Opuscula Math. 36, no. 3 (2016), 375–397
http://dx.doi.org/10.7494/OpMath.2016.36.3.375 Opuscula Mathematica

Dedicated to Professor Stepan E. Markosyan

THE HARDNESS OF THE INDEPENDENCE AND MATCHING CLUTTER OF A GRAPH

Sasun Hambardzumyan, Vahan V. Mkrtchyan, Vahe L. Musoyan, and Hovhannes Sargsyan

Communicated by Mariusz Meszka

Abstract. A *clutter* (or *antichain* or *Sperner family*) L is a pair (V, E), where V is a finite set and E is a family of subsets of V none of which is a subset of another. Usually, the elements of V are called *vertices* of L, and the elements of E are called *edges* of L. A subset s_e of an edge e of a clutter is called *recognizing* for e, if s_e is not a subset of another edge. The *hardness* of an edge e of a clutter is the ratio of the size of e's smallest recognizing subset to the size of e. The hardness of a clutter is the maximum hardness of its edges. We study the hardness of clutters arising from independent sets and matchings of graphs.

Keywords: clutter, hardness, independent set, maximal independent set, matching, maximal matching.

Mathematics Subject Classification: 05C69, 05C70, 05C15.

1. INTRODUCTION

A *clutter* (or *antichain* or *Sperner family*) L is a pair (V, E), where V is a finite set and E is a family of subsets of V none of which is a subset of another. Following [2], the elements of V will be called *vertices* of L, and the elements of E are called *edges* of L.

Given a clutter $L = (V, E)$, a subset $e_0 \subseteq e$ of an edge e is a *recognizing* subset for e, if $e_0 \subseteq e'$ for some $e' \in E$, then $e' = e$. Let S_e be a smallest recognizing subset of $e \in E$, $c(e) = |S_e|/|e|$, and

$$c(L) = \max_{e \in E} c(e).$$

$c(L)$ is called the *hardness* of L. Note that $0 \leq c(L) \leq 1$ for any clutter $L = (V, E)$. Moreover, if $|E| \leq 1$, then clearly $c(L) = 0$. Thus, it is natural to consider clutters L

© AGH University of Science and Technology Press, Krakow 2016

with at least two edges. In this case any edge contains no more than $|V|-1$ vertices and any recognizing subset of an edge $e \in E$ must contain at least one vertex. Thus,

$$\frac{1}{|V|-1} \leq c(L) \leq 1,$$

and the lower bound is tight, since if for any positive integer n ($n \geq 2$) we take $V_n = \{1, \ldots, n\}$, $E_n = \{\{1, 3, \ldots, n\}, \{2, 3, \ldots, n\}\}$, then clearly $L_n = (V_n, E_n)$ is a clutter, the sets $\{1\}$ and $\{2\}$ are recognizing subsets for the edges $\{1, 3, \ldots, n\}$ and $\{2, 3, \ldots, n\}$, respectively, and

$$c(L_n) = \frac{1}{n-1} = \frac{1}{|V_n|-1}.$$

Note that the main reason why the clutter L_n has such a low hardness, is that the elements $3, \ldots, n$ are present in every edge. This means that they cannot be present in any smallest recognizing subset of an edge, and therefore they do not contribute to the numerator of the hardness of an edge, however, they do contribute to its denominator. This situation prompts the following problem.

Problem 1.1. *Find a best-possible function f such that any clutter $L = (V, E)$ satisfying the condition*

(C1) *no vertex of L is present in all edges of L,*

has hardness $c(L) \geq f(|V|)$.

Our considerations above imply that $f(|V|) \geq \frac{1}{|V|-1}$. Let us also note that any clutter satisfying (C1) has at least two edges, moreover, without loss of generality, we can assume that the clutters in the formulation of Problem 1.1 satisfy

(C2) each vertex of L is present in at least one edge of L,

since isolated vertices (vertices, that do not belong to an edge) can be removed from the clutter without affecting its hardness.

In this paper, we address Problem 1.1 for two classes of clutters that arise from graphs. Let us note that the graphs considered in this paper are finite, undirected and do not contain multiple edges or loops. Formally, such a graph G can be considered as a clutter (V, E), in which E is any subset of the set of pairs of elements from V.

For a graph G, let $V(G)$ and $E(G)$ be the sets of vertices and edges of G, respectively. There is an important comment that should be made here concerning the terminology. An edge of a clutter is a subset of the set of vertices, and therefore it can contain more than two vertices, however, an edge of a graph contains exactly two vertices.

For a vertex $v \in V(G)$ let $d(v)$ be the *degree* of v, and let $\Delta(G)$ be the *maximum degree* of a vertex of G. If $E \subseteq E(G)$, then let $V(E)$ be the set of vertices of G, which are incident to an edge from E. For $S \subseteq V(G)$ let $G[S]$ denote the subgraph of G *induced* by the set S.

If u, v are vertices of a graph G, then let $\rho(u, v)$ denote the *distance* between the two vertices, that is, the length of the shortest path connecting vertices u and v. Moreover, let $diam(G)$ denote the *diameter* of G, that is, the maximum distance among any two vertices of G.

For a positive integer n let K_n denote the complete graph on n vertices. If m and n are positive integers, then assume $K_{m,n}$ to be the complete bipartite graph one side of which has m vertices and the other side has n vertices.

A set $V' \subseteq V(G)$ is said to be *independent*, if V' contains no adjacent vertices. Similarly, $E' \subseteq E(G)$ is independent, if E' contains no adjacent edges. An independent set of vertices (edges) is called *maximal*, if it does not lie in a larger independent set. An independent set of edges is also called *matching*.

Independent sets give rise to clutters. If for a graph $G = (V, E)$ we denote the set of all maximal independent sets of vertices of G by U_G, then (V, U_G) is a clutter. In the paper we use $\mathcal{U}_G$ to denote the clutter (V, U_G). We will need the following properties:

(P1) any independent set of vertices (particularly, a vertex) of a graph G can be extended to a member of U_G.

Note that if a vertex v belongs to a member U_v of U_G, then the neighbours of v are not in U_v. This implies that

(P2) in any graph G, $\min_{U \in U_G} |U| + \Delta(G) \leq |V(G)|$.

If a graph G contains at least one edge, then U_G contains at least two maximal independent sets. Hence, the empty set is not a recognizing subset. Consider a smallest maximal independent set U_0 from U_G. (P2) implies that $|U_0| \leq |V(G)| - \Delta(G)$. Since the empty set is not a recognizing subset for U_0, we have that $c(U_0) \geq \frac{1}{|V(G)|-\Delta(G)}$. Thus,

(P3) if $|E(G)| \geq 1$, then $c(\mathcal{U}_G) \geq \frac{1}{|V(G)|-\Delta(G)}$.

Another clutter that a graph $G = (V, E)$ gives rise is (E, M_G), where M_G denotes the set of all maximal matchings of G. This clutter will be denoted by $\mathcal{M}_G$.

The aim of this paper is the investigation of $c(\mathcal{U}_G)$ and $c(\mathcal{M}_G)$. In Theorem 2.4 in Section 2, we show that

$$c(\mathcal{U}_G) \geq \frac{1}{1 + |V(G)| - 2\sqrt{|V(G)| - 1}}$$

provided that G is a connected graph different from $K_1, K_{2,2}, K_{3,3}, K_{4,4}$. Moreover, we show that this bound is attained by infinitely many graphs. Note that this implies that in the search of the function f for Problem 1.1, one should restrict herself/himself exclusively to those functions that satisfy the following inequality:

$$\frac{1}{|V| - 1} \leq f(|V|) \leq \frac{1}{1 + |V| - 2\sqrt{|V| - 1}}.$$

The following example shows that the last inequality is not sharp. Let k be any positive integer with $k \geq 2$. Take $n = k^2$, and let U_0 be a set with $n - k = k(k-1)$ elements. Consider an n-vertex graph G obtained from a k-clique Q, by joining every vertex of Q to $k - 1$ elements of U_0 (each element of U_0 is joined to exactly one vertex of Q).

Note that $U_0 \in U_G$. Let L be the clutter that is obtained from $\mathcal{U}_G$ by removing the edge U_0. Observe that all edges of L contain exactly $1+(k-1)^2$ vertices, moreover, a set comprised of a vertex of Q is a smallest recognizing subsets for an edge of L. Thus

$$c(L) = \frac{1}{1+(k-1)^2}.$$

It is routine to verify that since $k \geq 2$, we have

$$c(L) < \frac{1}{1+|V|-2\sqrt{|V|-1}}.$$

In Theorem 2.6 we show that any rational number between 0 and 1 is a hardness of a clutter $\mathcal{U}_G$ for some graph G. In the end of Section 2, we make an attempt to characterize the class of trees T, for which $c(\mathcal{U}_T) = 1$. Though we fail to do this, we present some necessary and some sufficient conditions. We close the section by giving some examples of trees, which show that our conditions are merely necessary or sufficient.

In Section 3, we investigate the hardness of clutters $\mathcal{M}_G$. In a direct analogy with Theorem 2.6, we show that any rational number between 0 and 1 is a hardness of a clutter $\mathcal{M}_G$ for some graph G. Theorem 3.4 offers a tight and a better bound for $c(\mathcal{M}_G)$, than one can derive from Theorem 2.4. And finally, in the end of the section we investigate the hardness of clutters $\mathcal{M}_G$ arising from regular graphs.

The final Section 4 is devoted to the investigation of some computational problems that are intimately related to the algorithmic computation of $c(\mathcal{U}_G)$. Our investigations show that these problems are NP-hard. Let us note that we failed to achieve similar results for the clutters $\mathcal{M}_G$.

Data compression provides a suitable language for the explanation of the essence of our hardness. Suppose that we want to save a maximal matching H of a graph G. Of course, it does not make sense for us to save the whole matching H. We can keep only its smallest recognizing subset H_S, as the set $H \backslash H_S$ is unique and it can be easily reconstructed from H_S. Clearly, $c(\mathcal{M}_G)$ shows the relative hardness of the "worst" maximal matching of G.

The hardness of a clutter, that we introduce in the paper, is new (see [1,4] where the authors introduce two different types of hardness for graphs). Terms and concepts that we do not define can be found in [2,5,7].

2. THE HARDNESS OF $\mathcal{U}_G$

We start with a lemma, which for a fixed $U \in U_G$ gives a necessary and sufficient condition for a set to be recognizing for U.

Lemma 2.1. *Let $U \in U_G$. A set $U_0 \subseteq U$ is recognizing for U if and only if each vertex $v \in V(G) \backslash U$ has a neighbour in U_0.*

Proof. Necessity. Assume that U_0 is recognizing for U. Let us show that each vertex lying outside U has a neighbour in U_0. Suppose that there is a vertex $v \in V(G) \backslash U$ that has no neighbour in U_0. Then, due to (P1), there is $U' \in U_G$ such that $U_0 \cup \{v\} \subseteq U'$. Note that $U' \neq U$ since $v \notin U$. Taking into account that $U_0 \subseteq U$ and $U_0 \subseteq U'$, we deduce that the set U_0 is not recognizing for U, which contradicts our assumption.

Sufficiency. Now assume that each vertex lying outside U has a neighbour in U_0. Let us show that U_0 is recognizing for U. Suppose that the set U_0 is not recognizing for U. Then there is $U' \in U_G$, $U' \neq U$ such that $U_0 \subseteq U'$. Since $U' \neq U$, there is $v \in U' \backslash U$. Note that the vertex v has no neighbour in the set U_0, a contradiction. □

Corollary 2.2. *If $U \in U_G$ and there is a vertex $v \in V(G)\backslash U$ that has only one neighbour u in the set U, then all recognizing sets of U contain the vertex u.*

Our next result gives some structural properties of connected graphs G, for which any smallest recognizing set of $U \in U_G$ has exactly one vertex.

Lemma 2.3. *Let $G = (V, E)$ be a connected graph such that all the smallest recognizing sets of members of U_G contain one vertex. Then:*

(a) *for each $U \in U_G$ and its smallest recognizing set S_U, the vertex from S_U is adjacent to all vertices outside U;*
(b) $\min_{U \in U_G} |U| + \Delta(G) = |V(G)|$;
(c) *Suppose that $U_G = \{U_1, \ldots, U_l\}$. Define*

$$S_G = \{v \in V(G) : v \text{ lies in exactly one } U \in U_G\},$$

and for $i = 1, \ldots, l$ let $S_G(U_i) = \{x \in S_G : x \in U_i\}$. Then any l vertices $u_1, \ldots, u_l$ with $u_i \in S_G(U_i)$ induce a maximum clique of G. Moreover, every maximum clique of G can be obtained in this way;
(d) *$diam(G) \leq 3$.*

Proof. (a) directly follows from Lemma 2.1.

(b) Choose $U_0 \in U_G$ with $|U_0| = \min_{U \in U_G} |U|$. According to (a), there is an $x \in U_0$ that is adjacent to all vertices from $V(G)\backslash U$. Note that

$$\Delta(G) \geq d(x) = |V(G)| - |U_0| = |V(G)| - \min_{U \in U_G} |U|,$$

thus

$$\Delta(G) \geq |V(G)| - \min_{U \in U_G} |U|.$$

(P2) implies that

$$\Delta(G) + \min_{U \in U_G} |U| = |V(G)|.$$

(c) Let $U_i \in U_G$ and $U_j \in U_G$ $(i \neq j)$, and consider vertices $v_i \in S_G(U_i)$ and $v_j \in S_G(U_j)$. Clearly, $v_i \notin U_j$ and $v_j \notin U_i$, hence due to (a) $(v_i, v_j) \in E(G)$. This implies that any vertices $u_1, \ldots, u_l$ with $u_i \in S_G(U_i), i = 1, \ldots, l$ induce a clique of G, and particularly, the size of the maximum clique of G is at least l.

Thus to complete the proof of (c), we only need to show that for any maximum clique Q of the graph G there are $u_1, \ldots, u_l$ with $u_i \in S_G(U_i), i = 1, \ldots, l$, such that $V(Q) = \{u_1, \ldots, u_l\}$.

Let Q be a maximum clique of the graph G, and let $U \in U_G$. Clearly, $|V(Q) \cap U| \leq 1$. Let us show that $|V(Q) \cap U| = 1$. If $V(Q) \cap U = \emptyset$, then due to (a), there is $x \in U$ such that x is adjacent to all vertices of Q. This implies that the set $V(Q) \cup \{x\}$ forms a larger clique of G contradicting the choice of Q.

Thus $|V(Q) \cap U| = 1$. Suppose that $V(Q) \cap U = \{x\}$. Let us show that $x \in S_G(U)$. Suppose not. Then there is $U' \in U_G, U' \neq U$ such that $x \in U'$. Clearly, $V(Q) \cap U' = \{x\}$. Let $u \in U$ and $u' \in U'$ be vertices such that $\{u\}$ and $\{u'\}$ are recognizing subsets for U and U', respectively. (a) implies that the vertices u and u' are adjacent to all vertices lying outside U and U', respectively. Since $x \in U, U'$, we imply that u and u' do not belong to the clique Q. Now, it is not hard to see that the set $(V(Q) \backslash \{x\}) \cup \{u, u'\}$ induces a clique that is larger than Q contradicting the choice of Q. Thus $x \in S_G(U)$ and the proof of (c) is completed.

(d) Suppose that $diam(G) \geq 4$, and consider the vertices $u, v \in V(G)$ with $\rho(u,v) = diam(G) \geq 4$. Let $u = u_0, u_1, \ldots, u_k = v$, $k = \rho(u,v) \geq 4$ be a shortest path connecting the vertices u and v. Note that $(u_1, u_3) \notin E(G)$, thus due to (P1), there is $U \in U_G$ with $\{u_1, u_3\} \subseteq U$. Let $z \in U$ be a vertex such that $\{z\}$ is recognizing for U. (a) implies that $(u, z) \in E(G)$ and $(u_4, z) \in E(G)$. Note that $u = u_0, z, u_4, \ldots, u_k = v$ is a path connecting the vertices u and v, whose length is smaller than $k = \rho(u,v)$, which is a contradiction. The proof of the Lemma 2.3 is completed. $\square$

We are ready to present the first main result of the paper, which is a tight lower bound for $c(\mathcal{U}_G)$ in the class of connected graphs G if one is willing to disregard finitely many exceptions.

Theorem 2.4. *If $G = (V, E)$ is a connected graph, with $|V(G)| \geq 2$, that is not isomorphic to $K_{2,2}, K_{3,3}, K_{4,4}$, then*

$$c(\mathcal{U}_G) \geq \frac{1}{1 + |V(G)| - 2\sqrt{|V(G)| - 1}}.$$

Proof. Suppose that there is a $U \in U_G$ with $|S_U| \geq 2$, where S_U is a smallest recognizing subset for U. Since G is connected and $|V(G)| \geq 2$, we have $|U| \leq |V(G)| - 1$, thus

$$c(\mathcal{U}_G) \geq c(U_0) = \frac{|S_{U_0}|}{|U_0|} \geq \frac{2}{|V(G)| - 1} \geq \frac{1}{1 + |V(G)| - 2\sqrt{|V(G)| - 1}}.$$

Thus, without loss of generality, we may assume that for each $U \in U_G$ we have $|S_U| = 1$. Note that if we could prove that in such graphs

$$|V(G)| \leq 1 + \left(\frac{1 + \Delta(G)}{2}\right)^2, \tag{2.1}$$

which is equivalent to

$$\Delta(G) \geq 2\sqrt{|V(G)| - 1} - 1,$$

then, due to (P3), we would have

$$c(\mathcal{U}_G) \geq \frac{1}{1 + |V(G)| - 2\sqrt{|V(G)| - 1}},$$

and the proof of the theorem would be complete. Thus, to complete the proof, it suffices to show that if G is a graph satisfying the conditions of Theorem 2.4 and for each $U \in U_G$ we have $|S_U| = 1$, then the inequality (2.1) holds.

Let $U_G = \{U_1, \ldots, U_l\}$, and suppose Q is a maximum clique of G with $V(Q) = \{v_1, \ldots, v_l\}$, $v_i \in S_G(U_i), i = 1, \ldots, l$ (see (c) of Lemma 2.3). Set: $V_0 = V(G) \backslash V(Q)$.

First of all, let us show that each $x \in V_0$ has a neighbour in Q. Since G is connected and $|V(G)| \geq 2$, there is a $y \in V(G)$ such that $(x, y) \in E(G)$. Due to (P1), there is a $U_y \in U_G$ containing the vertex y. Due to (a) and (c) of Lemma 2.3 there is a $z \in V(Q) \cap U_y$ such that z is adjacent to all vertices lying outside U_y, and particularly, to x.

To complete the proof of the theorem, we need to consider three cases. Note that since G is a connected graph with at least two vertices, we have $l \geq 2$.

Case 1. $l = 2$.

Due to (c) of Lemma 2.3, l is the size of a maximum clique of G, thus G does not contain a triangle. We claim that G is bipartite. Suppose not, and let C be the shortest odd cycle of the graph G, with $V(C) = \{z_1, \ldots, z_k\}$, $E(C) = \{(z_1, z_2), \ldots, (z_{k-1}, z_k), (z_k, z_1)\}$ and $k \geq 5$. Since C is the shortest odd cycle, we have $(z_1, z_4) \notin E(G)$. Due to (P1), there is $U_{z_1,z_4} \in U_G$ containing the vertices z_1 and z_4. Let $x \in U_{z_1,z_4}$ be a vertex, such that $\{x\}$ is recognizing for U_{z_1,z_4}. (a) of Lemma 2.3 implies that x is adjacent to all vertices lying outside U_{z_1,z_4}. Since $z_2 \notin U_{z_1,z_4}$ and $z_3 \notin U_{z_1,z_4}$, we have $(x, z_2) \in E(G)$ and $(x, z_3) \in E(G)$. This is a contradiction since the vertices x, z_2, z_3 induce a triangle.

Thus G is a bipartite graph, and let (X_1, X_2) be the bipartition of G, where $V(G) = X_1 \cup X_2$, $X_1 \cap X_2 = \emptyset$. It is clear that $X_1 \in U_G$ and $X_2 \in U_G$. Since, by assumption $l = 2$, we have $U_G = \{X_1, X_2\}$. This and (P1) imply that for each $x_1 \in X_1$ and $x_2 \in X_2$ we have $(x_1, x_2) \in E(G)$. Thus the graph G is isomorphic to the complete bipartite graph $K_{m,n}$ for some m, n with $m \geq n$.

Now if $n = 1$, then $|V(G)| = m + 1$, $\Delta(G) = m$, and therefore

$$|V(G)| = m + 1 \leq 1 + \left(\frac{1+m}{2}\right)^2 = 1 + \left(\frac{1+\Delta(G)}{2}\right)^2,$$

thus, we can assume that $n \geq 2$. On the other hand, if $m \geq 5$, then $|V(G)| = m + n$, $\Delta(G) = m$, and therefore

$$|V(G)| = m + n \leq 2m \leq 1 + \left(\frac{1+m}{2}\right)^2 = 1 + \left(\frac{1+\Delta(G)}{2}\right)^2,$$

thus, we can assume that $m \leq 4$. Since by assumption G is not isomorphic to $K_{2,2}, K_{3,3}, K_{4,4}$, then G is either $K_{2,3}$ or $K_{2,4}$ or $K_{3,4}$. It is a matter of direct verification that these three graphs satisfy the inequality (2.1).

Case 2. $l \geq 3$ and V_0 is an independent set.

(P1) and Lemma 2.3 imply that there is $w \in V(Q)$ such that $(\{w\} \cup V_0) \in U_G$. Note that all neighbours of w belong to Q and $d(w) = l - 1$. Taking into account that all vertices of V_0 are adjacent to a vertex from Q, we deduce

$$|V_0| \leq (l-1)(\Delta(G) - (l-1)),$$

and therefore

$$|V(G)| = |V_0| + l \leq l + (l-1)(\Delta(G) - (l-1))$$
$$= 1 + (l-1)((\Delta(G)+1) - (l-1)) \leq 1 + \left(\frac{1+\Delta(G)}{2}\right)^2.$$

Case 3. $l \geq 3$ and V_0 is not an independent set.

Let $x, y \in V_0$ such that $(x, y) \in E(G)$. Assume that $V_0 = \{x, y, w_1, \ldots, w_k\}$ $(k \geq 0)$. Choose a vertex $z \in V(Q)$. (P1) and (c) of Lemma 2.3 imply that there is $U_z \in U_G$ such that $\{z\}$ is recognizing for U_z. (a) of Lemma 2.3 implies that $(z, x) \in E(G)$ or $(z, y) \in E(G)$. Taking into account that each vertex of V_0 has a neighbour in Q, we have

$$|V_0| \leq l(\Delta(G) - (l-1)) - (l-2).$$

Let us show that without loss of generality, we can assume that one has equality above. Suppose that $|V_0| < l(\Delta(G) - (l-1)) - (l-2)$. Then $|V_0| \leq l(\Delta(G) - (l-1)) - l + 1$, and therefore

$$|V(G)| = |V_0| + l \leq 1 + l((\Delta(G)+1) - l) \leq 1 + \left(\frac{1+\Delta(G)}{2}\right)^2.$$

Thus, we can assume that $|V_0| = l(\Delta(G) - (l-1)) - (l-2)$. Note that this equality implies:

(1) for each $z \in V(Q)$ we have $d(z) = \Delta(G)$;
(2) the vertices $w_1, \ldots, w_k$ are of degree one;
(3) each vertex $z \in V(Q)$ is adjacent to $\Delta(G) - l$ vertices from $w_1, \ldots, w_k$, and exactly one of x and y.

Since each vertex of V_0 has a neighbour in Q, (2) implies that $V_0 \backslash \{x\}$ is an independent set, thus due to (P1) and (c) of Lemma 2.3, there is $z_0 \in V(Q)$, such that $((V_0 \backslash \{x\}) \cup \{z_0\}) \in U_G$. This particularly means that z_0 is not adjacent to any vertex from $\{w_1, \ldots, w_k\}$. This, (1) and (3) imply that

$$k = 0, |V(G)| = l + 2 \text{ and } \Delta(G) = l.$$

Taking into account that $l \geq 3$, we deduce

$$|V(G)| = l + 2 = 1 + (l+1) \leq 1 + \left(\frac{1+l}{2}\right)^2 = 1 + \left(\frac{1+\Delta(G)}{2}\right)^2.$$

The proof of the Theorem 2.4 is completed. □

Remark 2.5. There is an infinite sequence of graphs attaining the bound of Theorem 2.4. For a positive integer n consider the graph G from Figure 1. Note that $|V(G)| = 1 + n^2, \Delta(G) = 2n - 1$ and $c(\mathcal{U}_G) = \frac{1}{n^2-2n+2}$.

Theorem 2.6. *For any $m, n \in N$ with $1 \leq m \leq n$ there is a connected bipartite graph G such that $c(\mathcal{U}_G) = \frac{m}{n}$.*

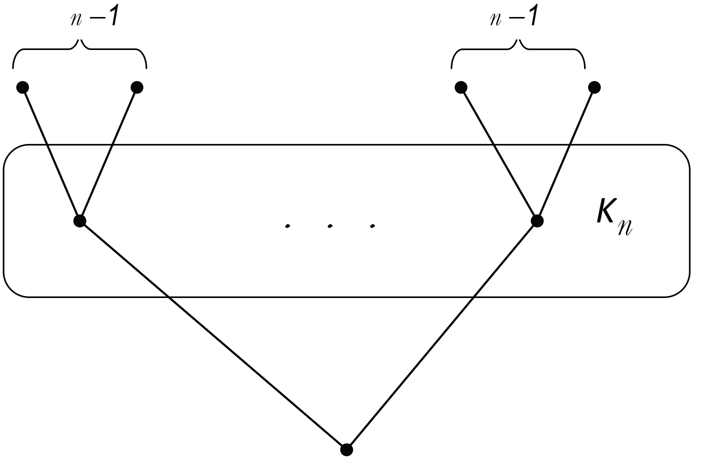

Fig. 1. Example attaining the bound of Theorem 2.4

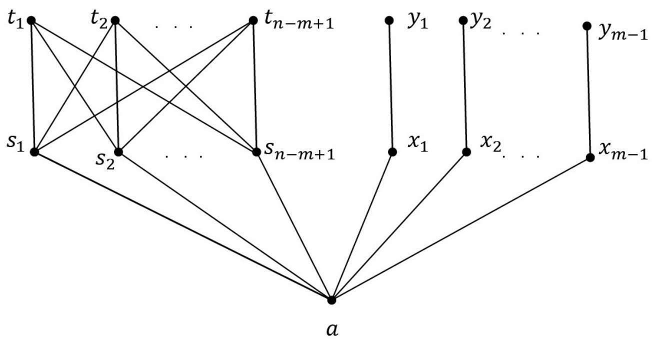

Fig. 2. A graph G with $c(\mathcal{U}_G) = \frac{m}{n}$

Proof. For any $m, n \in N$ with $1 \le m \le n$ consider the connected bipartite graph G from Figure 2.

Define

$$S = \{s_1, \dots, s_{n-m+1}\}, \qquad T = \{t_1, \dots, t_{n-m+1}\},$$
$$X = \{x_1, \dots, x_{m-1}\}, \qquad Y = \{y_1, \dots, y_{m-1}\}.$$

Let us show that $c(\mathcal{U}_G) = \frac{m}{n}$. Choose any $U \in U_G$. We will consider two cases.

Case 1. $a \in U$.

Clearly, for each $s \in S, s \notin U$ and for each $x \in X, x \notin U$, therefore $U = \{a\} \cup T \cup Y$. Lemma 2.1 implies that $S_U = \{a\}$ is a smallest recognizing subset for U, thus

$$c(U) = \frac{|S_U|}{|U|} = \frac{1}{n+1}.$$

Case 2. $a \notin U$.

It is clear that

$$|\{x_i, y_i\} \cap U| = 1 \quad \text{for} \quad i = 1, \dots, m-1; \tag{2.2}$$

$$T \cap U = \emptyset \Leftrightarrow S \cap U = S \Leftrightarrow S \subseteq U; \tag{2.3}$$

$$S \cap U = \emptyset \Leftrightarrow T \cap U = T \Leftrightarrow T \subseteq U; \tag{2.4}$$

(2.2)–(2.4) imply that $|U| = n$.

Now, let S_U be any smallest recognizing subset of U. Note that if there is $x_i \in U$, then x_i, with respect to y_i and U, satisfies the conditions of Corollary 2.2, thus $x_i \in S_U$. Similarly, if there is $y_i \in U$ then y_i, with respect to x_i and U, satisfies the conditions of Corollary 2.2 (as $a \notin U$), thus $y_i \in S_U$.

On the other hand, if $S \subset U$ then due to (2.3) $T \cap U = \emptyset$, hence Lemma 2.1 implies that there is $s \in S$ such that $s \in S_U$. Similarly, if $T \subset U$ then there is $t \in T$ such that $t \in S_U$. This implies that either there is $s \in S$ such that $(X \cap U) \cup (Y \cap U) \cup \{s\} \subseteq S_U$ or there is $t \in T$ such that $(X \cap U) \cup (Y \cap U) \cup \{t\} \subseteq S_U$. Now, it is not hard to see that either $(X \cap U) \cup (Y \cap U) \cup \{s\}$ or $(X \cap U) \cup (Y \cap U) \cup \{t\}$ is recognizing for U, hence $(X \cap U) \cup (Y \cap U) \cup \{s\} = S_U$ or $(X \cap U) \cup (Y \cap U) \cup \{t\} = S_U$, and therefore

$$|S_U| = |X \cap U| + |Y \cap U| + 1 = m,$$

since, due to (2.2), we have $|X \cap U| + |Y \cap U| = m - 1$. Thus

$$c(U) = \frac{|S_U|}{|U|} = \frac{m}{n}.$$

The considered two cases imply

$$c(\mathcal{U}_G) = \max\left\{\frac{1}{n+1}, \frac{m}{n}\right\} = \frac{m}{n}.$$

The proof of the Theorem 2.6 is completed. □

In the end of the section we study the hardness of clutters $\mathcal{U}_T$ arising from trees T. Our goal is to try to characterize the class of trees T, for which $c(\mathcal{U}_T) = 1$. Though we fail to achieve this, we are able to present some non-trivial necessary and sufficient conditions.

Definition 2.7. In a tree T a vertex $t \in V(T)$ is

(a) α-vertex if there is $t' \in V(T)$ with $d(t') = 1$ and $\rho(t, t') = 2$;

(b) β-vertex if it is adjacent to an α-vertex, whose all neighbours that differ from t, are α-vertices;
(c) γ-vertex if it is adjacent to a β-vertex;
(d) β-vertex if it is adjacent to an α-vertex, whose all neighbours that differ from t, are α or γ-vertices;
(e) δ-vertex if all its neighbours are α or γ-vertices;

Remark 2.8. By definition, a vertex of a tree can satisfy more than one of conditions of Definition 2.7, and thus be of more than one type.

Remark 2.9. The definition has a recursive structure, and in (c), in the definition of a γ-vertex, a β-vertex is understood as one which is defined by (b) or (d). For the sake of clear explanation and to prove the next lemma, we will imagine that our definition works as a labeling algorithm. The algorithm for its input gets a tree. During the initialization it labels all α-vertices according to (a) of Definition 2.7. Then at the first step it labels all β-vertices and their neighbour γ-vertices according to (b) and (c) of Definition 2.7, respectively. If at the k^{th} step, the labeling is already done, then in the $(k+1)^{th}$ step it labels all β-vertices and their neighbour γ-vertices according to (d) and (c) of Definition 2.7, respectively. The process continues until no new vertex receives a label. Finally, in the last step, the algorithm labels all δ-vertices according to (e) of Definition 2.7 and presents the labeling of the input tree as the output.

Remark 2.10. By definition, every β-vertex of a tree is a δ-vertex, therefore it is natural to introduce the following definition

Definition 2.11. A δ-vertex is called pure if it is not a β-vertex.

The following lemma explains the essence of Definition 2.7.

Lemma 2.12. *Let T be a tree. Suppose that $U \in U_T$ and $c(U) = 1$. Then:*

(1) *all α-vertices do not belong to U;*
(2) *all β-vertices belong to U;*
(3) *all γ-vertices do not belong to U;*
(4) *all δ-vertices belong to U.*

Proof. (1) Suppose that t is an α-vertex. Then, due to (a) of Definition 2.7, there is $t' \in V(T)$ with $d(t') = 1$ and $\rho(t, t') = 2$. If $t \in U$, then the only neighbour of t', which is also a neighbour of t, does not lie in U, hence $t' \in U$ as $U \in U_T$. Now, observe that $U \backslash \{t'\}$ is a recognizing set for U, since it trivially satisfies the condition of Lemma 2.1. This implies that

$$c(U) \leq \frac{|U| - 1}{|U|} < 1,$$

which is a contradiction.

(2),(3) We will give a simultaneous proof of (2) and (3) by induction on k, where k is the current step of the labeling algorithm (Remark 2.9).

So, assume that $k = 1$, t is a β-vertex and it "became" such a one due to (b) of Definition 2.7. Let us show that $t \in U$.

According to (b) of Definition 2.7, there is an α-vertex t', all of whose neighbours except t, are α-vertices. Due to (1) of Lemma 2.12, neither t' nor its α-neighbours that differ from t, belong to U. Since $U \in U_T$, we deduce $t \in U$.

This implies that all γ-vertices that are adjacent to a β-vertex that was labeled in the first step, do not belong to U. Thus (2) and (3) are true for $k = 1$.

Now, assume that (2) and (3) are true for vertices which receive their labels in the steps up to k. Consider a β-vertex t which gets its label according to (d) of Definition 2.7 in the $(k+1)^{th}$ step of the labeling algorithm. Let us show that $t \in U$.

According to (d) of Definition 2.7, there is an α-vertex t', whose all neighbours except t, are α or γ-vertices, which have received their labels earlier than the $(k+1)^{th}$ step. Due to the induction hypothesis and (1) of Lemma 2.12, neither t' nor its α or γ-neighbours that differ from t, belong to U. Since $U \in U_T$, we deduce $t \in U$.

This implies that all γ-vertices that are adjacent to a β-vertex that was labeled in the $(k+1)^{th}$ step, do not belong to U. Thus (2) and (3) are true for $k+1$ and the proof is completed.

(4) If t is a δ-vertex, then due to (e) of Definition 2.7, and (1) and (3) of Lemma 2.12, all the neighbours of t do not belong to U, hence $t \in U$ as $U \in U_T$.

The proof of the Lemma 2.12 is completed. □

The proved lemma implies the following necessary condition for a tree T to satisfy $c(\mathcal{U}_T) = 1$.

Corollary 2.13. *If T is a tree with $c(\mathcal{U}_T) = 1$, then:*

(a) *there is no α or γ-vertex, which is also a β or a δ-vertex;*
(b) *each δ-vertex t is adjacent to an α or a γ-vertex, that has a neighbour that is different from t and which is neither a β nor a δ-vertex.*

Proof. (a) is clear.

(b) On the opposite assumption, consider a δ-vertex t, all of whose neighbours are α or γ-vertices ((e) of Definition 2.7), and whose every neighbour that is different from t is adjacent to a β or a δ-vertex. Due to Lemma 2.12, the vertex t and these β or δ-vertices lying on a distance two from t belong to any $U \in U_T$ with $c(U) = 1$. Now, note that $U \backslash \{t\}$ is a recognizing set for U, since it trivially satisfies the condition of Lemma 2.1. This implies that

$$c(U) \leq \frac{|U| - 1}{|U|} < 1,$$

which is a contradiction. □

Theorem 2.14. *If a tree T contains neither a β nor a pure δ-vertex, then for each $u \in V(T)$ with $d(u) = 1$ there is $U \in U_T$ with $c(U) = 1$ and $u \in U$.*

Proof. Unfortunately, the proof of existence of such $U \in U_T$ is not easy. This is the main reason that we will give an algorithmic construction of such $U \in U_T$.

Given $u \in V(T)$ with $d(u) = 1$, we will assume that T is represented as a tree rooted at u.

Step 0.

$$U := \{u\}, \quad Spec := \{\text{the neighbours of } u\}$$

Consider the sets $B_1, \ldots, B_k$ of vertices lying at a distance three from u, where it is assumed that the vertices of $B_j, 1 \leq j \leq k$ are adjacent to the same vertex. Let $List$ be a list comprised of the sets $B_1, \ldots, B_k$. Note that since T does not contain a β-vertex, we have that all of $B_1, \ldots, B_k$ contain a non-α vertex.

Step 1. While $List \neq \emptyset$

remove the first element B of $List$.

Define $A = \{v \in B : v \text{ is not a } \alpha\text{-vertex}\}$

$A' = \{v \in A : \text{ all children of } v \text{ are } \alpha\text{-vertices}\}$

Case 1. $A' \neq \emptyset$

$U := U \cup A'$

Add all children of vertices from A' (which are α -vertices, by definition) to the set $Spec$.

Note that, by definition of A', for each $w \in A \backslash A'$ the set B_w, which is the set of children of w, contains a non-α vertex. Moreover, for each $z \in A'$ if we consider the sets $B_{z_1}, \ldots, B_{z_s}$ of vertices lying at a distance three from z (the vertices of $B_{z_j}, 1 \leq j \leq s$ are adjacent to the same vertex), then since T contains no δ-vertex, each of these sets contains a non-α vertex.

Add all $B_w, B_{z_1}, \ldots, B_{z_s}$ to $List$;

Case 2. $A' = \emptyset$

Take any $w \in A$.

$U := U \cup \{w\}$; add the parent x of w to the set $Spec$.

Note that $A' = \emptyset$ implies that for each $y \in B \backslash \{w\}$ the set B_w of children of y contains a non-α vertex. On the other hand, since T contains no β-vertex, then for each $z \in B \backslash A$ the set B_z of children of z contains a non-α vertex.

Add all B_w, B_z to $List$;

Consider the sets B_i of vertices lying at a distance three from w, where it is assumed that B_i is the set of children of z_i.

Case 2.1. B_i contains a non-α vertex;

Add B_i to $List$;

Case 2.2. All vertices of B_i are α-vertices;

$U := U \cup \{z_i\}$; $Spec := Spec \cup B_i$;

Consider the sets $B_{z_1}^{(i)}, \ldots, B_{z_s}^{(i)}$ of vertices lying at a distance three from z, where we assume that $B_{z_j}^{(i)}$ coincides with the set of children of a vertex ${z_j}^{(i)}$. Since T contains no δ-vertex, then each $B_{z_j}^{(i)}$ contains a non-α vertex.

Add $B_{z_1}^{(i)}, \ldots, B_{z_s}^{(i)}$ to $List$;

The description of the algorithm is completed.

Let us note that if the algorithm cannot choose the set A then the last vertex from which it is impossible to choose a vertex lying on a distance three, is either a pendant vertex, which has a specific vertex in the set *Spec*, or is a vertex that is adjacent to a pendant vertex, and this pendant vertex will be the specific vertex for it.

It can be easily seen that the algorithm constructs a maximal independent set U of T containing the vertex u. The construction of the set *Spec* implies that each vertex $v \in U$ has a specific neighbour in *Spec*, that is, a neighbour, which is not adjacent to any other vertex of U. This and Corollary 2.2 imply that the hardness of U is one. The proof of Theorem 2.14 is complete. □

Remark 2.15. The Theorem 2.14 presents merely a sufficient condition. The trees from Figure 3 contain a pure δ-vertex, do not contain a β vertex, and nevertheless, the first of them has a hardness that is less than one, while the second one is of hardness one. On the other hand, the trees from Figure 4 contain a β-vertex, do not contain a pure δ vertex, and nevertheless, the first of them has a hardness that is less than one, while the second one is of hardness one.

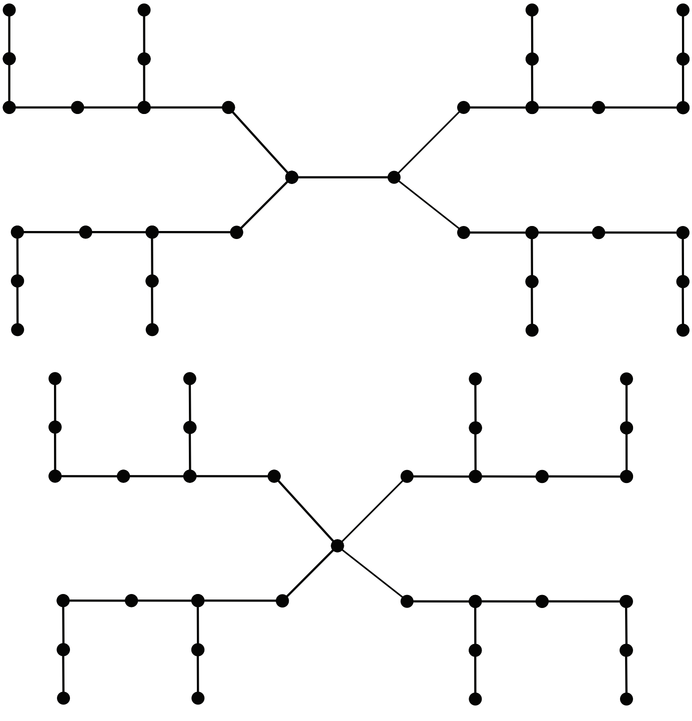

Fig. 3. Trees with pure δ-vertices, without β-vertices

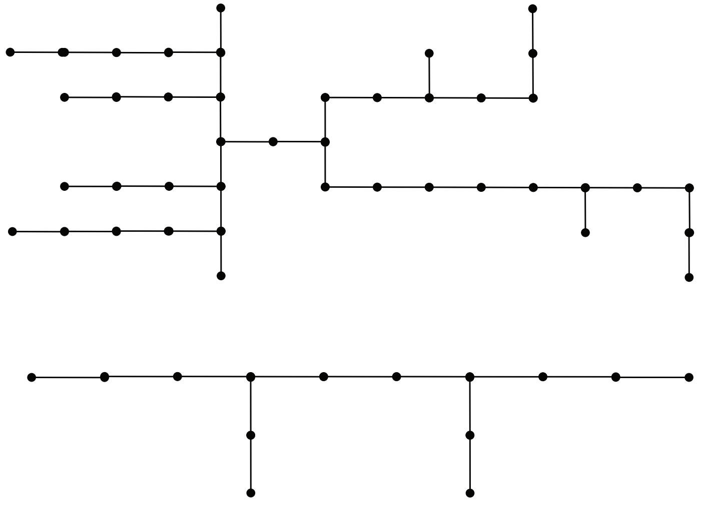

Fig. 4. Trees with β-vertices, without pure δ-vertices

3. THE HARDNESS OF $\mathcal{M}_G$

Below we investigate the hardness of the clutter $\mathcal{U}_G$ in the class of line graphs G. This class is interesting not only for its own sake, but also for its connection with another clutter related to graphs. Taking into account, that the clutter U_G of a line graph G coincides with the clutter $\mathcal{M}_H$ of some graph H, in this sections we will directly work with the latter clutter without remembering that it originated from a line graph.

3.1. STRUCTURAL LEMMAS

Lemma 3.1. *Assume that* $H \in \mathcal{M}_G$ *and let* S_H *be any smallest recognizing subset of* H*. Then:*

1. *the vertices of* $V(H \backslash S_H)$ *can only be connected to the vertices of* $V(S_H)$*;*
2. *each edge in* S_H *has at least one endpoint connected to a vertex not in* $V(S_H)$*.*

Proof. Let $e = (u, v)$ be an edge in $H \backslash S_H$. Let us first prove that both u and v are not connected to vertices, which are not covered by H. If this is not true, then without loss of generality we may assume that there exists $p \in V(G) \backslash V(H)$ such that $(p, u) \in E(G)$. $H \cup \{(p, u)\} \backslash \{(u, v)\}$ is a maximal matching containing S_H. This contradicts the definition of S_H.

We have proven that the vertices of $V(H \backslash S_H)$ can only be connected to the vertices of $V(H)$.

Now, if there are there are vertices $\{u_1, u_2, u_3, u_4\}$, such that

$$(u_1, u_2), (u_3, u_4) \in H \backslash S_H$$

and $(u_2, u_3) \in E(G)$, then there is a maximal matching that contains $(H\backslash\{(u_1, u_2), (u_3, u_4)\}) \cup \{(u_2, u_3)\}$. That maximal matching is different from H and contains S_H. This is a contradiction proving point 1.

If the statement of point 2 does not take place for an edge e, then every maximal matching, which contains $S_H\backslash\{e\}$ also contains S_H. Thus H is the only maximal matching, which contains $S_H\backslash\{e\}$, and consequently S_H is not a minimum subset of H with this property. The contradiction proves point 2. □

Lemma 3.2. *Suppose H is a smallest maximal matching in G and $e \in H$. The endpoints of e cannot be connected to endpoints of different edges of $H\backslash S_H$, where S_H is any smallest recognizing subset of H.*

Proof. Let (u, v) be an edge in S_H. If there are edges (u_1, v_1) and (u_2, v_2) from $H\backslash S_H$, such that u is connected to u_1 and v is connected to u_2, then H is not a smallest maximal matching since the cardinality of

$$H \cup \{(u, u_1), (v, u_2)\}\backslash\{(u, v), (u_1, v_1), (u_2, v_2)\}$$

is less than that of H. □

Also, recall the following result [5, 6]:

Lemma 3.3. *If G is a connected graph, whose every maximal matching is a perfect matching, then G is either K_{2n} or $K_{n,n}$.*

3.2. A LOWER BOUND FOR HARDNESS

Note that the hardness of $\mathcal{M}_G$ for disconnected graphs G does not have a lower bound better than zero. For instance, for the graph K that consists of a single matching, we have $c(\mathcal{M}_K) = 0$. Moreover, it can be shown that for every rational number r, $0 \leq r \leq 1$ there exists a graph with hardness r. To construct one just consider the graph G_r from Figure 5, where we assume that $r = \frac{a+1}{b+1}$.

The following theorem proves a tight lower bound for the hardness of $\mathcal{M}_G$ in the class of connected graphs G. Before we move on, let us note that the bound given in the theorem below, is significantly better than the one that Theorem 2.4 provides.

Theorem 3.4. *For every connected graph G with $|V(G)| > 4$, we have $c(\mathcal{M}_G) \geq \frac{2}{|V(G)|-2}$.*

Proof. Let H be a smallest maximal matching of G, and let S_H be any smallest recognizing subset of H. If $|H| < \lfloor |V|/2 \rfloor$, then $|H| \leq \frac{|V|-2}{2}$ and

$$c(\mathcal{M}_G) \geq c(H) = \frac{|S_H|}{|H|} \geq \frac{1}{(|V|-2)/2} = \frac{2}{|V|-2}. \tag{3.1}$$

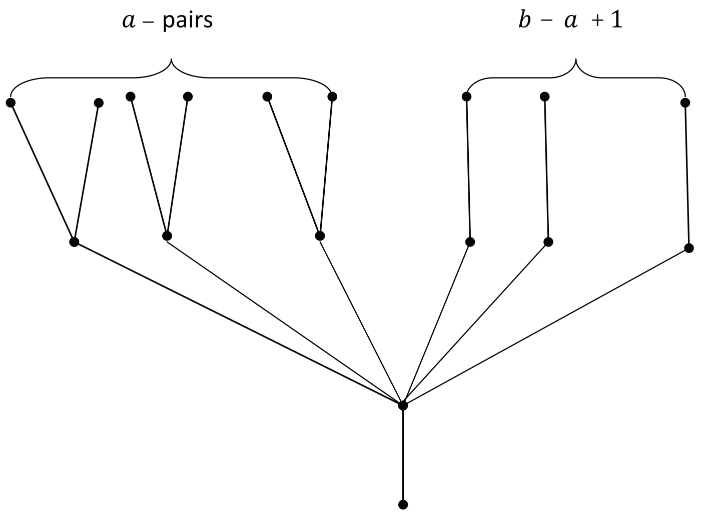

Fig. 5. A graph G with $c(\mathcal{M}_G) = r$

If $|H| = \lfloor |V|/2 \rfloor$, then there are two cases.

- $|V|$ is even. Since H is a smallest maximal matching, every maximal matching of G is a perfect matching. Due to Lemma 3.3, G is isomorphic to either K_{2n} or $K_{n,n}(n = |V|/2 > 2)$. For these graphs
$$c(\mathcal{M}_G) = \frac{n-1}{n} > \frac{1}{n-1} = \frac{2}{|V|-2}.$$
- $|V|$ is odd. If $|S_H| \geq 2$, then
$$c(\mathcal{M}_G) \geq 2/|H| = 4/(|V|-1) \geq 2/(|V|-2). \tag{3.2}$$
Assume $S_H = \{(u,v)\}$. Lemma 3.1 implies that either $|H| = 2(|V| = 5)$ or all the vertices of $V(H \backslash S_H)$ are connected to only one of the endpoints of (u,v). Without loss of generality we may assume that they are connected to u.
If $|V| = 5$, there are only a few graphs for which it is possible to have $|H| = 2$ and $|S_H| = 1$. All these graphs G can be easily checked to satisfy $c(\mathcal{M}_G) = 1$.
Assume $|H| \geq 3$. Let w be the vertex, which is not covered by H. If w is connected to v then due to 1 of Lemma 3.1, we have $|S_{H \cup \{(v,w)\} \backslash \{(u,v)\}}| > 1$, since all the edges of $H \cup \{(v,w)\} \backslash \{(u,v)\}$ are connected to u. As a result, according to (3.2),
$$c(\mathcal{M}_G) > 2/(|V|-2).$$
If w is connected to u, take an edge $(u_1, v_1) \in H$ such that $(u, u_1) \in E$. $(H \cup \{(u,u_1)\}) \backslash \{(u,v),(u_1,v_1)\}$ is a maximal matching with a smaller cardinality than H. Thus H is not smallest and this case is impossible.

The proof is now complete. □

Figure 5 with $a = 0$ illustrates that the bound achieved in the previous theorem is tight. The depicted graph G contains $2(b+2)$ vertices and it satisfies $c(\mathcal{M}_G) = 1/(b+1)$, therefore

$$c(\mathcal{M}_G) = \frac{2}{|V(G)| - 2}.$$

3.3. BOUNDS FOR $c(\mathcal{M}_G)$ IN THE CLASS OF REGULAR GRAPHS G

For regular graphs G, it is possible to find lower bounds for $c(\mathcal{M}_G)$ that do not depend on the number of edges in those graphs.

Theorem 3.5. *For an r-regular graph G with $r > 1$ we have $c(\mathcal{M}_G) \geq \frac{1}{2}$.*

Proof. Take any $H \in \mathcal{M}_G$, and let S_H be any smallest recognizing subset of H. Let E_1 be the set of edges that connect $V(S_H)$ with $V(H \backslash S_H)$, E_2 be the set of edges that connect $V(S_H)$ with $V(G) \backslash V(H)$, and E_3 be the set of edges in the spanning subgraph of $V(S_H)$, not including the edges from S_H.
According to point 1 of Lemma 3.1, all the vertices of $V(H \backslash S_H)$ are only connected to the vertices of $V(S_H)$. Therefore,

$$2|S_H|(r-1) = |V(S_H)|(r-1) = \sum_{v \in V(S_H)} (d(v) - 1) = |E_1| + |E_2| + 2|E_3| \geq |E_1|$$
$$= \sum_{v \in V(H \backslash S_H)} (d(v) - 1) = (r-1)|V(H \backslash S_H)| = 2|H \backslash S_H|(r-1).$$

Since $r \neq 1$, we have $|S_H| \geq |H \backslash S_H|$, thus $c(H) = |S_H|/|H| \geq \frac{1}{2}$, and therefore $c(\mathcal{M}_G) \geq \frac{1}{2}$. □

Corollary 3.6. *If G is a regular graph and $c(\mathcal{M}_G) = \frac{1}{2}$, then for every maximal matching H, $c(H) = \frac{1}{2}$.*

Corollary 3.7. *If G is a regular graph and $c(\mathcal{M}_G) = \frac{1}{2}$, then every maximal matching is a perfect matching.*

Proof. Since $c(H) = \frac{1}{2}$, we have that $|S_H| = |H \backslash S_H|$, and therefore $E_2 = \emptyset$. Now suppose there is vertex v, which is not covered by H. As H is maximal, it covers all the neighbors of v. Due to 1 of Lemma 3.1, these neighbors cannot belong to $V(H \backslash S_H)$; consequently, they belong to $V(S_H)$. This contradicts with E_2 being empty. □

Corollary 3.8. *The hardness of $\mathcal{M}_G$ for a connected regular graph G equals $\frac{1}{2}$ if and only if G is K_4 or $K_{2,2}$.*

Proof. It is not hard to see that $c(\mathcal{M}_{K_{2n}}) = c(\mathcal{M}_{K_{n,n}}) = \frac{n-1}{n}$. That said, the corollary follows from Lemma 3.3 and Corollary 3.7. □

The following theorem shows that there exist better bounds for the complexities of $\mathcal{M}_G$ for regular graphs G, if we do not consider graphs of small regularity.

Theorem 3.9. *For an r-regular graph G, we have*

(a) *If $r > 4$, then $c(\mathcal{M}_G) \geq \frac{2}{3}$;*
(b) *If $r = 4$, then $c(\mathcal{M}_G) > \frac{3}{5}$.*

Proof. (a) Due to Lemma 3.1, for each $(u, v) \in S_H$ there are two options:

- u and v can be connected to the endpoints of only one edge from $H \backslash S_H$.
- u is not connected to any vertex covered by $H \backslash S_H$ and v may be connected to any number of endpoints of edges from $H \backslash S_H$.

Therefore, the edges of S_H are divided into two categories. Let A denote the set of edges of the first category, and B the set of the edges of the second category. If an edge from S_H falls in both categories, we will consider it to be in category A and not B.

Retaining the notations of the proof of Theorem 3.5, we have $|E_1| = 2(r-1)|H\backslash S_H|$. The endpoints of each edge in category A are the endpoints of at most 4 edges from $|E_1|$. The endpoints of each edge in category B are the endpoints of at most $r - 1$ edges of E_1. This implies

$$|E_1| \leq 4|A| + (r-1)|B| = (r-1)|S_H| - (r-5)|A| \leq (r-1)|S_H|.$$

We got that $2|H\backslash S_H| \leq |S_H|$, hence,

$$c(\mathcal{M}_G) \geq c(H) = \frac{|S_H|}{|H|} \geq \frac{2}{3}.$$

(b) We will assume that G is connected, because the case of disconnected graphs easily follows from the case of connected graphs. Choose any smallest maximal matching H of G.

Note that (2) of Lemma 3.1 implies that if $e = (u, v) \in A$ then $u_1 = v_1$ or $(u_1, v_1) \in H\backslash S_H$. Moreover, $S_H = A \cup B$, $A \cap B = \emptyset$, and

$$|E_1| = 2(r-1)|H\backslash S_H| = 6|H\backslash S_H|.$$

The endpoints of each edge in category A are the endpoints of at most 4 edges from E_1, while the endpoints of each edge in category B are the endpoints of at most 3 edges of E_1. This implies

$$6|H\backslash S_H| = |E_1| \leq 4|A| + 3|B| \leq 4|A| + 4|B| = 4|S_H|,$$

or

$$6|H| \leq 10|S_H|,$$

and therefore

$$c(H) = \frac{|S_H|}{|H|} \geq \frac{3}{5}. \tag{3.3}$$

Now, we claim that $c(H) > \frac{3}{5}$. If $c(H) = \frac{3}{5}$, then

$$|E_1| = 4|S_H| = 4|A|,$$

and therefore $B = \emptyset$. This implies that for each $e = (u, v) \in S_H$ there is exactly one $f = (u_1, v_1) \in H \backslash S_H$ such that

$$\{(u, u_1), (u, v_1), (v, u_1), (v, v_1), \} \subseteq E_1.$$

The uniqueness of f follows from Lemma 3.2. Note that this correspondence is one-to-one since G is 4-regular and an edge from $H \backslash S_H$ cannot be connected to two different edges from A. Thus,

$$|H| = |S_H|,$$

and

$$c(H) = \frac{|S_H|}{|H|} = \frac{1}{2} < \frac{3}{5},$$

contradicting (3.3). The proof is now complete. □

Note that the bound from (a) of the previous theorem is reachable, since K_6 is a 5-regular graph with $c(\mathcal{M}_{K_6}) = \frac{2}{3}$.

Our interest toward the hardness and particularly, the hardness of clutters arising from regular graphs was motivated by the following conjecture.

Conjecture 3.10. *If G is a connected regular graph with $c(\mathcal{M}_G) < 1$, then G is either isomorphic to C_7, or there is $n, n \geq 1$ such that G is isomorphic either to $K_{n,n}$ or to K_{2n}, where C_7 is the cycle of length seven.*

In some sense, our conjecture states that all regular structures are "hard" except some "uninteresting" cases.

4. COMPUTATIONAL COMPLEXITY RESULTS FOR HARDNESS

The aim of this section is the investigation of some problems that are related to the algorithmic computation of the hardness of $\mathcal{U}_G$.

We start with a problem that is related to finding a recognizing set for a given maximal independent set.

Problem 1:
Condition: Given a graph G, $U \in U_G$ and a positive integer k.
Question: Is there a recognizing set $U' \subseteq U$ for U with $|U'| = k$?

Theorem 4.1. *Problem 1 is NP-complete already for bipartite graphs.*

Proof. Lemma 2.1 implies that Problem 1 belongs to the class NP. To show the completeness of the problem, we will reduce the classical Set Cover problem to our problem restricted to bipartite graphs. Recall that the Set Cover is formulated as follows ([3]):

Problem: Set Cover
Condition: Given a set $A = \{a_1, \ldots, a_n\}$, a family $\mathcal{A} = \{A_1, \ldots, A_m\}$ of subsets of the set A with $A_1 \cup \ldots \cup A_m = A$, and a positive integer $l, l \leq m$.

Question: Are there $A_{i_1}, \ldots, A_{i_l} \in \mathcal{A}$ with $A_{i_1} \cup \ldots \cup A_{i_l} = A$?

For instance I of Set Cover consider the graph $G_I = (V, E)$, where

$$V = \{a_1, \ldots, a_n, A_1, \ldots, A_m\}, \quad E = \{(a_i, A_j) : a_i \in A_j, 1 \le i \le n, 1 \le j \le m\}.$$

Note that G_I is bipartite. Consider the set $U = \{A_1, \ldots, A_m\}$. Since $A_1 \cup \ldots \cup A_m = A$, we have $U \in U_{G_I}$.

It can be easily verified that the set U has a recognizing subset comprised of l elements if and only if there are $A_{i_1}, \ldots, A_{i_l} \in \mathcal{A}$ with $A_{i_1} \cup \ldots \cup A_{i_l} = A$. The proof of the theorem is complete. □

Now, we are turning to the investigation of the computation of $c(\mathcal{U}_G)$. Consider the following

Problem 2:
Condition: Given a graph G and positive integers k, m with $1 \le k \le m$.
Question: Does the inequality $c(\mathcal{U}_G) \le \frac{k}{m}$ hold?

Theorem 4.2. *Problem 2 is NP-hard already for bipartite graphs.*

Proof. We will reduce Set Cover to our problem restricted to bipartite graphs. Given an instance I of Set Cover, consider the graph $G_I = (V, E)$, where

$$V = \{A_1, \ldots, A_m\} \cup \{a_i^{(k)} : 1 \le i \le n, 1 \le k \le (n+m)^2\},$$
$$E = \{(a_i^{(k)}, A_j) : a_i \in A_j, 1 \le i \le n, 1 \le j \le m, 1 \le k \le (n+m)^2\}.$$

Note that G_I is bipartite. Let us show that

$$c(\mathcal{U}_{G_I}) = \frac{l_{min}}{m},$$

where l_{min} denotes the size of minimum cover of A, that is, the minimum number l_{min} for which there are $A_{i_1}, \ldots, A_{i_{l_{min}}} \in \mathcal{A}$ with $A_{i_1} \cup \ldots \cup A_{i_{l_{min}}} = A$.

Choose any $U \in U_{G_I}$. We will consider two cases.

Case 1. $U = \{A_1, \ldots, A_m\}$.

Lemma 2.1 and the definition of G_I imply that $|S_U| = l_{min}$, therefore

$$c(U) = \frac{l_{min}}{m}.$$

Case 2. $U \neq \{A_1, \ldots, A_m\}$.

Suppose that $U \cap \{A_1, \ldots, A_m\} = \{A_{i_1}, \ldots, A_{i_r}\}$. Since $U \neq \{A_1, \ldots, A_m\}$, we imply that $A_{i_1} \cup \ldots \cup A_{i_r} \neq A$. Assume that there are $r', r' \ge 1$ elements of A that do not belong to either of A_{i_j}'s. Note that all $r'(n+m)^2$ copies of these r' elements belong to U, and

$$|U| = r + r'(n+m)^2.$$

On the other hand, if we consider the set $U' \subseteq U$, where

$$U' = \{A_{i_1}, \ldots, A_{i_r}\} \cup \{a_i^{(1)} : a_i \text{ does not belong to either of } A_{i_j}\text{'s}\},$$

then, according to Lemma 2.1, this would be a recognizing set for U, therefore

$$c(U) = \frac{|S_U|}{|U|} \leq \frac{|U'|}{|U|} = \frac{r + r'}{r'(n+m)^2} \leq \frac{n+m}{(n+m)^2} = \frac{1}{n+m} < \frac{1}{m} \leq \frac{l_{min}}{m}.$$

The considered two cases imply $c(\mathcal{U}_{G_I}) = \frac{l_{min}}{m}$. Now, it is not hard to verify that in the instance I of Set Cover, there is a cover of length l, if and only if $l_{min} \leq l$, which is equivalent to $c(\mathcal{U}_{G_I}) \leq \frac{l}{m}$. The proof of the theorem is complete. □

In the end of the paper, let us note that we have failed to achieve similar results for the clutters $\mathcal{M}_G$. We leave the investigation of the computational complexity of the calculation of $c(\mathcal{M}_G)$ as a research problem.

Acknowledgments
We thank the referees for their comments that helped us to improve the presentation of the paper.
The authors are supported by a grant of Armenian National Science and Education Fund

REFERENCES

[1] J.C. Claussen, *Offdiagonal complexity: A computationally quick complexity measure for graphs and networks*, Physica A **375** (2007), 365–373.

[2] G. Cornuejols, *Combinatorial Optimization: Packing and Covering*, SIAM (January, 2001).

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

[4] S. Jukna, *On graph complexity*, ECCC Report, No. 5, 2004.

[5] L. Lovász, M.D. Plummer, *Matching theory*, Ann. Discrete Math. **29** (1986).

[6] D.P. Sumner, *Randomly matchable graphs*, J. Graph Theory **3** (1979), 183–186.

[7] D.B. West, *Introduction to Graph Theory*, Prentice-Hall, Englewood Cliffs, 1996.

Sasun Hambardzumyan
xustup@gmail.com

Department of Informatics and Applied Mathematics
Yerevan State University, Yerevan, 0025, Armenia

Vahan V. Mkrtchyan (corresponding author)
vahanmkrtchyan2002@ysu.am

Institute for Informatics and Automation Problems
National Academy of Sciences of Republic of Armenia
0014, Armenia

Vahe L. Musoyan
vmusoyan@cs.stanford.edu
vahe.musoyan@gmail.com

Stanford University
Computer Science Department
Stanford, CA 94305, USA

Hovhannes Sargsyan
hsargsian@gmail.com

Department of Informatics and Applied Mathematics
Yerevan State University, Yerevan, 0025, Armenia

Received: March 30, 2014.
Revised: December 1, 2015.
Accepted: December 17, 2015.