Theoretical Informatics and Applications Will be set by the publisher
Informatique Théorique et Applications

RECOGNIZING WHEN HEURISTICS CAN APPROXIMATE MINIMUM VERTEX COVERS IS COMPLETE FOR PARALLEL ACCESS TO NP [*]

Edith Hemaspaandra[1], Jörg Rothe[2] and Holger Spakowski[2]

Abstract. For both the edge deletion heuristic and the maximum-degree greedy heuristic, we study the problem of recognizing those graphs for which that heuristic can approximate the size of a minimum vertex cover within a constant factor of r, where r is a fixed rational number. Our main results are that these problems are complete for the class of problems solvable via parallel access to NP. To achieve these main results, we also show that the restriction of the vertex cover problem to those graphs for which either of these heuristics can find an optimal solution remains NP-hard.

1991 Mathematics Subject Classification. 68Q15,68Q17.

1. Introduction

The minimum vertex cover problem is the problem of finding in a given graph a smallest possible set of vertices that covers at least one vertex of each edge. The decision version of the minimum vertex cover problem, VC, is one of the standard NP-complete problems [GJ79]. To cope with the intractability that appears to be inherent to this problem, various heuristics for finding minimum vertex covers have

Keywords and phrases: Computational complexity; completeness; minimum vertex cover heuristics; approximation; parallel access to NP.

[*] *This work was supported in part by the NSF under grant NSF-CCR-0311021, by the NSF and the DAAD under grant NSF-INT-9815095/DAAD-315-PPP-gü-ab, and by the DFG under grant RO 1202/9-1 and a Heisenberg Fellowship.*

[1] Department of Computer Science, Rochester Institute of Technology, Rochester, NY 14627, USA; e-mail: eh@cs.rit.edu.

[2] Institut für Informatik, Heinrich-Heine-Universität Düsseldorf, 40225 Düsseldorf, Germany; e-mail: rothe@cs.uni-duesseldorf.de & spakowsk@cs.uni-duesseldorf.de.

been proposed. Two of the most prominent such heuristics are the *edge deletion heuristic* and the *maximum-degree greedy heuristic*, see, e.g., [PS82, Pap94]. These algorithms run in linear time and, depending on the structure of the given input graph, may find a minimum vertex cover, or may provide a good approximation of the optimal solution.

It is common to evaluate heuristics for optimization problems by analyzing their worst-case ratio for approximating the optimal solution. In this regard, the two heuristics considered behave quite differently: the edge deletion heuristic always approximates the size of a minimum vertex cover within a factor of 2 and thus achieves the best approximation ratio known, whereas the maximum-degree greedy heuristic, in the worst case, can have an approximation ratio as bad as logarithmic in the input size. The latter result follows from the early analysis of the approximation behavior of the greedy algorithm for the minimum set cover problem that was done by Johnson [Joh74], Lovász [Lov75], and Chvátal [Chv79] (who studied the weighted version of minimum set cover). Note that the vertex cover problem is the special case of the set cover problem, restricted so that each element occurs in exactly two sets. More recently, building on the work of Lund and Yannakakis [LY94], Feige [Fei98] showed that, unless NP has slightly superpolynomial-time algorithms, the set cover problem cannot be approximated within $(1-\epsilon)\ln n$, where $\epsilon > 0$ and ln denotes the natural logarithm.

In this paper, we study the problem of recognizing those input graphs for which either of the two heuristics can approximate the size of a minimum vertex cover within a constant factor of r, where $r \geq 1$ is a fixed rational number. Let $\mathcal{S}_r^{\mathrm{ED}}$ and $\mathcal{S}_r^{\mathrm{MDG}}$, respectively, denote this recognition problem for the edge deletion heuristic and for the maximum-degree greedy heuristic. Our main results are:

- For each rational number r with $1 \leq r < 2$, $\mathcal{S}_r^{\mathrm{ED}}$ is $\mathrm{P}_{\parallel}^{\mathrm{NP}}$-complete (see Theorem 3.2).
- For each rational number $r \geq 1$, $\mathcal{S}_r^{\mathrm{MDG}}$ is $\mathrm{P}_{\parallel}^{\mathrm{NP}}$-complete (see Theorems 4.3 and 4.4).

Here, $\mathrm{P}_{\parallel}^{\mathrm{NP}}$ denotes the class of problems that can be decided in polynomial time by parallel (i.e., truth-table) access to NP. Papadimitriou and Zachos [PZ83] introduced this class under the name $\mathrm{P}^{\mathrm{NP}[\mathcal{O}(\log n)]}$, where "$[\mathcal{O}(\log n)]$" denotes that at most logarithmically many Turing queries are made to the NP oracle. Hemaspaandra [Hem89] proved that $\mathrm{P}^{\mathrm{NP}[\mathcal{O}(\log n)]} = \mathrm{P}_{\parallel}^{\mathrm{NP}}$, and in fact many more characterizations of $\mathrm{P}_{\parallel}^{\mathrm{NP}}$ are known [KSW87, Wag90]. Other natural $\mathrm{P}_{\parallel}^{\mathrm{NP}}$-hard and $\mathrm{P}_{\parallel}^{\mathrm{NP}}$-complete problems can be found in the papers by Wagner [Wag87], Köbler et al. [KSW87], Krentel [Kre88], Kadin [Kad89], Eiter and Gottlob [EG97], Hemaspaandra et al. [HHR97a, HR98, HW02, HSV], and Rothe et al. [RSV03, RR04]; see also the surveys [HHR97b, HH00].

The type of recognition problem studied in this paper was investigated for other problems and other heuristics as well. Bodlaender, Thilikos, and Yamazaki [BTY97] defined and studied the analogous problem for the independent set problem and the minimum-degree greedy heuristic, which they denoted by $\mathcal{S}_r$. They proved that

$\mathcal{S}_r$ is coNP-hard and belongs to P^{NP}. Closing the gap between these lower and upper bounds, Hemaspaandra and Rothe [HR98] proved that $\mathcal{S}_r$ is $\mathrm{P}_{\|}^{\mathrm{NP}}$-complete. As in [HR98], we obtain $\mathrm{P}_{\|}^{\mathrm{NP}}$-hardness by reducing from a problem (namely, $\mathtt{VC}_{\mathtt{geq}}$, see Section 2) that can be shown to be $\mathrm{P}_{\|}^{\mathrm{NP}}$-complete using the techniques of Wagner [Wag87]. Also, we show that the vertex cover problem, restricted to those input graphs for which the heuristics considered can find an optimal solution, remains NP-hard. We then lift these NP lower bounds to $\mathrm{P}_{\|}^{\mathrm{NP}}$ lower bounds that prove our main results. This lifting requires a padding technique such that the given approximation ratio r is precisely met. In particular, to achieve $\mathrm{P}_{\|}^{\mathrm{NP}}$-hardness of $\mathcal{S}_r^{\mathrm{MDG}}$ for each rational number $r \geq 1$, we modify a construction by Papadimitriou and Steiglitz [PS82] that they use to analyze the worst-case approximation behavior of the maximum-degree greedy heuristic.

2. Two Heuristics for the Vertex Cover Problem

We use the following notation. Fix the two-letter alphabet $\Sigma = \{0,1\}$. Σ^* is the set of all strings over Σ. Let $\langle\cdot,\cdot\rangle : \Sigma^* \times \Sigma^* \to \Sigma^*$ be a standard pairing function. For any set L, let $\|L\|$ denote the number of elements of L.

All graphs considered in this paper are undirected nonempty, finite graphs without multiple or reflexive edges. For any graph G, let $V(G)$ denote the set of vertices of G, and let $E(G)$ denote the set of edges of G. For any vertex $v \in V(G)$, the *degree of* v (denoted by $deg_G(v)$) is the number of vertices adjacent to v in G; if G is clear from the context, we omit the subscript and simply write $deg(v)$. Let $max\text{-}deg(G) = \max_{v\in V(G)} deg(v)$ denote the maximum degree of the vertices of graph G. Let G and H be two disjoint graphs. The *disjoint union of* G *and* H is defined to be the graph $U = G \cup H$ with vertex set $V(U) = V(G) \cup V(H)$ and edge set $E(U) = E(G) \cup E(H)$. The *join of* G *and* H is defined to be the graph $J = G \bowtie H$ with vertex set $V(J) = V(G) \cup V(H)$ and edge set $E(J) = E(G) \cup E(H) \cup \{\{x,y\} \mid x \in V(G) \ \wedge\ y \in V(H)\}$.

For any graph G, a subset $C \subseteq V(G)$ is a *vertex cover of* G if for all edges $\{v,w\} \in E(G)$, $\{v,w\} \cap C \neq \emptyset$. A vertex cover is said to be a *minimum vertex cover of* G if it is of minimum size. For any graph G, let $mvc(G)$ denote the size of a minimum vertex cover of G. The vertex cover problem ($\mathtt{VC}$, for short; see [GJ79]) is defined to be the set of all pairs $\langle G,k\rangle$ such that G is a graph, k a positive integer, and $mvc(G) \leq k$.

All hardness and completeness results in this paper are with respect to the polynomial-time many-one reducibility, denoted $\leq_{\mathrm{m}}^{\mathrm{p}}$. For sets A and B, we say $A \leq_{\mathrm{m}}^{\mathrm{p}} B$ if and only if there exists a polynomial-time computable function f such that for all inputs $x \in \Sigma^*$, $x \in A$ if and only if $f(x) \in B$.

We consider the following two heuristics (see, e.g., [PS82, Pap94]) for finding a minimum vertex cover of a given graph:

Edge Deletion Heuristic (ED): Given a graph G, the algorithm outputs a vertex cover C of G. Initially, C is the empty set. Nondeterministically choose an edge $\{u,v\} \in E(G)$, add both u and v to C, and delete u, v, and all edges incident to u and v from G. Repeat until there is no edge left in G.

Maximum-Degree Greedy Heuristic (MDG): Given a graph G, the algorithm outputs a vertex cover C of G. Initially, C is the empty set. Nondeterministically choose a vertex $v \in V(G)$ of maximum degree, add v to C, and delete v and all edges incident to v from G. Repeat until there is no edge left in G.

As mentioned in the introduction, these two heuristics have a quite different approximation behavior. While the worst-case ratio of the MDG algorithm is logarithmic in the input size [Pap94,Joh74], the ED algorithm always approximates the optimal solution within a factor of 2. Thus, despite its extreme simplicity, the edge deletion heuristic achieves the best approximation ratio known for finding minimum vertex covers [Pap94].

The central question raised in this paper is: How hard is it to determine for which graphs G either of these two heuristics can approximate the minimum vertex cover of G within a factor of r, for a given rational number $r \geq 1$? Let *min-ed*(G) (respectively, *min-mdg*(G)) denote the minimum size of the output set of the ED algorithm (respectively, of the MDG algorithm) on input G, where the minimum is taken over all possible sequences of nondeterministic choices the algorithms can make. For any fixed rational $r \geq 1$, $\mathcal{S}_r^{\mathrm{ED}}$ (respectively, $\mathcal{S}_r^{\mathrm{MDG}}$) is the class of graphs for which ED (respectively, MDG) can output a vertex cover of size at most r times the size of a minimum vertex cover. Formally,

$$\begin{aligned} \mathcal{S}_r^{\mathrm{ED}} &= \{G \mid G \text{ is a graph and } \textit{min-ed}(G) \leq r \cdot \textit{mvc}(G)\}; \\ \mathcal{S}_r^{\mathrm{MDG}} &= \{G \mid G \text{ is a graph and } \textit{min-mdg}(G) \leq r \cdot \textit{mvc}(G)\}. \end{aligned}$$

We will prove that for each fixed rational number r with $1 \leq r < 2$, $\mathcal{S}_r^{\mathrm{ED}}$ is $\mathrm{P}_{\|}^{\mathrm{NP}}$-complete, and that for each fixed rational number $r \geq 1$, $\mathcal{S}_r^{\mathrm{MDG}}$ is $\mathrm{P}_{\|}^{\mathrm{NP}}$-complete. To this end, we give reductions from the problem $\mathtt{VC}_{\mathtt{geq}}$, which is defined by

$$\mathtt{VC}_{\mathtt{geq}} = \{\langle G, H\rangle \mid G \text{ and } H \text{ are graphs such that } \textit{mvc}(G) \geq \textit{mvc}(H)\}.$$

It is known that $\mathtt{VC}_{\mathtt{geq}}$ is $\mathrm{P}_{\|}^{\mathrm{NP}}$-complete, cf. Wagner [Wag87]. A reduction from any problem in $\mathrm{P}_{\|}^{\mathrm{NP}}$ to $\mathtt{VC}_{\mathtt{geq}}$ that in addition has some useful properties (see Lemma 2.1 below) can easily be obtained using the techniques of Wagner [Wag87]; see [SV00, Thm. 12] for an explicit proof of Lemma 2.1.

Lemma 2.1 (cf. [Wag87,SV00]). *For any set $X \in \mathrm{P}_{\|}^{\mathrm{NP}}$, there exists a polynomial-time computable function f that reduces X to $\mathtt{VC}_{\mathtt{geq}}$ in such a way that for each $x \in \Sigma^*$, $f(x) = \langle G, H\rangle$ is an instance of $\mathtt{VC}_{\mathtt{geq}}$ and*

$$\begin{aligned} x \in X &\implies \textit{mvc}(G) = \textit{mvc}(H); \\ x \notin X &\implies \textit{mvc}(G) < \textit{mvc}(H). \end{aligned}$$

3. The Edge Deletion Heuristic

Lemma 3.1 below states that the vertex cover problem restricted to graphs in $\mathcal{S}_1^{\mathrm{ED}}$ is NP-hard. The reduction g from Lemma 3.1 will be used in the proof of the main result of this section, Theorem 3.2. Define the problem

$$\texttt{VC-}\mathcal{S}_1^{\mathrm{ED}} = \{\langle G, k\rangle \mid G \in \mathcal{S}_1^{\mathrm{ED}} \text{ and } k \in \mathbb{N}^+ \text{ and } mvc(G) \leq k\}.$$

Lemma 3.1. *There exists a polynomial-time many-one reduction g from* VC *to* $\texttt{VC-}\mathcal{S}_1^{\mathrm{ED}}$ *transforming any given graph G into a graph $H \in \mathcal{S}_1^{\mathrm{ED}}$ such that*

$$mvc(H) = 2(mvc(G) + \|V(G)\|). \tag{1}$$

Hence, $\texttt{VC-}\mathcal{S}_1^{\mathrm{ED}}$ *is* NP*-hard.*[1]

Proof. Given any graph G, we construct the graph $H \in \mathcal{S}_1^{\mathrm{ED}}$ as follows. For each vertex $v \in V(G)$, create a component G_v that is defined by the vertex set $V(G_v) = \{v_1, v_2, v_3, v_4\}$ and the edge set $E(G_v) = \{\{v_1, v_2\}, \{v_3, v_4\}, \{v_1, v_3\}\}$.

Define the graph H by joining every pair of components that correspond to adjacent vertices of G:

$$\begin{aligned} V(H) &= \bigcup_{v \in V(G)} V(G_v); \\ E(H) &= \{\{a_i, b_j\} \mid \{a, b\} \in E(G) \text{ and } i, j \in \{1, 2, 3, 4\}\} \cup \bigcup_{v \in V(G)} E(G_v). \end{aligned}$$

We now prove Equation (1). Let C be a minimum vertex cover of G, i.e., $mvc(G) = \|C\|$. Construct a vertex cover D of H as follows. For each vertex $v \in C$, add v_1, v_2, v_3, and v_4 to D; and for each vertex $w \in V(G) - C$, add w_1 and w_3 to D. Hence,

$$\|D\| = 2(\|C\| + \|V(G)\|).$$

Since $mvc(H) \leq \|D\|$, it follows that

$$mvc(H) \leq 2(mvc(G) + \|V(G)\|).$$

Conversely, let D be a minimum vertex cover of H, i.e., $mvc(H) = \|D\|$. Then, it holds that:

- for each edge $\{u, v\} \in E(G)$, $V(G_u) \subseteq D$ or $V(G_v) \subseteq D$;
- for each vertex $v \in V(G)$, $\|D \cap V(G_v)\| \geq 2$.

Hence,

$$\begin{aligned} \|D\| &\geq 4 \cdot mvc(G) + 2(\|V(G)\| - mvc(G)) \\ &= 2(mvc(G) + \|V(G)\|). \end{aligned}$$

[1] Theorem 3.2 will imply that $\texttt{VC-}\mathcal{S}_1^{\mathrm{ED}}$ in fact is $\mathrm{P}_{\|}^{\mathrm{NP}}$-complete.

It follows that

$$mvc(H) \geq 2(mvc(G) + \|V(G)\|),$$

which proves Equation (1).

It remains to prove that $H \in \mathcal{S}_1^{\mathrm{ED}}$. Let C be a minimum vertex cover of G. The edge deletion algorithm can find a vertex cover of H as follows. For every vertex $v \in C$, choose the edges $\{v_1, v_2\}$ and $\{v_3, v_4\}$. For the remaining vertices $w \in V(G) - C$, choose the edge $\{w_1, w_3\}$. Thus, *min-ed*$(H) = 2(mvc(G) + \|V(G)\|)$. By Equation (1), *min-ed*$(H) = mvc(H)$, so $H \in \mathcal{S}_1^{\mathrm{ED}}$. □

Theorem 3.2. *For each rational number r with $1 \leq r < 2$, $\mathcal{S}_r^{\mathrm{ED}}$ is $\mathrm{P}_{\|}^{\mathrm{NP}}$-complete.*

Proof. It is easy to see that $\mathcal{S}_r^{\mathrm{ED}}$ is in $\mathrm{P}_{\|}^{\mathrm{NP}}$. To prove $\mathrm{P}_{\|}^{\mathrm{NP}}$-hardness, let X be an arbitrary set in $\mathrm{P}_{\|}^{\mathrm{NP}}$, and let f be the reduction from X to $\mathtt{VC_{geq}}$ stated in Lemma 2.1. Fix any rational number r with $1 \leq r < 2$, and let ℓ and m be integers such that $r = \ell/m$. Note that $1 \leq m \leq \ell < 2m$.

For any string $x \in \Sigma^*$, let $f(x) = \langle G_1, G_2 \rangle$. Since we can add isolated vertices to any graph G without altering $mvc(G)$, we may without loss of generality assume that $\|V(G_1)\| = \|V(G_2)\|$. Let g be the reduction from Lemma 3.1 that transforms any given graph G into a graph $H \in \mathcal{S}_1^{\mathrm{ED}}$ such that Equation (1) holds. Let $H_1 = g(G_1)$ and $H_2 = g(G_2)$. Thus, both H_1 and H_2 are in $\mathcal{S}_1^{\mathrm{ED}}$, and for $i \in \{1, 2\}$, we have $mvc(H_i) = 2(mvc(G_i) + \|V(G_i)\|)$.

We will define a graph $\widehat{H}$ and an integer $k \geq 0$ such that:

$$\begin{aligned} \textit{min-ed}(\widehat{H}) &= r(m \cdot mvc(H_2) + 2km); \qquad (2) \\ mvc(\widehat{H}) &= m \cdot mvc(H_1) + 2km. \qquad (3) \end{aligned}$$

The reduction mapping any given string x (via the pair $\langle G_1, G_2 \rangle$ obtained according to Lemma 2.1 and via the pair $\langle H_1, H_2 \rangle$ obtained according to Lemma 3.1) to the graph $\widehat{H}$ such that Equations (2) and (3) are satisfied will establish that $X \leq_{\mathrm{m}}^{\mathrm{p}} \mathcal{S}_r^{\mathrm{ED}}$. In particular, from these equations, we have that:

- $mvc(H_2) = mvc(H_1)$ implies *min-ed*$(\widehat{H}) = r \cdot mvc(\widehat{H})$, and
- $mvc(H_2) > mvc(H_1)$ implies *min-ed*$(\widehat{H}) > r \cdot mvc(\widehat{H})$.

Note that, due to Lemma 2.1, $mvc(H_2) \geq mvc(H_1)$.

Look at Figure 1 for the construction of $\widehat{H}$ from H_1 and H_2. The graph $\widehat{H}$ consists of two subgraphs, L and R, that are joined by the join operation, plus some additional vertices and edges that are connected to R. Formally, let $H_1^1, H_1^2, \ldots, H_1^m$ be m pairwise disjoint copies of H_1, and let $H_2^1, H_2^2, \ldots, H_2^\ell$ be ℓ pairwise disjoint copies of H_2. Let $k = \ell\|V(H_2)\| + m\|V(H_1)\|$. Let I_1 and I_2 be independent sets such that L contains exactly $k(2m - \ell)$ vertices and R exactly $k\ell$ vertices. (This is possible, because $k(2m - \ell) - \ell\|V(H_2)\|$ is not negative, since $2m - \ell \geq 1$, and $k\ell - m\|V(H_1)\|$ is not negative, since $\ell \geq 1$.) Let $e_i = \{a_i, b_i\}$ $(1 \leq i \leq k\ell)$ be additional edges. Every vertex a_i is adjacent to exactly one vertex in R, and each vertex in R is adjacent to exactly one vertex a_i. The vertices a_i and b_i are not adjacent to any other vertices.

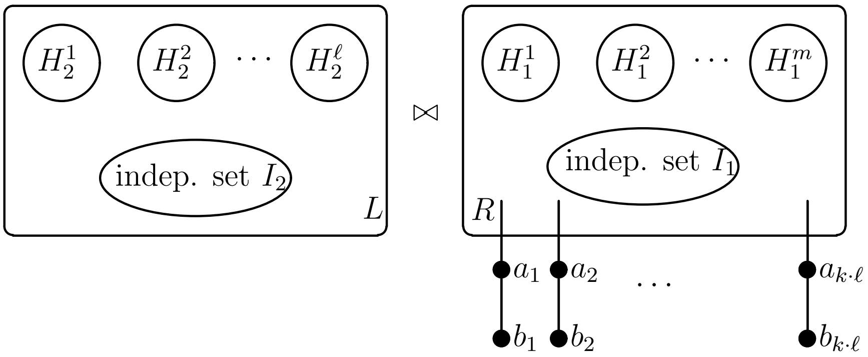

Figure 1. The graph $\widehat{H}$ constructed from H_1 and H_2.

(1) We first determine $\mathit{min\text{-}ed}(\widehat{H})$. Let $\widehat{E}$ be a fixed minimum-size output set of the ED algorithm on input $\widehat{H}$, i.e., $\mathit{min\text{-}ed}(\widehat{H}) = \|\widehat{E}\|$. Since $\widehat{E}$ is a vertex cover of $\widehat{H}$, $\widehat{E}$ must contain a_i or b_i for each $i \in \{1, \ldots, k\ell\}$. Since the ED-algorithm can delete only edges, and $\widehat{E}$ is a minimum-size output set, it follows that $\widehat{E}$ contains all vertices a_i, all vertices from R, and no vertex b_i.

Let C_L be a minimum-size output set of the ED-algorithm on input L. By construction of L, $\|C_L\| = \ell \cdot \mathit{min\text{-}ed}(H_2)$. Thus, since $H_2 \in \mathcal{S}_1^{\mathrm{ED}}$, $\|C_L\| = \ell \cdot \mathit{mvc}(H_2)$.

Define $\widehat{E}' = V(R) \cup C_L \cup \bigcup_{i=1}^{k\ell}\{a_i\}$. It is easy to see that $\widehat{E}'$ is a minimum-size output set of the ED algorithm on input $\widehat{H}$. Hence,

$$\begin{aligned} \mathit{min\text{-}ed}(\widehat{H}) &= 2k\ell + \ell \cdot \mathit{mvc}(H_2) \\ &= r(2km + m \cdot \mathit{mvc}(H_2)). \end{aligned}$$

This proves Equation (2).

(2) We now determine $\mathit{mvc}(\widehat{H})$. Let $\widehat{C}$ be a fixed minimum vertex cover of $\widehat{H}$, i.e., $\mathit{mvc}(\widehat{H}) = \|\widehat{C}\|$. Distinguish the following two cases.

Case 1: $V(R) \subseteq \widehat{C}$. In this case, $\widehat{C}$ contains all vertices from R, at least one of a_i or b_i for each i, $1 \leq i \leq k\ell$, and a minimum vertex cover of L. Hence,

$$\mathit{mvc}(\widehat{H}) = 2k\ell + \ell \cdot \mathit{mvc}(H_2).$$

Case 2: $V(L) \subseteq \widehat{C}$. In this case, $\widehat{C}$ contains all vertices from L, a minimum vertex cover of R, and exactly one of a_i or b_i for each i, $1 \leq i \leq k\ell$. Hence,

$$\begin{aligned} \mathit{mvc}(\widehat{H}) &= k(2m - \ell) + k\ell + m \cdot \mathit{mvc}(H_1) \\ &= 2km + m \cdot \mathit{mvc}(H_1). \end{aligned}$$

Since $mvc(H_1) \leq mvc(H_2)$, $m \leq \ell$, and $2km \leq 2k\ell$, it follows that

$$mvc(\widehat{H}) = 2km + m \cdot mvc(H_1).$$

This proves Equation (3).

The proof of Theorem 3.2 is complete. □

4. The Maximum-Degree Greedy Heuristic

Lemma 4.1 below states that the vertex cover problem restricted to graphs in $\mathcal{S}_1^{\mathrm{MDG}}$ is NP-hard. The proof of Lemma 4.1 is reminiscent of a proof by Bodlaender et al. [BTY97, Thm. 4], who show that the independent set problem restricted to graphs for which the minimum-degree greedy heuristic can find an optimal solution is NP-hard. The reduction g from Lemma 4.1 will be used in the proof of the main result of this section, Theorem 4.4. Define the problem

$$\texttt{VC-}\mathcal{S}_1^{\mathrm{MDG}} = \{\langle G, k\rangle \mid G \in \mathcal{S}_1^{\mathrm{MDG}} \text{ and } k \in \mathbb{N}^+ \text{ and } mvc(G) \leq k\}.$$

Lemma 4.1. *There is a polynomial-time many-one reduction g from* VC *to* $\texttt{VC-}\mathcal{S}_1^{\mathrm{MDG}}$ *transforming any given graph G into a graph $H \in \mathcal{S}_1^{\mathrm{MDG}}$ such that*

$$mvc(H) = mvc(G) + \|E(G)\|(max\text{-}deg(G) + 1). \tag{4}$$

Hence, $\texttt{VC-}\mathcal{S}_1^{\mathrm{MDG}}$ *is* NP*-hard.*[2]

Proof. Given any graph G, we construct the graph $H \in \mathcal{S}_1^{\mathrm{MDG}}$ as follows. Let $\Delta = max\text{-}deg(G)$. Replace each edge of G by a gadget that contains a complete bipartite graph of size $2(\Delta + 1)$. Formally, H is defined by:

$$\begin{aligned} V(H) &= V(G) \cup \bigcup_{e = \{u,v\} \in E(G)} \left(\{u_i^e \mid 1 \leq i \leq \Delta + 1\} \cup \{v_i^e \mid 1 \leq i \leq \Delta + 1\}\right); \\ E(H) &= \bigcup_{e = \{u,v\} \in E(G)} \left(\{\{u_i^e, v_j^e\} \mid 1 \leq i, j \leq \Delta + 1\} \cup \{\{u, u_1^e\}\} \cup \{\{v, v_1^e\}\}\right). \end{aligned}$$

We now prove Equation (4). Let C be a minimum vertex cover of G, i.e., $mvc(G) = \|C\|$. Note that $\{u, v\} \cap C \neq \emptyset$ for each edge $\{u, v\}$ in $E(G)$. Construct a vertex cover D of H as follows:

- D contains all vertices from C.
- For every edge $e = \{u, v\}$ in $E(G)$, add to D:
  - either all vertices u_i^e, $1 \leq i \leq \Delta + 1$, if $u \notin C$ or if both u and v are in C;
  - or all vertices v_i^e, $1 \leq i \leq \Delta + 1$, if $v \notin C$.

[2] Theorem 4.3 will imply that $\texttt{VC-}\mathcal{S}_1^{\mathrm{MDG}}$ in fact is $\mathrm{P}_{\|}^{\mathrm{NP}}$-complete.

It follows that

$$mvc(H) \leq mvc(G) + \|E(G)\|(\Delta + 1).$$

Conversely, let D be a minimum vertex cover of H, i.e., $mvc(H) = \|D\|$. Construct a vertex cover C of G as follows. Initially, set $C = D$. Let $e = \{u, v\}$ be any fixed edge in $E(G)$. Suppose that at least one vertex from $\{u, v\}$ is in D. Since D is a vertex cover of H, it contains at least $\Delta + 1$ of the vertices u_i^e and v_i^e, $1 \leq i \leq \Delta + 1$, that correspond to the edge e. Remove any $\Delta + 1$ such vertices from C. Suppose now that neither u nor v is in D. Since D is a vertex cover of H, it contains at least $\Delta + 2$ of the vertices u_i^e and v_i^e, $1 \leq i \leq \Delta + 1$, that correspond to the edge e. Remove any $\Delta + 2$ such vertices from C, and add to C one of u or v instead. Since the set C thus obtained is a vertex cover of G, we have

$$mvc(H) \geq mvc(G) + \|E(G)\|(\Delta + 1),$$

which proves Equation (4).

It remains to prove that $H \in \mathcal{S}_1^{\mathrm{MDG}}$. Let C be a minimum vertex cover of G. The maximum-degree greedy algorithm can find a vertex cover of H as follows. For every edge $e = \{u, v\}$ in $E(G)$, the MDG algorithm on input H can choose:

- either all vertices u_i^e, $1 \leq i \leq \Delta + 1$, if $u \notin C$ or if both u and v are in C;
- or all vertices v_i^e, $1 \leq i \leq \Delta + 1$, if $v \notin C$.

Note that the MDG heuristic can always do so, since every vertex in $V(G)$ has degree at most Δ. Subsequently, all vertices that are not in C are isolated. Thus, the MDG algorithm can now choose all vertices from C. Hence,

$$\textit{min-mdg}(H) = mvc(G) + \|E(G)\|(\Delta + 1).$$

By Equation (4), $\textit{min-mdg}(H) = mvc(H)$, so $H \in \mathcal{S}_1^{\mathrm{MDG}}$. □

Lemma 4.2 below will be used in the proof of Theorem 4.4. The construction of the graph G in this lemma is a modification of a construction given by Papadimitriou and Steiglitz [PS82, p. 408, Fig. 17-3], which shows that the worst-case approximation ratio of the MDG heuristic can be as bad as logarithmic in the input size, and so grows unboundedly. Similar constructions for achieving the worst-case approximation behavior of the greedy heuristic solving the more general minimum set cover problem were given by Johnson [Joh74], Lovász [Lov75], and Chvátal [Chv79].

Lemma 4.2. *For all positive integers n_1, n_2, $\delta \geq 6$, and μ satisfying*

$$\mu(\ln \mu - 2\ln(\delta + 2) - 1) \geq n_1 + n_2, \tag{5}$$

there exists a bipartite graph G with the following properties:

(1) *$V(G) = V \cup \tilde{V}$ such that $V \cap \tilde{V} = \emptyset$ and both V and $\tilde{V}$ are independent sets, where*
   - *$V = \{u_1, u_2, \ldots, u_{n_1}, w_1, w_2, \ldots, w_\mu, z_1, z_2, \ldots z_{n_2}\}$ and*

- $\tilde{V} = \{\tilde{u}_1, \tilde{u}_2, \dots, \tilde{u}_{n_1}, \tilde{w}_1, \tilde{w}_2, \dots, \tilde{w}_\mu\}$.

(2) $\{\{u_i, \tilde{u}_i\} \mid 1 \leq i \leq n_1\} \cup \{\{w_i, \tilde{w}_i\} \mid 1 \leq i \leq \mu\} \subseteq E(G)$.
(3) *Every vertex $\tilde{u}_i$, where $1 \leq i \leq n_1$, has degree one.*
(4) *For each induced subgraph S of G that can be obtained by deleting vertices from V such that $V \cap V(S) \neq \emptyset$, it holds that*

$$\max_{v \in V \cap V(S)} deg_S(v) > \max_{v \in \tilde{V}} deg_S(v) + \delta.$$

Proof. Let the constants n_1, n_2, δ, and μ be given such that Equation (5) is satisfied. We describe the construction of the graph G. As stated in the lemma, the vertex set of G is given by $V(G) = V \cup \tilde{V}$, where V and $\tilde{V}$ are two disjoint independent sets. Rename the vertices of V by $V = \{\alpha_1, \alpha_2, \dots, \alpha_{n_1+\mu+n_2}\}$.

Let $\tilde{W} = \{\tilde{w}_1, \tilde{w}_2, \tilde{w}_3, \dots, \tilde{w}_\mu\}$. The edge set of G is defined as follows:

- Create the edges $\{u_i, \tilde{u}_i\}$ for each i with $1 \leq i \leq n_1$ and the edges $\{w_j, \tilde{w}_j\}$ for each j with $1 \leq j \leq \mu$.
- Partition $\tilde{W}$ into $\left\lfloor \frac{\mu}{\delta+3} \right\rfloor$ disjoint sets $\tilde{W}_1^{\delta+3}, \tilde{W}_2^{\delta+3}, \dots, \tilde{W}_{\lfloor \frac{\mu}{\delta+3} \rfloor}^{\delta+3}$ of size $\delta+3$ each, possibly leaving out some vertices from $\tilde{W}$ and taking care that no vertex in $\tilde{W}_i^{\delta+3}$ already is connected with α_i, $1 \leq i \leq \left\lfloor \frac{\mu}{\delta+3} \right\rfloor$. For each i with $1 \leq i \leq \left\lfloor \frac{\mu}{\delta+3} \right\rfloor$, connect α_i with each vertex in $\tilde{W}_i^{\delta+3}$ by an edge.
- Partition $\tilde{W}$ into $\left\lfloor \frac{\mu}{\delta+4} \right\rfloor$ disjoint sets $\tilde{W}_1^{\delta+4}, \tilde{W}_2^{\delta+4}, \dots, \tilde{W}_{\lfloor \frac{\mu}{\delta+4} \rfloor}^{\delta+4}$ of size $\delta+4$ each, possibly leaving out some vertices from $\tilde{W}$ and taking care that no vertex in $\tilde{W}_i^{\delta+4}$ already is connected with $\alpha_{\lfloor \frac{\mu}{\delta+3} \rfloor + i}$, $1 \leq i \leq \left\lfloor \frac{\mu}{\delta+4} \right\rfloor$. For each i with $1 \leq i \leq \left\lfloor \frac{\mu}{\delta+4} \right\rfloor$, connect $\alpha_{\lfloor \frac{\mu}{\delta+3} \rfloor + i}$ with each vertex in $\tilde{W}_i^{\delta+4}$ by an edge.
- Partition $\tilde{W}$ into $\left\lfloor \frac{\mu}{\delta+5} \right\rfloor$ disjoint sets $\tilde{W}_1^{\delta+5}, \tilde{W}_2^{\delta+5}, \dots, \tilde{W}_{\lfloor \frac{\mu}{\delta+5} \rfloor}^{\delta+5}$ of size $\delta+5$ each, possibly leaving out some vertices from $\tilde{W}$ and taking care that no vertex in $\tilde{W}_i^{\delta+5}$ already is connected with $\alpha_{\lfloor \frac{\mu}{\delta+3} \rfloor + \lfloor \frac{\mu}{\delta+4} \rfloor + i}$, $1 \leq i \leq \left\lfloor \frac{\mu}{\delta+5} \right\rfloor$. For each i with $1 \leq i \leq \left\lfloor \frac{\mu}{\delta+5} \right\rfloor$, connect $\alpha_{\lfloor \frac{\mu}{\delta+3} \rfloor + \lfloor \frac{\mu}{\delta+4} \rfloor + i}$ with each vertex in $\tilde{W}_i^{\delta+5}$ by an edge.
- Continue in this way until all vertices α_i are connected with vertices in $\tilde{W}$.

The construction is possible, since Equation (5) implies

$$\left\lfloor \frac{\mu}{\delta+3} \right\rfloor + \left\lfloor \frac{\mu}{\delta+4} \right\rfloor + \dots + \left\lfloor \frac{\mu}{\mu-1} \right\rfloor \geq n_1 + \mu + n_2, \tag{6}$$

and thus there are enough vertices in $\tilde{W}$. To see why, note that

$$\begin{aligned}
&\left\lfloor\frac{\mu}{\delta+3}\right\rfloor+\left\lfloor\frac{\mu}{\delta+4}\right\rfloor+\cdots+\left\lfloor\frac{\mu}{\mu-1}\right\rfloor\\
&= \left\lfloor\frac{\mu}{1}\right\rfloor+\left\lfloor\frac{\mu}{2}\right\rfloor+\cdots+\left\lfloor\frac{\mu}{\mu}\right\rfloor-1-\left(\left\lfloor\frac{\mu}{1}\right\rfloor+\left\lfloor\frac{\mu}{2}\right\rfloor+\cdots+\left\lfloor\frac{\mu}{\delta+2}\right\rfloor\right)\\
&\geq \frac{\mu}{1}+\frac{\mu}{2}+\cdots+\frac{\mu}{\mu}-\mu-\left(\frac{\mu}{1}+\frac{\mu}{2}+\cdots+\frac{\mu}{\delta+2}\right)\\
&= \mu H_\mu-\mu-\mu H_{\delta+2}\\
&\geq \mu\ln\mu-\mu-\mu\ln(\delta+2)-\mu\\
&\geq \mu\ln\mu-\mu\ln(\delta+2)-2\mu\\
&\geq \mu\ln\mu-\mu\ln(\delta+2)-\mu\ln(\delta+2) \qquad (7)\\
&= \mu\ln\mu-2\mu\ln(\delta+2).
\end{aligned}$$

Here, H_k denotes the kth harmonic number, which is defined by $H_k=\sum_{i=1}^{k}\frac{1}{i}$. It is well known that for all k, $\ln k \leq H_k \leq \ln k+1$ (see, e.g., Graham, Knuth, and Patashnik [GKP89]). Equation (7) holds because $\delta \geq 6$ and hence $\ln(\delta+2)\geq 2$.

It is evident from the construction that G has all the required properties. In particular, to see why Property 4 holds, let S be any induced subgraph of G that can be obtained by deleting vertices from V such that $V\cap V(S)\neq\emptyset$. Let $y_S=\max_{v\in V\cap V(S)} deg_S(v)$. By construction, S can have only edges of the form $\{u_i,\tilde{u}_i\}$ or $\{w_j,\tilde{w}_j\}$ or edges that are added during the stages $\delta+3,\delta+4,\ldots,y_S$, where $\delta+i$ denotes the stage in which $\tilde{W}$ is partitioned into subsets of size $\delta+i$. It follows that

$$\max_{v\in\tilde{V}} deg_S(v)\leq 1+y_S-(\delta+3)+1=y_S-\delta-1<y_S-\delta,$$

which proves the lemma. □

Theorem 4.3. *$\mathcal{S}_1^{\mathrm{MDG}}$ is $\mathrm{P}_{\|}^{\mathrm{NP}}$-complete.*

Proof. It is easy to see that $\mathcal{S}_1^{\mathrm{MDG}}$ is in $\mathrm{P}_{\|}^{\mathrm{NP}}$. To prove $\mathrm{P}_{\|}^{\mathrm{NP}}$-hardness of $\mathcal{S}_1^{\mathrm{MDG}}$, let X be an arbitrary set in $\mathrm{P}_{\|}^{\mathrm{NP}}$, and let f be the reduction from X to $\texttt{VC}_{\texttt{geq}}$ stated in Lemma 2.1. For any string $x\in\Sigma^*$, let $f(x)=\langle G_1,G_2\rangle$.

We will define a graph $\widehat{G}$ and an integer $q\geq 0$ such that:

$$\begin{aligned}
\textit{min-mdg}(\widehat{G}) &= \textit{mvc}(G_2)+q; \qquad (8)\\
\textit{mvc}(\widehat{G}) &= \textit{mvc}(G_1)+q. \qquad (9)
\end{aligned}$$

The reduction mapping any given string x (via the pair $\langle G_1,G_2\rangle$ obtained according to Lemma 2.1) to the graph $\widehat{G}$ such that Equations (8) and (9) are satisfied will establish that $X\leq_{\mathrm{m}}^{\mathrm{p}}\mathcal{S}_1^{\mathrm{MDG}}$. In particular, from these equations, we have that:

- $mvc(G_2) = mvc(G_1)$ implies $min\text{-}mdg(\widehat{G}) = mvc(\widehat{G})$, and
- $mvc(G_2) > mvc(G_1)$ implies $min\text{-}mdg(\widehat{G}) > mvc(\widehat{G})$.

Note that, due to Lemma 2.1, $mvc(G_2) \geq mvc(G_1)$.

We now describe the construction of $\widehat{G}$. Let g be the reduction from Lemma 4.1 and let $H_2 = g(G_2)$. Thus, H_2 is in $\mathcal{S}_1^{\mathrm{MDG}}$ and, by Equation (4),

$$mvc(H_2) = mvc(G_2) + \|E(G_2)\|(max\text{-}deg(G_2) + 1). \tag{10}$$

Since one can add isolated vertices to any graph G without affecting the values of $mvc(G)$ or $min\text{-}mdg(G)$, we may without loss of generality assume that

$$\|V(H_2)\| = \|V(G_1)\| + \|E(G_2)\|(max\text{-}deg(G_2) + 1). \tag{11}$$

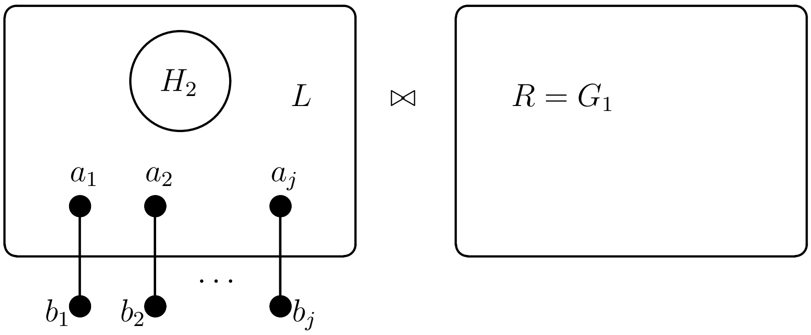

Figure 2. The graph $\widehat{G}$ constructed from G_1 and H_2.

Look at Figure 2 for the construction of $\widehat{G}$ from G_1 and H_2. The graph $\widehat{G}$ consists of two subgraphs, L and R, that are joined by the join operation, plus some additional vertices and edges that are connected to L. Formally, choose $2j$ new vertices a_i and b_i, $1 \leq i \leq j$, where j is a fixed integer large enough such that the degree of each vertex in R is larger than the maximum degree of the vertices in L. Note that the degree of each vertex in R must remain larger than the degree of any vertex in L even after some vertices have been removed from R.

Let B be the bipartite matching with the vertex set

$$V(B) = \{a_i \mid 1 \leq i \leq j\} \cup \{b_i \mid 1 \leq i \leq j\}$$

and the edge set $E(B) = \{\{a_i, b_i\} \mid 1 \leq i \leq j\}$. Let $R = G_1$, and let L be the graph with the vertex set $V(L) = \{a_i \mid 1 \leq i \leq j\} \cup V(H_2)$ and the edge set $E(L) = E(H_2)$. The graph $\widehat{G}$ is defined by forming the join $L \bowtie R$, i.e., there are edges connecting each vertex of L with each vertex of R, plus attaching the vertices b_i, $1 \leq i \leq j$, to L by adding the j edges from $E(B)$.

We first consider $min\text{-}mdg(\widehat{G})$. By our choice of j, each vertex in R has a degree larger than the degree of any vertex not in R. Hence, on input $\widehat{G}$, the MDG algorithm first deletes all vertices from R. Subsequently, it can find a minimum vertex cover of H_2, which has size $mvc(G_2) + \|E(G_2)\|(max\text{-}deg(G_2)+1)$ by Equation (10), and eventually it can choose, say, the vertices a_i, $1 \leq i \leq j$, to cover the edges of B. Hence,

$$\begin{aligned} min\text{-}mdg(\widehat{G}) &= \|V(G_1)\| + mvc(G_2) + \|E(G_2)\|(max\text{-}deg(G_2)+1) + j \\ &\overset{(11)}{=} mvc(G_2) + \|V(H_2)\| + j. \end{aligned}$$

We now consider $mvc(\widehat{G})$. Since every vertex cover of $\widehat{G}$ must contain all vertices of L or all vertices of R to cover the edges connecting L and R, it follows from Equations (10) and (11) that:

$$\begin{aligned} mvc(\widehat{G}) &= \min\{\|V(G_1)\| + mvc(H_2) + j,\ \|V(H_2)\| + j + mvc(G_1)\} \\ &= \min\{mvc(G_2) + \|V(H_2)\| + j,\ mvc(G_1) + \|V(H_2)\| + j\}. \end{aligned}$$

Since $mvc(G_2) \geq mvc(G_1)$, it follows that

$$mvc(\widehat{G}) = mvc(G_1) + \|V(H_2)\| + j.$$

Hence, setting $q = \|V(H_2)\| + j$, Equations (8) and (9) are satisfied, which completes the proof that $\mathcal{S}_1^{\mathrm{MDG}}$ is $\mathrm{P}_{\|}^{\mathrm{NP}}$-complete. □

Theorem 4.4. *For each rational number $r > 1$, $\mathcal{S}_r^{\mathrm{MDG}}$ is $\mathrm{P}_{\|}^{\mathrm{NP}}$-complete.*

Proof. Fix any rational number $r = \ell/m$, where ℓ and m are integers with $1 \leq m < \ell$. Without loss of generality, we may assume that $\gcd(\ell - m, m) = 1$, where $\gcd(a, b)$ denotes the greatest common divisor of the integers a and b. It is easy to see that $\mathcal{S}_r^{\mathrm{MDG}}$ is in $\mathrm{P}_{\|}^{\mathrm{NP}}$. To prove $\mathrm{P}_{\|}^{\mathrm{NP}}$-hardness of $\mathcal{S}_r^{\mathrm{MDG}}$, let X be an arbitrary set in $\mathrm{P}_{\|}^{\mathrm{NP}}$, and let f be the reduction from X to $\mathtt{VC_{geq}}$ stated in Lemma 2.1. For any string $x \in \Sigma^*$, let $f(x) = \langle G_1, G_2\rangle$. Note that $mvc(G_2) \geq mvc(G_1)$.

We will define a graph $\widehat{G}_r$ and integers $p, q \geq 0$ such that:

$$\begin{aligned} min\text{-}mdg(\widehat{G}_r) &= r(p \cdot mvc(G_2) + q); && (12) \\ mvc(\widehat{G}_r) &= p \cdot mvc(G_1) + q. && (13) \end{aligned}$$

The reduction mapping any given string x (via the pair $\langle G_1, G_2\rangle$ obtained according to Lemma 2.1) to the graph $\widehat{G}_r$ such that Equations (12) and (13) are satisfied will establish that $X \leq_{\mathrm{m}}^{\mathrm{p}} \mathcal{S}_r^{\mathrm{MDG}}$. In particular, from these equations, we have that:

- $mvc(G_2) = mvc(G_1)$ implies $min\text{-}mdg(\widehat{G}_r) = r \cdot mvc(\widehat{G}_r)$, and
- $mvc(G_2) > mvc(G_1)$ implies $min\text{-}mdg(\widehat{G}_r) > r \cdot mvc(\widehat{G}_r)$.

We now describe the construction of $\widehat{G}_r$:

- Let g be the reduction from Lemma 4.1 and let $H_2 = g(G_2)$. Thus, $H_2 \in \mathcal{S}_1^{\mathrm{MDG}}$ and Equation (10) holds:

$$\mathit{mvc}(H_2) = \mathit{mvc}(G_2) + \|E(G_2)\|(\mathit{max\text{-}deg}(G_2) + 1).$$

- Let $G_1^1, G_1^2, \ldots, G_1^m$ be m pairwise disjoint copies of G_1, and let H_2^1, H_2^2, …, H_2^ℓ be ℓ pairwise disjoint copies of H_2.
- Let $\tilde{U} = \bigcup_{i=1}^{\ell} H_2^i$ be the disjoint union of these copies of H_2, and rename the vertices of $\tilde{U}$ by $V(\tilde{U}) = \{\tilde{u}_1, \tilde{u}_2, \ldots, \tilde{u}_{\ell \cdot \|V(H_2)\|}\}$.
- Let $Z = \bigcup_{i=1}^{m} G_1^i$ be the disjoint union of these copies of G_1, and rename the vertices of Z by $V(Z) = \{z_1, z_2, \ldots, z_{m \cdot \|V(G_1)\|}\}$.
- To apply Lemma 4.2, choose $n_1 = \ell \cdot \|V(H_2)\|$, $n_2 \geq m \cdot \|V(G_1)\|$, and $\delta = \max\{6, \mathit{max\text{-}deg}(H_2)+1\}$, where the exact value of n_2 will be specified below. Choose the constant μ so as to satisfy Equation (5):

$$\mu(\ln \mu - 2\ln(\delta + 2) - 1) \geq n_2 + n_1.$$

- Given the constants n_1, n_2, δ, and μ, define $\widehat{G}_r$ to be the bipartite graph G from Lemma 4.2 extended by the edges between the $\tilde{u}_i$ vertices that were added above to represent the structure of the copies of H_2, and extended by the edges between the z_j vertices that were added above to represent the structure of the copies of G_1. That is, unlike G, the graph $\widehat{G}_r$ is no longer a bipartite graph. Formally, the vertex set of $\widehat{G}_r$ is given by

$$\begin{aligned} V(\widehat{G}_r) &= V(G) = V \cup \tilde{V}, \quad \text{where} \\ V &= \{u_1, u_2, \ldots, u_{n_1}, w_1, w_2, \ldots, w_\mu, z_1, z_2, \ldots z_{n_2}\} \quad \text{and} \\ \tilde{V} &= \{\tilde{u}_1, \tilde{u}_2, \ldots, \tilde{u}_{n_1}, \tilde{w}_1, \tilde{w}_2, \ldots, \tilde{w}_\mu\}, \end{aligned}$$

and the edge set of $\widehat{G}_r$ is given by $E(\widehat{G}_r) = E(G) \cup E(\tilde{U}) \cup E(Z)$, where $E(G)$ is constructed as in the proof of Lemma 4.2.

This completes the construction of $\widehat{G}_r$. We now prove Equations (12) and (13).

(1) We first consider $\mathit{min\text{-}mdg}(\widehat{G}_r)$. By construction, for each vertex v in $\tilde{V}$, we have

$$\mathit{deg}_{\widehat{G}_r}(v) \leq \mathit{deg}_G(v) + \mathit{max\text{-}deg}(H_2) < \mathit{deg}_G(v) + \delta. \tag{14}$$

Let S be any induced subgraph of $\widehat{G}_r$ that can be obtained by deleting vertices from V such that $V \cap V(S) \neq \emptyset$. Property 4 of Lemma 4.2 and Equation (14) imply that

$$\max_{v \in V \cap V(S)} \mathit{deg}_S(v) > \max_{v \in \tilde{V}} \mathit{deg}_S(v).$$

Hence, on input $\widehat{G}_r$, the MDG algorithm starts by choosing the $n_1+\mu+n_2$ vertices from V, which isolates each vertex $\tilde{w}_i \in \tilde{V}$ and leaves ℓ isolated copies of H_2. Subsequently, since $H_2 \in \mathcal{S}_1^{\text{MDG}}$, the MDG algorithm can choose a minimum vertex cover in each of these ℓ copies of H_2. By Equation (10),

$$mvc(H_2) = mvc(G_2) + \|E(G_2)\|(max\text{-}deg(G_2) + 1),$$

and hence,

$$min\text{-}mdg(\widehat{G}_r) = n_1 + \mu + n_2 + \ell(mvc(G_2) + \|E(G_2)\|(max\text{-}deg(G_2) + 1)).$$

(2) We now consider $mvc(\widehat{G}_r)$. Define the set $C = \tilde{V} \cup D$, where D with $\|D\| = m \cdot mvc(G_1)$ is a minimum vertex cover of Z. It is obvious from the construction of $\widehat{G}_r$ that C is a minimum vertex cover of $\widehat{G}_r$. Hence,

$$mvc(\widehat{G}_r) = n_1 + \mu + m \cdot mvc(G_1).$$

To complete the proof, we have to choose $n_2 \geq m \cdot \|V(G_1)\|$ such that Equations (12) and (13) are satisfied for suitable integers p and q. Setting $p = m$ and $q = n_1 + \mu$ and requiring

$$n_1 + n_2 + \mu + \ell \cdot \|E(G_2)\|(max\text{-}deg(G_2) + 1) = r(n_1 + \mu) \tag{15}$$

or, equivalently,

$$m \cdot n_2 + m \cdot \ell \cdot \|E(G_2)\|(max\text{-}deg(G_2) + 1) = (\ell - m)n_1 + (\ell - m)\mu \tag{16}$$

satisfies Equations (12) and (13).

Our assumption that $\gcd(\ell - m, m) = 1$ implies that the equation

$$m \cdot n_2' + 1 = (\ell - m)\mu' \tag{17}$$

has integer solutions. Fix one such solution (n_2', μ'). Multiplying this solution with $m \cdot \ell \cdot \|E(G_2)\|(max\text{-}deg(G_2) + 1) - (\ell - m)n_1$, we obtain an integer solution $(\widehat{n_2}, \widehat{\mu})$ for Equation (16). For every $k \in \mathbb{Z}$, (n_2, μ) is a solution of Equation (16), where

$$n_2 = \widehat{n_2} + k(\ell - m), \tag{18}$$
$$\mu = \widehat{\mu} + km. \tag{19}$$

Choosing k large enough, we can make sure that (a) n_2 and μ are positive integers,[3] (b) $n_2 \geq m \cdot \|V(G_1)\|$, and (c) n_2 and μ satisfy Equation (5) for given n_1 and δ. It is easy to see that k can be small enough such that n_2 and μ are polynomially bounded in the size of the input of the reduction being described. This completes the proof of the theorem. □

[3] Recall that $\ell - m > 0$ and $m > 0$.

In conclusion, we have shown that for both the edge deletion heuristic and the maximum-degree greedy heuristic, it is $\mathrm{P}_{\|}^{\mathrm{NP}}$-complete to recognize those graphs for which the heuristic can approximate the size of a minimum vertex cover within a constant factor of r, where r is a fixed rational number. These results add new problems ($\mathcal{S}_r^{\mathrm{ED}}$ and $\mathcal{S}_r^{\mathrm{MDG}}$) to the list of $\mathrm{P}_{\|}^{\mathrm{NP}}$-complete problems previously obtained.

Acknowledgments: We thank Dieter Kratsch and Andreas Brandstädt for interesting discussions on graph theory and graph-theoretical notation. We thank the anonymous referees for their helpful comments.

References

[BTY97] H. Bodlaender, D. Thilikos, and K. Yamazaki. It is hard to know when greedy is good for finding independent sets. *Information Processing Letters*, 61:101–106, 1997.

[Chv79] V. Chvátal. A greedy heuristic for the set-covering problem. *Mathematics of Operations Research*, 4(3):233–235, 1979.

[EG97] T. Eiter and G. Gottlob. The complexity class Θ_2^p: Recent results and applications. In *Proceedings of the 11th Conference on Fundamentals of Computation Theory*, pages 1–18. Springer-Verlag *Lecture Notes in Computer Science #1279*, September 1997.

[Fei98] U. Feige. A threshold of $\ln n$ for approximating set cover. *Journal of the ACM*, 45(4):634–652, July 1998.

[GJ79] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York, 1979.

[GKP89] R. Graham, D. Knuth, and O. Patashnik. *Concrete Mathematics*. Addison-Wesley, 1989.

[Hem89] L. Hemachandra. The strong exponential hierarchy collapses. *Journal of Computer and System Sciences*, 39(3):299–322, 1989.

[HH00] E. Hemaspaandra and L. Hemaspaandra. Computational politics: Electoral systems. In *Proceedings of the 25th International Symposium on Mathematical Foundations of Computer Science*, pages 64–83. Springer-Verlag *Lecture Notes in Computer Science #1893*, 2000.

[HHR97a] E. Hemaspaandra, L. Hemaspaandra, and J. Rothe. Exact analysis of Dodgson elections: Lewis Carroll's 1876 voting system is complete for parallel access to NP. *Journal of the ACM*, 44(6):806–825, November 1997.

[HHR97b] E. Hemaspaandra, L. Hemaspaandra, and J. Rothe. Raising NP lower bounds to parallel NP lower bounds. *SIGACT News*, 28(2):2–13, June 1997.

[HR98] E. Hemaspaandra and J. Rothe. Recognizing when greed can approximate maximum independent sets is complete for parallel access to NP. *Information Processing Letters*, 65(3):151–156, February 1998.

[HSV] E. Hemaspaandra, H. Spakowski, and J. Vogel. The complexity of Kemeny elections. *Theoretical Computer Science*. Accepted subject to minor revision.

[HW02] E. Hemaspaandra and G. Wechsung. The minimization problem for boolean formulas. *SIAM Journal on Computing*, 31(6):1948–1958, 2002.

[Joh74] D. Johnson. Approximation algorithms for combinatorial problems. *Journal of Computer and System Sciences*, 9(3):256–278, 1974.

[Kad89] J. Kadin. $\mathrm{P}^{\mathrm{NP}[\log n]}$ and sparse Turing-complete sets for NP. *Journal of Computer and System Sciences*, 39(3):282–298, 1989.

[Kre88] M. Krentel. The complexity of optimization problems. *Journal of Computer and System Sciences*, 36:490–509, 1988.

[KSW87] J. Köbler, U. Schöning, and K. Wagner. The difference and truth-table hierarchies for NP. *R.A.I.R.O. Informatique théorique et Applications*, 21:419–435, 1987.

[Lov75] L. Lovász. On the ratio of optimal integral and fractional covers. *Discrete Mathematics*, 13:383–390, 1975.

[LY94] C. Lund and M. Yannakakis. On the hardness of approximating minimization problems. *Journal of the ACM*, 41(5):960–981, September 1994.

[Pap94] C. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.

[PS82] C. Papadimitriou and K. Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Prentice-Hall, 1982.

[PZ83] C. Papadimitriou and S. Zachos. Two remarks on the power of counting. In *Proceedings of the 6th GI Conference on Theoretical Computer Science*, pages 269–276. Springer-Verlag *Lecture Notes in Computer Science #145*, 1983.

[RR04] T. Riege and J. Rothe. Complexity of the exact domatic number problem and of the exact conveyor flow shop problem. *Theory of Computing Systems*, December 2004. On-line publication DOI 10.1007/s00224-004-1209-8. Paper publication to appear.

[RSV03] J. Rothe, H. Spakowski, and J. Vogel. Exact complexity of the winner problem for Young elections. *Theory of Computing Systems*, 36(4):375–386, June 2003.

[SV00] H. Spakowski and J. Vogel. Θ_2^p-completeness: A classical approach for new results. In *Proceedings of the 20th Conference on Foundations of Software Technology and Theoretical Computer Science*, pages 348–360. Springer-Verlag *Lecture Notes in Computer Science #1974*, December 2000.

[Wag87] K. Wagner. More complicated questions about maxima and minima, and some closures of NP. *Theoretical Computer Science*, 51:53–80, 1987.

[Wag90] K. Wagner. Bounded query classes. *SIAM Journal on Computing*, 19(5):833–846, 1990.

Communicated by (The editor will be set by the publisher).
March 3, 2005.