

ON INDEPENDENT $[1, 2]$ -SETS IN TREES¹

SAHAR A. ALEID, JOSÉ CÁCERES

AND

MARÍA LUZ PUERTAS

Department of Mathematics
University of Almeria, Spain

e-mail: sahar.aleid@ual.es
jcaceres@ual.es
mpuertas@ual.es

Abstract

An $[1, k]$ -set S in a graph G is a dominating set such that every vertex not in S has at most k neighbors in it. If the additional requirement that the set must be independent is added, the existence of such sets is not guaranteed in every graph. In this paper we solve some problems previously posed by other authors about independent $[1, 2]$ -sets. We provide a necessary condition for a graph to have an independent $[1, 2]$ -set, in terms of spanning trees, and we prove that this condition is also sufficient for cactus graphs. We follow the concept of excellent tree and characterize the family of trees such that any vertex belongs to some independent $[1, 2]$ -set. Finally, we describe a linear algorithm to decide whether a tree has an independent $[1, 2]$ -set. This algorithm can be easily modified to obtain the cardinality of a smallest independent $[1, 2]$ -set of a tree.

Keywords: domination, independence, spanning trees, excellent trees.

2010 Mathematics Subject Classification: 05C69.

1. INTRODUCTION

All graphs considered here are finite, undirected, simple and connected. Undefined basic concepts can be found in introductory graph theory literature as in [2, 8]. Let $G = (V, E)$ be a graph. The *open neighborhood* of a vertex

¹Partially supported by Phoenix Project (Erasmus Mundus Programme), MTM2014-60127-P (Ministerio de Economía y Competitividad) and FQM305 (Junta de Andalucía).

$v \in V$ is the set $N(v) = \{u \mid uv \in E\}$ of vertices adjacent to v . Each vertex $u \in N(v)$ is called a *neighbor* of v . The *closed neighborhood* of a vertex $v \in V$ is the set $N[v] = N(v) \cup \{v\}$. The open neighborhood of a set $S \subseteq V$ of vertices is $N(S) = \bigcup_{v \in S} N(v)$, while the closed neighborhood of a set S is $N[S] = \bigcup_{v \in S} N[v]$. A set S is *independent* if no two vertices in S are adjacent. A set S is a *dominating set* of a graph G if $N[S] = V$, that is, for every $v \in V$, either $v \in S$ or $v \in N(u)$ for some vertex $u \in S$. A dominating set which is also independent is an *independent dominating set*.

In [3], Chellali *et al.* define a subset $S \subseteq V$ in a graph G to be a $[j, k]$ -set if for every vertex $v \in V \setminus S$, $j \leq |N(v) \cap S| \leq k$, that is, every vertex in $V \setminus S$ is adjacent to at least j vertices, but not more than k vertices in S . In [4] a similar definition was introduced with the additional condition of independence, and the minimum cardinality of an independent $[j, k]$ -set is denoted by $i_{[j, k]}(G)$. Note that the existence of such sets for all $j \leq k$ is not guaranteed in every graph and a characterization of trees having an independent $[1, k]$ -set can be found in [4].

In this paper we focus on independent $[1, 2]$ -sets, that is, an independent dominating set S of a graph G such that every vertex $u \in V(G) \setminus S$ has at most two neighbors in S . It is pointed out in [4] that $[1, 1]$ -sets are related to single error-correcting codes and the generalization to independent $[1, k]$ -sets is focused on that relationship. Thus from this point of view, in cases where independent $[1, 1]$ -sets are not available, independent $[1, 2]$ -sets could not correct some words, but they come from one of only two possible code words.

Here we deal with some open problems about this type of domination posed in [4]. In Section 2 we give a necessary condition for a graph G to have an independent $[1, 2]$ -set, in terms of its spanning trees, that is, an answer to Problem 2 in [4]. This necessary condition becomes also sufficient in the class of cactus graphs, what gives a partial answer to Problem 1 in [4]. In Section 3 we follow the concept of excellent tree proposed in [6] and we adapt it to the environment of our study, providing a characterization of trees in which every vertex belongs to some independent $[1, 2]$ -set, which is not necessarily minimum. The characterization of trees having an independent $[1, 2]$ -set in [4] does not allow to obtain a polynomial algorithm solving this decision problem, so in Section 4 we present a linear algorithm to decide whether a tree has an independent $[1, 2]$ -set. This algorithm can be easily modified to obtain the cardinality of a smallest independent $[1, 2]$ -set of a tree, therefore we can compute $i_{[1, 2]}(T)$; this solves the part of Problem 8 in [4] regarding this parameter.

2. A NECESSARY CONDITION FOR HAVING AN INDEPENDENT $[1, 2]$ -SET

In this section we provide a necessary condition for a graph G to have an independent $[1, 2]$ -set, in terms of its spanning trees, which gives an answer to Problem 2

posed in [4]. Recall that a *spanning tree* of a connected graph G is an acyclic connected subgraph that includes all the vertices of G . In addition we show that this condition is also sufficient in the family of *cactus graphs*, that is, graphs whose every edge belongs to at most one cycle, or equivalently, every block (maximal connected induced subgraph with no cut vertices) is a cycle or the path P_2 . This result gives a partial answer to Problem 1 in [4].

To this end we will need the family \mathcal{F}_2 of trees having an independent $[1, 2]$ -set that was characterized in Theorem 11 of [4]. For the sake of completeness we sketch here the construction. As a first step, we define the family of p_2 -trees in the following way: let T be a non-trivial tree and let $V(T) = X \cup Y$ be the unique bipartition of the vertex set. T will be a p_2 -tree if every vertex in one of the partite sets, say X , has degree at most 2; such a partite set is called a p_2 -set. It is clear that if X is a p_2 -set of T , then Y is an independent $[1, 2]$ -set of T . Theorem 11 of [4] states that a non-trivial tree T admits an independent $[1, 2]$ -set if and only if T can be obtained from a family T_1, \dots, T_t of p_2 -trees adding $t - 1$ edges where each edge joins vertices in two different sets X_i and X_j .

Such family of trees $\mathcal{G}(T) = \{T_1, \dots, T_t\}$ is a *generating family* of T and trees in family \mathcal{F}_2 are those trees having a generating family. The proof of Theorem 11 of [4] also shows the correspondence between generating families and independent $[1, 2]$ -sets in a tree T . We recall this relationship in the following definition.

Definition. Let $T \in \mathcal{F}_2$. The independent $[1, 2]$ -set associated to the generating family $\mathcal{G}(T) = \{T_1, \dots, T_t\}$ is $S = \bigcup_{i=1}^t Y_i$, where $V(T_i) = X_i \cup Y_i$ is the bipartition into a p_2 -set X_i and an independent $[1, 2]$ -set Y_i .

Conversely, the generating family associated to an independent $[1, 2]$ -set S is the family of trees of the forest obtained by removing from T all edges with both vertices in $V(T) \setminus S$.

The condition of having a generating family for trees that admit independent $[1, 2]$ -sets can be extended, in some sense, to every graph by means of spanning trees. To show it, we need the following definition.

Definition. Let G be a graph having a spanning tree $T \in \mathcal{F}_2$, with a generating family $\mathcal{G}(T) = \{T_1, \dots, T_t\}$ and let $V(T_j) = X_j \cup Y_j$ be the bipartition into a p_2 -set and an independent $[1, 2]$ -set, respectively. Let $e = uv$ be an edge of G that is not an edge of T .

1. e is called *type A* for the generating family $\mathcal{G}(T)$ if $u, v \in \bigcup_{j=1}^t X_j$.
2. e is called *type B* for the generating family $\mathcal{G}(T)$ if there exists $j_e \in \{1, \dots, t\}$ such that u is a leaf of T_{j_e} , $u \in X_{j_e}$ and $v \in Y_{j_e}$.

Theorem 1. Let G be a graph having an independent $[1, 2]$ -set. Then G has a spanning tree $T \in \mathcal{F}_2$ with a generating family $\mathcal{G}(T) = \{T_1, \dots, T_t\}$, such that any edge $e \in E(G) \setminus E(T)$ is type A or type B for $\mathcal{G}(T)$.

Proof. Let G be a graph having an independent $[1, 2]$ -set S . If G is a tree, then the conditions are trivially true. Now assume that G has an induced cycle C_0 . If there exists an edge e in C_0 with both vertices in $V(G) \setminus S$, then pick $e_1 = e$ (we call this case A), if each edge of C_0 has exactly one vertex in S , take e_1 any edge of C_0 (we call this case B). We define $G_1 = G - e_1$; if it is not a tree it has an induced cycle C_1 . Again either there is an edge $e_2 = u_2v_2$ in C_1 such that $u_2, v_2 \notin S$ (case A) or every edge of C_1 has exactly one vertex in S (case B). For the second case although C_0 and C_1 could share same edges, we can take e_2 an edge of C_1 which is not an edge of C_0 , because C_1 is an induced cycle in G_1 however vertices of C_0 do not induce a cycle in G_1 .

We repeat this process until we obtain $G_k = G - \{e_1, \dots, e_k\}$ a spanning tree of G , where each edge $e_i = u_i v_i$ belongs to an induced cycle C_{i-1} of $G_{i-1} = G - \{e_1, \dots, e_{i-1}\}$ and satisfies either $u_i, v_i \notin S$ or e_i is not an edge of any cycle $C_r, r < i - 1$, and every edge in C_{i-1} has exactly one vertex in S .

Now note that S is also an independent $[1, 2]$ -set of G_k , because removing edges from G does not affect independence and both cases A and B ensure that S dominates G_k . So $G_k \in \mathcal{F}_2$ and we can take $\mathcal{G}(G_k) = \{T_1, \dots, T_t\}$ the generating family of G_k associated to S . If edge $e_i = u_i v_i$ is in case A, then $u_i, v_i \in V(G_k) \setminus S = \bigcup_{j=1}^t X_j$ (type A edge). If $e_i = u_i v_i$ is in case B, then every edge of C_{i-1} has exactly one vertex in S and note that no other edge of C_{i-1} will be removed in successive steps of the construction of G_k , so u_i, v_i are connected in G_k by the path $C_{i-1} - e_i$, where each edge has one vertex in S . This means that u_i, v_i are in the same connected component of the forest resulting of removing from G_k all edges with no vertex in S , or equivalently that there exists $j_i \in \{1, \dots, t\}$ with $u_i, v_i \in V(T_{j_i})$. Moreover $u_i \notin S$ and $v_i \in S$ gives $u_i \in X_{j_i}$ and $v_i \in Y_{j_i}$.

Finally both neighbors of u_i in the cycle C_{i-1} belong to S , so if u_i has any other neighbor z in G which is not in C_{i-1} , it is clear that $z \notin S$, so edge $u_i z$ joins two different trees of the generating family $\mathcal{G}(G_k) = \{T_1, \dots, T_t\}$. That means u_i is a leaf of T_{j_i} (type B edge). ■

The following example shows that the converse of Theorem 1 is not true in general. The graph in Figure 1(a) has no independent $[1, 2]$ -set because all black vertices should be in such set, so vertex v would be dominated three times. However, the set of black vertices is an independent $[1, 2]$ -set of the tree in Figure 1(b), which is the spanning tree of G resulting from removing type B edges e_1 and e_2 .

The key point of this counterexample is that the spanning tree is obtained from G by removing some edges, such that at least one of them belongs to two induced cycles, in this example the edge e_1 . This idea leads us to the family of cactus graphs, where every edge belong to at most one cycle. In this graph class the necessary condition to have an independent $[1, 2]$ -set shown in Theorem 1 is also sufficient.

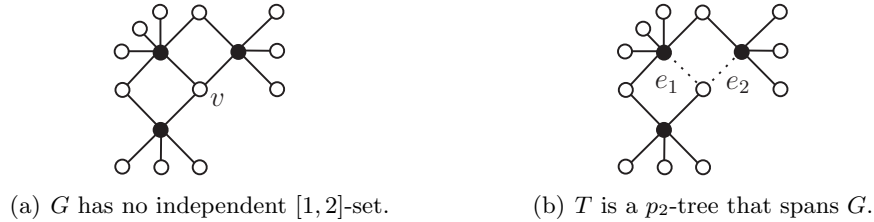


Figure 1. The converse of Theorem 1 is not true in general.

Theorem 2. *Let G be a cactus graph. Then G has an independent $[1, 2]$ -set if and only if it has a spanning tree $T \in \mathcal{F}_2$ with a generating family $\mathcal{G}(T) = \{T_1, \dots, T_t\}$, such that any edge $e \in E(G) \setminus E(T)$ is type A or type B for $\mathcal{G}(T)$.*

Proof. We just need to prove the sufficiency. Let $S = \bigcup_{j=1}^t Y_j$ be the independent $[1, 2]$ -set of T associated to the generating family $\mathcal{G}(T)$ and let us see that it is also an independent $[1, 2]$ -set of G . The graph G is obtained from the spanning tree T by adding some edges, so S is also a dominating set of G . Moreover, by hypothesis no added edge has both vertices in S , therefore S is independent in G .

Now we need to show that S is a $[1, 2]$ -set of G . Let $x \in V(G) \setminus S$. If every edge of G incident to x is an edge of T , then $N_G(x) = N_T(x)$ and x has at most two neighbors in S . On the contrary suppose that the set of edges incident to x which are in $E(G) \setminus E(T)$ is non-empty and denote those edges as e_1, \dots, e_r with $e_i = xy_i$. Using that G is a cactus graph and that removing these edges does not disconnect the graph, each edge e_i belongs to exactly one cycle C_i in G , with $C_i \neq C_j$ for $i \neq j$, and x is a common vertex of all of them (see Figure 2(a)).

Firstly assume that all edges e_1, \dots, e_r are of type A, that is, $y_i \in \bigcup_{j=1}^t X_j = V(G) \setminus S$, for all $i = 1, \dots, r$. Then the neighbors of x in G other than y_1, \dots, y_r , are also neighbors of x in T , so it is clear that x has at most two neighbors in G belonging to S . On the other hand suppose, without loss of generality, that $e_1 = xy_1$ is type B, so there exists $j \in \{1, \dots, t\}$ such that x is a leaf of T_j , $x \in X_j$ and $y_1 \in Y_j$. Therefore, x has just one neighbor in T_j , say z_1 , which is in Y_j , and both y_1, z_1 are neighbors of x in G belonging to S .

Let $w \in N_G(x) \setminus \{y_1, z_1\}$. If $w \in N_T(x)$, then w belongs to a tree $T_l \neq T_j$, the edge xw connects two different trees of the forest T_1, \dots, T_t , and by construction $w \notin S$. Finally, if $w \notin N_T(x)$, then $w \in \{y_2, \dots, y_k\}$, say $w = y_2$. Vertex y_2 belongs to cycle C_2 in G , different from cycle C_1 containing y_1 , and we denote the neighbor of x in C_2 , other than y_2 , by z_2 . Using the fact that x is a leaf of T_j with neighbor z_1 , which is a vertex of cycle C_1 , we obtain that $z_2 \neq z_1$, the edge xz_2 does not belong to T_j and thus $z_2 \notin V(T_j)$. So z_2 belongs to a tree of the forest T_1, \dots, T_t different from T_j and y_2 belongs to the same one. Therefore, y_2 does not belong to $V(T_j)$ (see Figure 2(b)). This means that edge xy_2 must be of type A and $w = y_2 \notin S$ as desired. ■

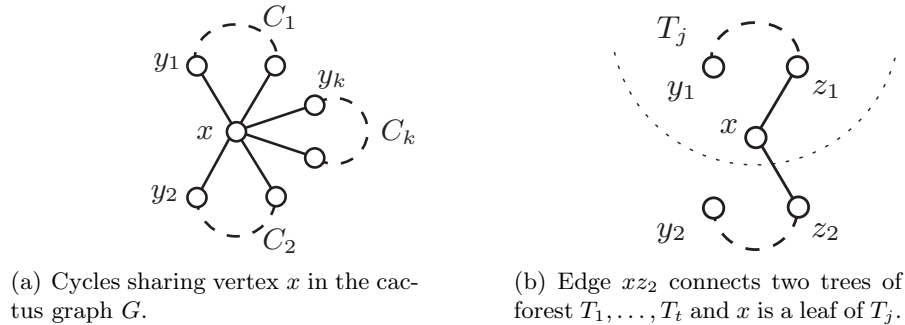


Figure 2. Some cases of Theorem 2.

3. EXCELLENT TREES

In this section we focus the study of graphs having an independent $[1, 2]$ -set from a different point of view, characterizing those graphs such that every vertex belongs one of these sets. This idea leads us to the concept of *excellent graph* introduced in [6]. For a graph $G = (V, E)$, let \mathcal{P} denote a property of subsets $S \subseteq V$. We call a set S with property \mathcal{P} having $\{\text{minimum, maximum}\}$ cardinality $\mu(G)$ a $\mu(G)$ -set. A vertex is called μ -good if it is contained in some $\mu(G)$ -set. A graph G is called μ -excellent if every vertex in V is μ -good. For instance G is γ -excellent if every vertex of G belongs to a minimum dominating set. This concept has been studied in the family of trees for different domination-type properties such as domination, irredundance and independence [6, 9], restrained domination [7] and total domination [10].

We define a similar concept for the independent $[1, 2]$ -domination. Having in mind that the existence of such sets is a key problem, we relax the conditions in the following way.

Definition. A graph G is $[1, 2]$ -semiexcellent if every vertex belongs to some independent $[1, 2]$ -set, not necessarily minimum.

Our target is to characterize the family of trees that are $[1, 2]$ -semiexcellent and to this end we will again use the concept of p_2 -tree and the family \mathcal{F}_2 described in Section 2. Firstly we show a necessary condition for a vertex in order to belong to some independent $[1, 2]$ -set.

Lemma 3. Let T be a tree and let $v \in V(T)$. If there exists an independent $[1, 2]$ -set S_v containing v , then for each $u \in N(v)$, the set $N(u) \setminus \{v\}$ contains at most one leaf.

Proof. If S_v is an independent $[1, 2]$ -set containing v and $u \in N(v)$, it is clear that $u \notin S_v$ and any leaf in $N(u) \setminus \{v\}$ must belong to S_v in order to be dominated,

so $N(u) \setminus \{v\}$ can have at most one leaf because u has at most two neighbors in S_v . ■

That condition turns out to be sufficient in the family of p_2 -trees, as the following lemma shows.

Lemma 4. *Let T be a p_2 -tree with $V(T) = X \cup Y$ the bipartition into a p_2 -set X and an independent $[1, 2]$ -set Y . Let $x \in X$ be such that any $y \in N(x)$ satisfies that $N(y) \setminus \{x\}$ contains at most one leaf. Then $S_x = (Y \setminus N(x)) \cup L(x) \cup \{x\}$ is an independent $[1, 2]$ -set of T containing x , where $L(x)$ is the set of leaves at distance two of x .*

Proof. Firstly the set $S_x = (Y \setminus N(x)) \cup L(x) \cup \{x\}$ is independent because Y is independent and all neighbors of vertices in $L(x) \cup \{x\}$ belong to $N(x)$. Let us see that S_x is a $[1, 2]$ -set. Let $y \in N(x)$; it is clear that y is dominated by x and using the hypothesis that $N(y) \setminus \{x\}$ has at most one leaf, there is at most one vertex in $L(x)$ that dominates y .

On the other hand, if $z \in V(T) \setminus S_x$ and $z \in N(y)$ for some $y \in N(x)$, then it is not a leaf so it has degree 2, because X is p_2 -set. Therefore, z has a unique neighbor $y' \neq y$ and it satisfies $y' \in S_x$ (see Figure 3). Finally, let $t \in V(T) \setminus S_x$ be such that $t \notin N[y]$ for any $y \in N(x)$. Then $t \in X$ has no neighbors in $L(x) \cup \{x\}$ and it has at least one and at most two neighbors in $Y \setminus N(x)$. ■

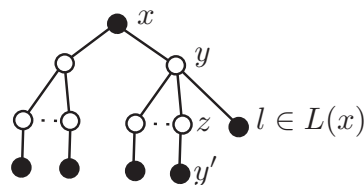


Figure 3. Black vertices are in S_x and white vertices are not in S_x .

A *strong support* vertex is a vertex having at least two leaves in its neighborhood. In any p_2 -tree it is possible to obtain an independent $[1, 2]$ -set that skips a fixed pair of non-leaves adjacent vertices, under the condition of having no strong support vertices.

Lemma 5. *Let T be a p_2 -tree with no strong support vertices, $V(T) = X \cup Y$ the bipartition into a p_2 -set X and an independent $[1, 2]$ -set Y , and let $x, y \in V(T)$ be two adjacent vertices such that none of them is a leaf, $x \in X$ and $y \in Y$. Then there exists an independent $[1, 2]$ -set $S(x, y)$ such that $x, y \notin S(x, y)$ and x has just one neighbor in $S(x, y)$.*

Proof. Using the fact that y is not a leaf, the set $N(y) \setminus \{x\}$ is non-empty. Firstly assume that $N(y)$ contains a (unique) leaf x_1 . Then $S(x, y) = (Y \setminus \{y\}) \cup \{x_1\}$ is an independent $[1, 2]$ -set of T with $x, y \notin S(x, y)$ and such that x has just one neighbor in it (see Figure 4(a)).

Let now $N(y)$ contains no leaves and take any vertex $x_1 \in N(y) \setminus \{x\}$. Then x_1 has degree 2 and let y_1 be a neighbor of x_1 other than y . If y_1 is a leaf or if $N(y_1)$ contains no leaves, then define $S(x, y) = (Y \setminus \{y, y_1\}) \cup \{x_1\}$ (see Figure 4(b)). If $N(y_1)$ contains a (unique) leaf, say x_2 , then define $S(x, y) = (Y \setminus \{y, y_1\}) \cup \{x_1, x_2\}$ (see Figure 4(c)). In any case $S(x, y)$ is an independent $[1, 2]$ -set of T with $x, y \notin S(x, y)$ and such that x has just one neighbor in it. ■

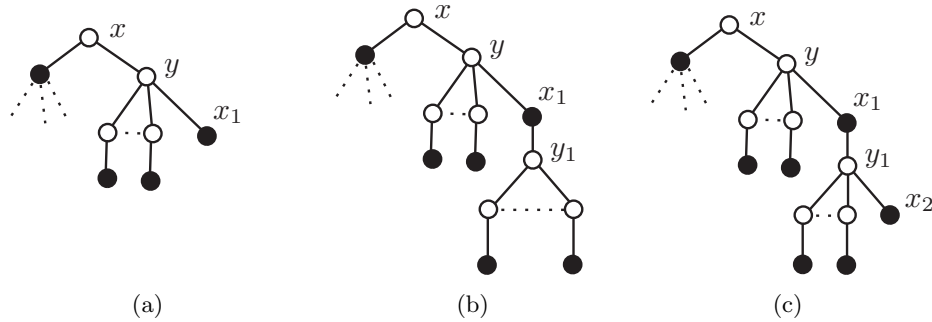


Figure 4. Black vertices are in $S(x, y)$.

The last lemma of this section shows that having no strong support vertices is a sufficient condition for a tree for belonging to the family \mathcal{F}_2 .

Lemma 6. *Let T be a tree with no strong support vertices. Then $T \in \mathcal{F}_2$.*

Proof. We root the tree T in a leaf v and we label the vertices of T as X or Y with the following rules. First of all we label v as X and its unique neighbor as Y . All the children of any vertex labeled as Y are labeled as X . If a vertex with label X has just one child we label it as Y . If a vertex with label X has two or more children and (just) one of them is a leaf, we label this leaf as Y and the rest of children as X and finally, if a vertex with label X has two or more children and none of them is a leaf, we label one of the children as Y and the rest of children as X .

Removing all edges of T between two vertices labeled as X gives a forest T_1, T_2, \dots, T_t and note that each T_i is a p_2 -tree where vertices labeled as X are a p_2 -set and vertices labeled as Y are an independent $[1, 2]$ -set. So we obtain a generating family for T and $T \in \mathcal{F}_2$ as desired. ■

Finally we show the characterization of $[1, 2]$ -semiexcellent trees, as trees having no strong support vertices with the exception of the path P_3 .

Theorem 7. *Let T be a tree, $T \neq P_3$. Then T is $[1, 2]$ -semiexcellent if and only if T has no strong support vertices.*

Proof. Suppose that T is $[1, 2]$ -semiexcellent and that $v \in V(T)$ is a strong support vertex of T . Let $u_1, u_2 \in N(v)$ be two leaves of T . Using the fact that $T \neq P_3$, there exists $w \in N(v) \setminus \{u_1, u_2\}$. By hypothesis there exists an independent $[1, 2]$ -set S_w containing w and by Lemma 3 the set $N(v) \setminus \{w\}$ contains at most one leaf, that contradicts the fact $u_1, u_2 \in N(v) \setminus \{w\}$.

Conversely, suppose that T has no strong support vertices and let $v \in V(T)$. By Lemma 6, $T \in \mathcal{F}_2$ so let $\mathcal{G}(T) = \{T_1, \dots, T_t\}$ be a generating family for T . If $v \in Y_i$ for some $i \in \{1, \dots, t\}$, then $v \in Y = \bigcup_{j=1}^t Y_j$ that is an independent $[1, 2]$ -set of T . So assume that $v = x \in X = \bigcup_{j=1}^t X_j$ and without loss of generality consider the case $x \in X_1$. We are going to construct an independent $[1, 2]$ -set of T containing x .

By Lemma 4, the set $S_x = (Y_1 \setminus N_{T_1}(x)) \cup L_{T_1}(x) \cup \{x\}$ is an independent $[1, 2]$ -set if T_1 , so it is clear that $S^1 = S_x \cup (\bigcup_{j=2}^t Y_j)$ is independent and dominates T . If S^1 is a $[1, 2]$ -set we are done. On the contrary, if there exists $u \in V(T) \setminus S^1$ with more than two neighbors in S^1 it must be (w.l.o.g.) $x_2 \in X_2$ with exactly one neighbor in S_x (by definition of the generating family $\mathcal{G}(T)$) and two neighbors in Y_2 , at least one of them, say y_2 , is not a leaf of T because T has no strong support vertices. Using Lemma 5, let $S_2 = S(x_2, y_2)$ be an independent $[1, 2]$ -set of T_2 such that $x_2, y_2 \notin S(x_2, y_2)$ and x_2 is dominated just once. Now we define $S^2 = S_x \cup S_2 \cup (\bigcup_{j=3}^t Y_j)$. Again S^2 is an independent dominating set of T , and if it is also a $[1, 2]$ -set, then we are done. If it is not the case, there exists $x_3 \in X_3$ (w.l.o.g.) with exactly one neighbor in $S_x \cup S_2$ (again by definition of the generating family $\mathcal{G}(T)$) and two neighbors in Y_3 . We repeat the same construction in T_3 as in T_2 (see Figure 5). Iterating the process as many times as necessary we finally obtain $S^r = S_x \cup S_2 \cup \dots \cup S_r \cup (\bigcup_{j=r+1}^t Y_j)$ which is an independent $[1, 2]$ -set of T containing x . ■

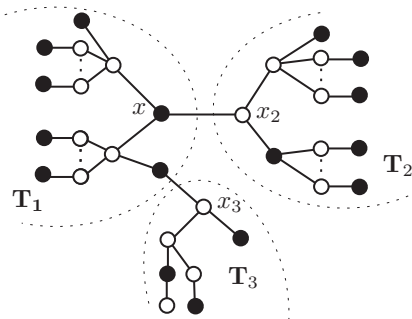


Figure 5. Black vertices are in the independent $[1, 2]$ -set containing x .

4. A LINEAR ALGORITHM FOR TREES

The characterization of trees having an independent $[1, 2]$ -set shown in Theorem 11 of [4] does not allow one to devise a polynomial algorithm to solve this decision problem. In this final section we provide an algorithm for this graph class. In addition our algorithm can be easily modified to obtain the cardinality of a smallest independent $[1, 2]$ -set of a tree, which provides an answer to Problem 8 in [4] regarding the parameter $i_{[1,2]}(T)$.

There are several different techniques for constructing a linear algorithm that computes a domination-like parameter in trees. In [5] Cockayne *et al.* gave the first linear-time algorithm for solving the domination problem in trees by a labeling method, and this method was later used in [14] to solve the k -domination problem in trees. Also in [13, 15] a labeling algorithm for the edge domination problem in trees is given and in [11] the same kind of algorithm for the total domination problem in trees can be found.

On the other hand, a number of algorithms using dynamic programming techniques also show that this method is useful in domination-like problems. As examples, in [1] the independent domination problem in trees is solved by this technique and in [12] a solution for weighted domination in trees is provided.

In all cases above the existence of a particular dominating set is guaranteed in every graph. This is not the case with independent $[1, 2]$ -sets and we adapt the ideas of the first technique to construct our algorithm that decides if an arbitrary tree has an independent $[1, 2]$ -set. We begin with the following definition.

Definition. Let G be a graph with at least two vertices, and let $v \in V(G)$. An independent set $S \subseteq V(G)$ is of *type I for v* if every vertex $u \in V(G) \setminus (S \cup \{v\})$ has one or two neighbors in S and, in addition, one of the following conditions holds:

1. $v \in S$, or
2. $v \notin S$ and it has at most two neighbors in S .

Let $I(v, G)$ denote the family of type I sets for v in G . Given $S \in I(v, G)$ we define the following labeling of v .

$$L_S(v) = \begin{cases} 0 & \text{if } v \in S, \\ k & \text{if } v \notin S \text{ and } v \text{ has } k \text{ neighbors in } S, 1 \leq k \leq 2, \\ -1 & \text{if } N[v] \cap S = \emptyset \text{ and every neighbor of } v \text{ has exactly one} \\ & \text{neighbor in } S, \\ -2 & \text{if } N[v] \cap S = \emptyset \text{ and there exists a neighbor of } v \text{ having} \\ & \text{two neighbors in } S. \end{cases}$$

Remark 8. Note that any independent $[1, 2]$ -set of G is of type I for every vertex in G . It is also clear from the definition that if there exists $R \in I(v, G)$ with $L_R(v) = -1$, then $S = R \cup \{v\}$ satisfies $S \in I(v, G)$ and $L_S(v) = 0$.

The following lemma is straightforward.

Lemma 9. *Let $K_{1,r}$ be the star with x_1, \dots, x_r ($r \geq 1$) leaves and center v . If $r \geq 3$, then the unique type I set for v is $S = \{v\}$ and $L_S(v) = 0$. If $r = 2$, then $S = \{v\}$ and $S' = \{x_1, x_2\}$ are the unique type I sets for v and they satisfy $L_S(v) = 0$, $L_{S'}(v) = 2$. If $r = 1$, then $S = \{v\}$ and $S' = \{x_1\}$ are the unique type I sets for v and they satisfy $L_S(v) = 0$, $L_{S'}(v) = 1$.*

In the following lemma we add one new vertex and just one edge to a graph and we show how to obtain all type I sets for the new vertex.

Lemma 10. *Let G be a graph with at least two vertices and let $v \in V(G)$. Let G' be the graph obtained from G and a new vertex v' by adding edge vv' and let $S' \subseteq V(G')$. Then*

1. $S' \in I(v', G')$ and $L_{S'}(v') = -2$ if and only if $S' \in I(v, G)$ and $L_{S'}(v) = 2$.
2. $S' \in I(v', G')$ and $L_{S'}(v') = -1$ if and only if $S' \in I(v, G)$ and $L_{S'}(v) = 1$.
3. $S' \in I(v', G')$ and $L_{S'}(v') = 0$ if and only if $S' = S \cup \{v'\}$, $S \in I(v, G)$ and $L_S(v) \in \{-2, -1, 1\}$.
4. $S' \in I(v', G')$ and $L_{S'}(v') = 1$ if and only if $S' \in I(v, G)$ and $L_{S'}(v) = 0$.
As a particular case, if there exists $R \in I(v, G)$ with $L_R(v) = -1$, then $S = R \cup \{v\}$ satisfies $S \in I(v', G')$ with $L_S(v') = 1$.

Proof. First and second statements are clear by the definition of type I set. Now assume that $S' \in I(v', G')$ and $L_{S'}(v') = 0$, then $v' \in S'$ and we define $S = S' \setminus \{v'\}$ which is a type I set for v in G . If v has two neighbors in S' , then $L_S(v) = 1$. If v' is the unique neighbor of v in S' and any other neighbor of v in G is dominated just once by vertices in S' , then $L_S(v) = -1$. If v' is the unique neighbor of v in S' and there is a neighbor of v in G which is dominated twice by vertices in S' , then $L_S(v) = -2$. The converse is trivial by definition of type I set.

To prove the last statement, just by definition, $S' \in I(v', G')$ and $L_{S'}(v') = 1$ if and only if $S' \in I(v, G)$ and $L_{S'}(v) = 0$. The additional implication comes from Remark 8. ■

Remark 11. In addition to characterize sets $S' \in I(v', G')$, Lemma 10 also ensures that from any $S \in I(v, G)$ at least one $S' \in I(v', G')$ can be obtained and it shows the labeling $L_{S'}(v')$ in each case.

In the next lemma we join two graphs with one new edge and we show how to obtain all type I sets for one vertex of this edge.

Lemma 12. *Let G, G' be two graphs with at least two vertices and let $v \in V(G)$, $v' \in V(G')$. Let G'' be the graph obtained from G and G' by adding edge vv' . Then*

1. $S'' \in I(v', G'')$ and $L_{S''}(v') = -2$ if and only if $S'' = S \cup S'$, $S \in I(v, G)$, $S' \in I(v', G')$ and $(L_S(v), L_{S'}(v')) \in \{(1, -2), (2, -2), (2, -1)\}$.
2. $S'' \in I(v', G'')$ and $L_{S''}(v') = -1$ if and only if $S'' = S \cup S'$, $S \in I(v, G)$, $S' \in I(v', G')$ and $(L_S(v), L_{S'}(v')) = (1, -1)$.
3. $S'' \in I(v', G'')$ and $L_{S''}(v') = 0$ if and only if $S'' = S \cup S'$, $S \in I(v, G)$, $S' \in I(v', G')$ and $(L_S(v), L_{S'}(v')) \in \{(-2, 0), (-1, 0), (1, 0)\}$.
As a particular case, if there exists $R' \in I(v', G')$ with $L_{R'}(v') = -1$, then $S'' = S \cup (R' \cup \{v'\})$, where $S \in I(v, G)$ and $L_S(v) \in \{-2, -1, 1\}$, satisfies $S'' \in I(v', G'')$ and $L_{S''}(v') = 0$.
4. $S'' \in I(v', G'')$ and $L_{S''}(v') = 1$ if and only if $S'' = S \cup S'$, $S \in I(v, G)$, $S' \in I(v', G')$ and $(L_S(v), L_{S'}(v')) \in \{(0, -2), (0, -1), (1, 1), (2, 1)\}$. As a particular case, if there exists $R \in I(v, G)$ with $L_R(v) = -1$, then $S'' = (R \cup \{v\}) \cup S'$, where $S' \in I(v', G')$ and $L_{S'}(v') \in \{-2, -1\}$, satisfies $S'' \in I(v', G'')$ and $L_{S''}(v') = 1$.
5. $S'' \in I(v', G'')$ and $L_{S''}(v') = 2$ if and only if $S'' = S \cup S'$, $S \in I(v, G)$, $S' \in I(v', G')$ and $(L_S(v), L_{S'}(v')) \in \{(0, 1), (1, 2), (2, 2)\}$. Furthermore suppose that there exists $R \in I(v, G)$ with $L_R(v) = -1$, then the set $S'' = (R \cup \{v\}) \cup S'$, where $S' \in I(v', G')$ and $L_{S'}(v') = 1$, satisfies $S'' \in I(v', G'')$ and $L_{S''}(v') = 2$.

Proof. The sufficient implication of each equivalence is trivial using the definition of type I set so we just prove the necessity. To this end let $S'' \in I(v', G'')$ and denote by $S = S'' \cap V(G)$ and $S' = S'' \cap V(G')$. Using that each graph has at least two vertices, S and S' are non-empty sets and it is clear that $S \in I(v, G)$ and $S' \in I(v', G')$.

1. If $L_{S''}(v') = -2$, then v' has no neighbors in S'' so $v' \notin S''$ and $L_S(v) \neq 0$. If v has just one neighbor z in S'' , then $z \in V(G)$ and $L_S(v) = 1$ and using the fact that $L_{S''}(v') = -2$, there exists a neighbor of v' in G' with two neighbors in S'' so $L_{S'}(v') = -2$. If v has two neighbors in S'' , then both of them belong to $V(G)$ and $L_S(v) = 2$. Moreover, the neighbors of v' in G' could have one or two neighbors in S'' , so $L_{S'}(v') \in \{-2, -1\}$.

2. If $L_{S''}(v') = -1$, then by hypothesis every neighbor of v' in G'' has just one neighbor in S'' and this easily implies that $L_S(v) = 1$ and $L_{S'}(v') = -1$.

3. If $L_{S''}(v') = 0$, then $v' \in S'$ so $L_{S'}(v') = 0$. If v has two neighbors in S'' , then $L_S(v) = 1$ and if v' is the unique neighbor of v in S'' , then $L_S(v) \in \{-2, -1\}$. The additional implication comes from Remark 8.

4. If $L_{S''}(v') = 1$, then there are two cases. If $v \in S''$, then $L_S(v) = 0$ and $L_{S'}(v') \in \{-2, -1\}$ and if $v \notin S''$, then $L_S(v) \in \{1, 2\}$ and $L_{S'}(v') = 1$. The additional implication comes from Remark 8.

5. If $L_{S''}(v') = 2$, then there are two cases. If $v \in S''$, then $L_S(v) = 0$ and $L_{S'}(v') = 1$ and if $v \notin S''$, then $L_S(v) \in \{1, 2\}$ and $L_{S'}(v') = 2$. The additional implication comes from Remark 8. ■

Remark 13. In addition to characterizing sets $S'' \in I(v', G'')$, Lemma 12 also ensures that from sets $S \in I(v, G)$ and $S' \in I(v', G')$ at least one $S'' \in I(v', G'')$ can be obtained if and only if $(L_S(v), L_{S'}(v')) \in \{(1, -2), (2, -2), (2, -1), (1, -1), (-2, 0), (-1, 0), (1, 0), (-2, -1), (-1, -1), (0, -2), (0, -1), (1, 1), (2, 1), (-1, -2), (0, 1), (1, 2), (2, 2), (-1, 1)\}$ and it shows $L_{S''}(v')$ in each case. As a consequence, it is not possible to obtain $S'' \in I(v', G'')$ containing S and S' in the remaining cases $(L_S(v), L_{S'}(v')) \in \{(-2, -2), (-2, 1), (-2, 2), (-1, 2), (0, 0), (0, 2), (2, 0)\}$.

Finally, we present a linear algorithm that decides whether or not a tree T has an independent $[1, 2]$ -set. The algorithm defines an order in the set of non-leaf vertices and proceeds bottom up in the tree. We denote by T_v the subtree rooted in any non-leaf vertex v . At any moment of the execution, the list $R(v)$ contains all the possible labels $L_S(v)$, where $S \in I(v, T_v)$.

There are three key steps in the algorithm that deserve special attention. When v is a support vertex, Lemma 9 is applied to the star with v as center and its descendant leaves to actualize $R(v)$, so it just contains the labels $L_S(v)$ for all possible type I sets of that star. For a non-leaf descendant u of v , Lemma 10 allows to actualize $R(v)$, so it just contains the labels $L_S(v)$ of all possible type I sets of the tree obtained from T_u and vertex $\{v\}$, by adding edge uv . Finally, Lemma 12 also allows to actualize $R(v)$, so it just contains the labels $L_S(v)$ for all possible type I sets of the tree obtained from T_u and T_v by adding edge uv .

Algorithm TREE-INDEPENDENT $[1, 2]$ -SET

Input: A tree T with n internal vertices.

Output: Whether or not T admits an independent $[1, 2]$ -set.

determine the leaves and the support vertices of the tree;

choose a non-leaf vertex as the root;

label all non-leaf vertices with different labels, and in such a

way that if u is a descendant of v , then $i(u) < i(v)$;

initialize a list for each vertex as $R(u) := \{\}$;

for $i := 1$ to n do

let v be the vertex with label i , i.e. $i(v) = i$;

if v is a support vertex, then apply Lemma 9 to the star with

v as center and its descendant leaves to actualize $R(v)$;

for each non-leaf descendant u of v

apply Lemma 12 and actualize $R(v)$;

if v is not a support vertex, then

pick one of its descendant u and apply Lemma 10 for

```

    actualizing  $R(v)$ ;
    for the rest of its descendant  $u$  of  $v$ 
        apply Lemma 12 and actualize  $R(v)$ ;
    if  $R(v) = \emptyset$ , then answer NO and end;
od;
if  $R(v) \cap \{0, 1, 2\} = \emptyset$  for the root  $v$ , then answer NO
    otherwise answer YES;
end.

```

Theorem 14. *Let T be a tree with n vertices. Algorithm TREE-INDEPENDENT $[1, 2]$ -SET decides in $O(n)$ time whether or not T has an independent $[1, 2]$ -set.*

Proof. At any moment of the execution, the list $R(v)$ associated with the non-leaf vertex v contains all the possible labels $L_S(v)$, where $S \in I(v, T_v)$ and T_v is the subtree rooted in v .

Note that if S is an independent $[1, 2]$ -set of T , then $S \cap V(T_v) \in I(v, T_v)$ for each non-leaf vertex v , so if $R(v) = \emptyset$ for some v , then T has no independent $[1, 2]$ -set. Moreover at the end, there is a independent $[1, 2]$ -set if and only if $R(v) \cap \{0, 1, 2\} \neq \emptyset$, where v is the root.

Regarding the complexity, both determining the leaves and the support vertices of the tree, and the initial labeling of non-leaf vertices can be done in linear time. In the rest of the algorithm, every vertex different from the root and the leaves is considered twice and the operations over it, are done in constant time. Hence, the final complexity is in $O(n)$. ■

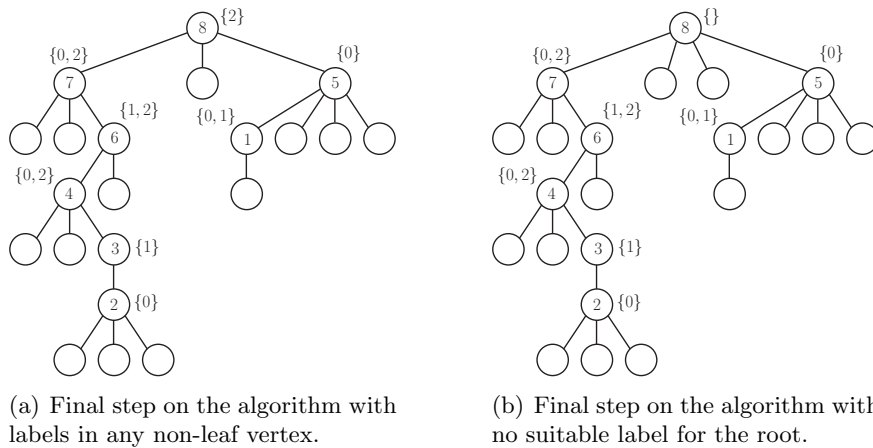


Figure 6. Two examples of the algorithm answering YES and NO.

Although we have preferred to introduce the algorithm in its present form for the sake of simplicity, it would not be difficult to modify it in order to keep

track of the minimum cardinality of the possible sets associated to a label in any vertex. Then we obtain a linear algorithm for computing the parameter $i_{[1,2]}(T)$, solving part of Problem 8 posed in [4].

We finally present two examples showing how the algorithm works. In Figure 6 there is a tree having an independent $[1, 2]$ -set and another one that has no such set. The vertex indices appear inside the circles. Figure 6(a) shows a tree and the final assignment of labels to every non-leaf vertex. The root v satisfies $R(v) = \{2\}$, so the tree has an independent $[1, 2]$ -set. On the other hand, in Figure 6(b) we show a different tree such that the root has no suitable label at the end of the algorithm, therefore the tree has no independent $[1, 2]$ -set.

Acknowledgement

We thank two anonymous referees for the comments and suggestions that resulted in an improved paper.

REFERENCES

- [1] T.A. Beyer, A. Proskurowski, S.T. Hedetniemi and S. Mitchell, *Independent domination in trees*, Congr. Numer. **19** (1997) 321–328.
- [2] G. Chartrand, L. Lesniak and P. Zhang, *Graphs and Digraphs*, 5th edition (CRC Press, 2011).
- [3] M. Chellali, S.T. Hedetniemi and A. McRae, *$[1, 2]$ -set in graphs*, Discrete Appl. Math. **161** (2013) 2885–2893.
doi:10.1016/j.dam.2013.06.012
- [4] M. Chellali, O. Favaron, T.W. Haynes, S.T. Hedetniemi and A. McRae, *Independent $[1, k]$ -sets in graphs*, Australas. J. Combin. **59** (2014) 144–156.
- [5] E. Cockayne, S. Goodman and S. Hedetniemi, *A linear algorithm for the domination number of a tree*, Inform. Process. Lett. **4** (1975) 41–44.
doi:10.1016/0020-0190(75)90011-3
- [6] G.F. Fricke, T.W. Haynes, S.M. Hedetniemi, S.T. Hedetniemi and R.C. Laskar, *Excellent trees*, Bull. Inst. Combin. Appl. **34** (2002) 27–38.
- [7] J.H. Hattingh and M.A. Henning, *Restrained domination excellent trees*, Ars Combin. **87** (2008) 337–351.
- [8] T.W. Haynes, S.T. Hedetniemi and P.J. Slater, *Fundamentals of Domination in Graphs* (Marcel Dekker Inc, 1998).
- [9] T.W. Haynes and M.A. Henning, *A characterization of i -excellent trees*, Discrete Math. **248** (2002) 69–77.
doi:10.1016/S0012-365X(01)00274-6
- [10] M.A. Henning, *Total domination excellent trees*, Discrete Math. **263** (2003) 93–104.
doi:10.1016/S0012-365X(02)00572-1

- [11] R. Laskar, J. Pfaff, S.M. Hedetniemi and S.T. Hedetniemi, *On the algorithmic complexity of total domination*, SIAM J. Algebraic Discrete Methods **5** (1984) 420–425. doi:10.1137/0605040
- [12] K.S. Natarajan and L.J. White, *Optimum domination in weighted trees*, Inform. Process. Lett. **7** (1978) 261–265. doi:10.1016/0020-0190(78)90012-1
- [13] S.L. Mitchell and S.T. Hedetniemi, *Edge domination in trees*, Congr. Numer. **19** (1977) 489–509.
- [14] P.J. Slater, *R-domination in graphs*, J. ACM **23** (1976) 446–450. doi:10.1145/321958.321964
- [15] M. Yannakakis and F. Gavril, *Edge dominating sets in graphs*, SIAM J. Appl. Math. **38** (1980) 364–372. doi:10.1137/0138030

Received 5 October 2016
Revised 23 December 2016
Accepted 23 January 2017