number of vertices in any of the $k + 1$ layers).

Ajtai has shown that cubic bipartite expander graphs of size N can be constructed in polynomial time in N [1]. Such an expander $G = (A \cup B, E)$ has the property that for every subset $A' \subseteq A$ with $|A'| \leqslant |A|/2$, A' is connected to at least $(1+\gamma)|A'|$ vertices in B, and vice versa, where γ is some fixed positive constant.

Consider a solution for the constructed MAX CUT instance and let the red part P_R be the part containing most of the b_j vertices. We will show that at least the number of cut edges we lose by moving all the b_j vertices that are in the green part P_G to P_R will be gained by cutting all edges in all expanders.

By moving the b_j vertices in P_G to P_R the number of cut edges not in the expander chain is decreased by at most $|\{b_j\} \cap P_G|$. On the other hand, by putting all the vertices in the layers $\{b_j\}$ and $\{c_{i,j}\}$ for every even i in P_R, and all the vertices in the layers $\{c_{i,j}\}$ for every odd i in P_G, we will show that the cardinality of the cut in the expander chain is increased by at least the same number.

To show this, we calculate the gain that is guaranteed between any layers of vertices achieved by putting all the vertices as specified above. We will make use of the property of bipartite expanders.

We first consider the number g_0 of uncut edges between the b_j vertices and the $c_{1,j}$ vertices in the given solution. By changing the partition as described above, the uncut edges will be cut, so the gain will be g_0. Let $m_0 = |\{b_j\} \cap P_G|$ and $m_1 = |\{c_{1,j}\} \cap P_R|$. By construction $m_0 \leqslant N/2$. We need to consider two cases.

Case 1: $m_1 \leqslant N/2$. The m_0 vertices in $\{b_j\} \cap P_G$ are connected to at least $(1+\gamma) \cdot m_0$ vertices in $\{c_{1,j}\}$. Of these vertices at least $(1+\gamma) \cdot m_0 - m_1$ must be in $\{c_{1,j}\} \cap P_G$, which means that at least $(1+\gamma) \cdot m_0 - m_1$ of the edges from $\{b_j\} \cap P_G$ are uncut. Similarly $(1+\gamma) \cdot m_1 - m_0$ of the edges from $\{c_{1,j}\} \cap P_R$ to $\{b_j\}$ are uncut. Therefore, $g_0 \geqslant (1+\gamma) \cdot m_0 - m_1 + (1+\gamma) \cdot m_1 - m_0 \geqslant \gamma(m_0 + m_1) \geqslant \gamma \cdot m_0$.

Case 2: $m_1 > N/2$. Since the expander property is valid only for subsets of size at most $N/2$ we just look at the noncut edges from a subset of size $N/2$ of $\{c_{1,j}\} \cap P_R$. Then we get

$$g_0 \geqslant (1+\gamma) \cdot (N/2) - m_0 \geqslant (1+\gamma) \cdot m_0 - m_0 = \gamma \cdot m_0.$$

Thus in both cases the number of uncut edges between the two layers of vertices is at least $\gamma \cdot m_0$.

By similarly counting noncut edges in the rest of the expander chain we will obtain that if at layer i the number m_i of vertices placed in the "wrong" part of the cut is no greater than $N/2$, the gain g_i between layer i and layer $i+1$ is at least $\gamma \cdot m_i$.

If in some layer i we have $m_i > N/2$, the gain g_i between layer i and layer $i+1$ can be calculated exactly as above, but with respect to the set of vertices placed in the right part instead of the wrong part. Indeed, in this case, the set of vertices placed in the right part has size $N - m_i \leqslant N/2$, and, thus, the property of bipartite expanders can be used. The gain then becomes $g_i \geqslant \gamma \cdot (N - m_i)$.

Therefore, if $(\frac{2}{3}) \cdot m_0 \leqslant m_i \leqslant N - (\frac{2}{3}) \cdot m_0$ for all i we will gain at least $(\frac{2}{3})\gamma \cdot m_0$ in each layer. If we choose $k \geqslant 3/(2 \cdot \gamma)$ the total gain will become at least m_0.

In order to consider the cases where m_i is small or large for some i we compute the gain in another way. The total number of expander edges from $\{b_j\} \cap P_G$ is $3m_0$, and the total number of edges from $\{c_{1,j}\} \cap P_R$ to $\{b_j\}$ is $3m_1$. Therefore, we can observe that if $m_0 > m_1$, at least $3(m_0 - m_1)$ of the edges from $\{b_j\} \cap P_G$ to $\{c_{1,j}\}$ must be incident to vertices in $\{c_{1,j}\} \cap P_G$ and thus must be uncut. In the same way, we get the gain $g_j \geqslant \max\{3(m_j - m_{j+1}), 0\}$ for the layer j. Suppose $m_i < (\frac{2}{3}) \cdot m_0$ for some i. Summing over all layers up to level i we get a total gain of at least $3(m_0 - m_i) > m_0$.

Finally, if for some i we have $m_i > N - (\frac{2}{3}) \cdot m_0$ we can do as for small m_i, but work in the other direction. Indeed, since $m_0 \leqslant N/2$, if $m_i > N - (\frac{2}{3}) \cdot m_0$, at least $3(m_1 - m_0)$ of the edges from $\{c_{1,j}\} \cap P_R$ to $\{b_j\}$ must be incident to vertices in $\{b_j\} \cap P_R$ and then must be uncut. Analogously, we get the gain $g_j \geqslant \max\{3(m_j - m_{j+1}), 0\}$ for the layer j. Therefore, if for some i we have $m_i > N - (\frac{2}{3}) \cdot m_0$, we sum over all layers up to level i and get a total gain of at least $3(N - (\frac{2}{3}) \cdot m_0 - (N - m_0)) > m_0$. Thus, we can move all b_j vertices to the same part without decreasing the size of the cut.

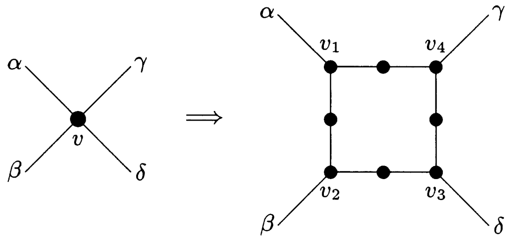

Fig. 6. A degree 4 vertex and the gadget originating from this vertex.

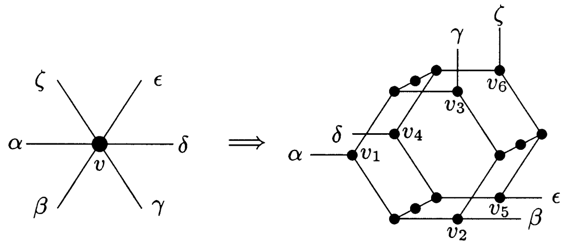

Fig. 7. A degree 6 vertex and the gadget originating from this vertex.

Finally, we have to show the first property of the L-reduction. If s is the number of satisfied clauses then the maximum number of cut edges is

$$2s + 2n + 4m + k \cdot 3N = 2s + 2n + (4 + 6k)m \leqslant (16 + 12k)s,$$

since we know that $s \geqslant m/2$ and $3m \geqslant 2n$. The reduction satisfies the first property of the L-reduction with $\alpha = 16 + 12k$ where $k = \lceil 3/(2 \cdot \gamma) \rceil$. □

Theorem 4.2. MAX CUT−3 *is* APX-*complete.*

Proof. We simply give a reduction from MAX CUT−6 for multigraphs to MAX CUT−3 for simple graphs.

For a vertex v of degree d, $2 \leqslant d \leqslant 5$ we do like follows: split the vertex into d *split vertices* $v_1, \ldots, v_d$ of degree 1. Then add d extra vertices and construct a ring where every other vertex is one of the split vertices and every other vertex is a new vertex, see Fig. 6.

It is easy to see that if the split vertices are put in the same part of the partition and the new vertices are put in the other part, then every edge in the ring will be cut. Otherwise, at least two of the edges in the ring will be uncut, and we can put all split vertices in the same part as the majority of them without decreasing the size of the cut.

For a vertex of degree 6 we do in a similar way, except that we need to construct two rings, one with v_1, v_2, v_3, and one with v_4, v_5, v_6. We add three new vertices to connect the two rings, see Fig. 7.

Let us study a given solution restricted to this gadget originating from v. Suppose $k \leqslant 3$ split vertices are in one part and the other $6 - k$ split vertices in the other part. If $k = 0$ we can put the rest of the vertices so that all edges in the gadget are cut.

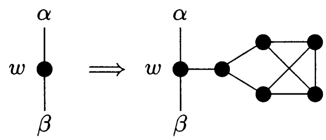

Fig. 8. The transformation of vertices of degree two in MAX CUT−3.

Otherwise, when putting all split vertices in the same part as the majority of them, it is easy to see that we can cut all edges in the gadget except k of the edges from the split vertices to the rest of the graph. We will show how to gain at least k edges in the gadget by putting all split vertices in the same part.

For each of the two rings we can reason as in the above case, thus showing that if all split vertices in one ring are not in the same part then we will lose at least two of the edges in that ring. This means that we have already gained enough if $k \leqslant 2$, or if $k = 3$ and we gained two edges from both rings. The only remaining case is when $k = 3$ and all split vertices in one ring are in the first part and all split vertices in the other ring are in the second part. But in this case three of the six edges connecting the two rings must be uncut, so we will gain at least three edges by putting all split vertices in the same part (and put the other vertices correctly).

Since we have split every vertex of degree at least 2 the constructed graph is, obviously, a simple graph and has maximum degree 3. Suppose m is the original number of edges and s is the maximum number of cut edges in the original problem. We know that $s \geqslant m/2$, and it is easy to check that we have added at most $3m$ edges. As we have seen all added edges can be cut, so the optimum solution of the constructed problem is bounded by $s + 3m \leqslant 7s$.

Thus, the constructed transformation is an L-reduction with $\alpha = 7$ and $\beta = 1$. □

MAX CUT is APX-hard even for cubic graphs. To show this we just have to extend the graph constructed in the above proof. Without loss of generality, we can assume that the graph does not contain any vertices of degree less than 2, and that two vertices of degree 2 never are neighbours. Transform each vertex w of degree two as shown in Fig. 8. It is not hard to show that this is an L-reduction with $\alpha = 8$ and $\beta = 1$.

Acknowledgements

We are grateful for comments from Magnus Halldórsson, S. Srinivasa Rao, and the anonymous referees.

References

[1] M. Ajtai, Recursive construction for 3-regular expanders, Proc. 28th Ann. IEEE Symp. on Foundations of Comput. Sci., 1987, pp. 295–304.

[2] E. Amaldi, V. Kann, The complexity and approximability of finding maximum feasible subsystems of linear relations, Theoret. Comput. Sci. 147 (1995) 181–210.
[3] S. Arora, S. Safra, Probabilistic checking of proof: a new characterization of NP, Proc. 33rd Annual IEEE Symp. on the Foundations of Computer Science, 1992, pp. 2–13.
[4] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, M. Protasi, Approximate Solution of NP-Hard Optimization Problems, Springer, Berlin, 1988, to appear.
[5] M. Bellare, O. Goldreich, M. Sudan, Free Bits, PCPs and non-approximability-towards tight results, Proc. of the 36th Annual IEEE Conf. on Foundations of Computer Science, 1995, pp. 422–431.
[6] P. Berman, T. Fujito, On approximation properties of the independent set problem for degree 3 graphs, Proc. 3rd Workshop on Algorithms and Data Structures, Lecture Notes in Computer Science, vol. 955, Springer, Berlin, 1995.
[7] P. Berman, M. Fürer, Approximating maximum independent set in bounded degree graphs, Proc. 5th Ann. ACM-SIAM Symp. on Discrete Algorithms, ACM-SIAM, 1994, pp. 365–371.
[8] P. Crescenzi, V. Kann, A compendium of NP optimization problems, Technical Report SI/RR-95/02, Dipartimento di Scienze dell'Informazione, Università di Roma "La Sapienza", 1995, The list is updated continuously. The latest version is available as http://www.nada.kth.se/theory/problemlist.html.
[9] P. Crescenzi, V. Kann, R. Silvestri, L. Trevisan, Structure in approximation classes, Proc. 1st Ann. Int. Conf. Computing and Combinatorics, Lecture Notes in Computer Science, vol. 959, Springer, Berlin, 1995, pp. 539–548.
[10] P. Crescenzi, A. Panconesi, Completeness in approximation classes, Inform. Comput. 93 (1991) 241–262.
[11] P. Crescenzi, R. Silvestri, L. Trevisan, To weight or not to weight: where is the question? Proc. 4th Ann. Israel Symp. Theory Comput. and Systems, IEEE, 1996, pp. 68–77.
[12] M.R. Garey, D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W.H. Freeman and Company, San Francisco, 1979.
[13] M.X. Goemans, D.P. Williamson, Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming, J. ACM 42 (1995) 1115–1145.
[14] R. Greenlaw, R. Petreschi, Cubic graphs, ACM Comput. Surveys 27 (1995) 471–495.
[15] M.M. Halldorsson, Approximating the minimum maximal independence number, Inform. Process. Lett. 46 (1993) 169–172.
[16] M.M. Halldórsson, Approximating k-set cover and complementary graph coloring, Proc. 5th Int. Conf. on Integer Prog. and Combinatorial Optimization, Lecture Notes in Computer Science, vol. 1084, Springer, Berlin, 1996, pp. 118–131.
[17] M.M. Halldórsson, K. Yoshihara, Greedy approximations of independent sets in low degree graphs, Proc. 6th Int. Symp. Algorithms and Comput., Lecture Notes in Computer Science, vol. 1004, Springer, Berlin, 1995, pp. 152–161.
[18] V. Kann, Maximum bounded 3-dimensional matching is MAX SNP-complete, Inform. Process. Lett. 37 (1991) 27–35.
[19] V. Kann, On the approximability of NP-complete optimization problems, Ph.D. Thesis, Department of Numerical Analysis and Computing Science, Royal Institute of Technology, Stockholm, 1992.
[20] V. Kann, Polynomially bounded minimization problems that are hard to approximate, Nordic J. Comput. 1 (1994) 317–331.
[21] H. Karloff, U. Zwick, A 7/8-approximation algorithm for MAX 3SAT?, Proc. 37th Ann. IEEE Symp. on Foundations of Comput. Sci., IEEE, 1997, pp. 406–415.
[22] C. Lund, M. Yannakakis, On the hardness of approximating minimization problems, J. ACM 41 (1994) 960–981.
[23] P. Orponen, H. Mannila, On approximation preserving reduction: complete problems and robust measures, Tech. Rep. C-1987-28, Department of Computer Science, Univ. of Helsinki, 1987.
[24] C.H. Papadimitriou, M. Yannakakis, Optimization, approximation, and complexity classes, J. Comput. Systems Sci. 43 (1991) 425–440.
[25] C.H. Papadimitriou, Computational Complexity, Addison-Wesley, Reading, MA, 1994.