On the 2-maximal independence number of a graph

David F. Manlove

Department of Computing Science, University of Glasgow, Glasgow G12 8QQ, UK.
Email davidm@dcs.gla.ac.uk.

Abstract

An independent set S of a graph $G = (V, E)$ is k-maximal ($k \geq 1$) if, for all subsets A of S (where $|A| \leq k-1$), and all subsets B of $V \backslash S$ (where $|B| = |A|+1$), $(S \backslash A) \cup B$ is non-independent. Halldórsson [12] uses k-maximal independent sets as a means of approximating maximum independent sets in certain graph classes. Here we study the parameter $\beta_{0,2}^-(G)$, the smallest order of a 2-maximal independent set of a graph G, from an algorithmic point of view. We obtain a linear time and space algorithm for computing $\beta_{0,2}^-(T)$ for a tree T and also for constructing a minimum 2-maximal independent set of T. We also show that the decision problem related to computing $\beta_{0,2}^-$ is NP-complete for planar graphs of maximum degree 3.

1 Introduction

Independent sets in graphs have been extensively studied. The principal algorithmic focus has been the investigation of how we may efficiently determine maximum independent sets in a graph. The computational complexity of the problem of computing $\beta_0(G)$, the cardinality of a maximum independent set in a graph G, is surveyed comprehensively by Johnson [15], for a variety of classes of graphs.

Independent sets that are *maximal* have also been of interest. (An independent set I for a graph G is maximal if no proper superset of I is independent). For instance, the problem of counting the number of maximal independent sets in a graph has been shown to be #P-complete [22]. Also, a graph is *well-covered* if every maximal independent set is maximum – results in this area are surveyed by Plummer [21]. A much-studied parameter is $\beta_0^-(G)$, the cardinality of a minimum maximal independent set in G. The parameter β_0^- is also referred to as the minimum independent domination parameter. Independent domination was first studied by Cockayne and Hedetniemi [4]. The decision problem related to finding β_0^- is NP-complete for bipartite graphs [8] and planar cubic graphs [17], though polynomial-time solvable for chordal graphs [10] and trees [2], to name only a few graph classes.

For a graph G, the definition of $\beta_0^-(G)$ can be obtained by defining the strict partial order $\subset^G$ (strict set inclusion) on $\mathcal{I}(G)$, the set of all independent sets in G, and by considering the minimum over all $\subset^G$-maximal elements of $\mathcal{I}(G)$. However, it is of interest to consider other partial orders that may be defined on $\mathcal{I}(G)$, and the corresponding *minimaximal* optimization problems [16] that result from their definition.

In this paper we define a partial order $\prec_k^G$ on $\mathcal{I}(G)$, for $k \geq 1$, and consider the $\prec_k^G$-maximal (or simply k-maximal) members of $\mathcal{I}(G)$, for a given graph $G = (V, E)$. An independent set S is k-maximal if, for all subsets A of S (where $|A| \leq k-1$), and all subsets B of $V \backslash S$ (where $|B| = |A| + 1$), $(S \backslash A) \cup B$ is non-independent. The parameter

$\beta^-_{0,k}(G)$[1] will denote the minimum over all k-maximal independent sets for G. The concept of k-maximal independence in graphs was introduced by Bollobás et al. [3], and several non-algorithmic results concerning $\beta^-_{0,k}$, for $k \geq 1$, have been obtained [19, 5]. The related concept of *k-minimal domination* was also introduced by Bollobás et al. [3], and further details may be found in [20, 6, 7]. Halldórsson's approximation algorithms for finding maximum independent sets in various graph classes [12] involve constructing k-maximal independent sets. Nevertheless, k-maximal independent sets are interesting in their own right.

Investigating the computational complexity of $\beta^-_{0,k}$ was given as an open problem by Cockayne et al. [5]. However, the parameter $\beta^-_{0,2}$ has been studied by McRae [18], from an algorithmic point of view. She showed that the decision problem related to determining $\beta^-_{0,2}$ is NP-complete for bipartite graphs and line graphs of bipartite graphs. In this paper we give a linear algorithm for computing the cardinality of a minimum 2-maximal independent set, and for constructing such a set. The algorithm is based on that of Beyer et al. [2] for computing $\beta^-_0(T)$. We also demonstrate that the decision problem related to computing $\beta^-_{0,2}$ is NP-complete for planar graphs of maximum degree 3.

The remainder of the paper is organised as follows. In Section 2, we define formally $\subset^G_k$ and other notions related to k-maximal independence. In Section 3 we present the linear-time algorithm for computing $\beta^-_0(T)$, given a tree T. The NP-completeness result for planar graphs of maximum degree 3 is given in Section 4. Finally, in Section 5, we discuss some possible directions for further study, based on hierarchies of k-maximal independence parameters.

2 Definitions related to k-maximal independence

We begin by defining a family of partial orders, $\prec^G_k$, for a graph G and integer $k \geq 1$.

Definition 2.1. Let $G = (V, E)$ be a graph, and let $\mathcal{I}(G)$ denote the set of all independent sets of G. Define the following relation on $\mathcal{I}(G)$:

$$\subset^G_k = \left\{ (S', S'') \in \mathcal{I}(G) \times \mathcal{I}(G) : \begin{array}{l} \exists A \subseteq S' \ \wedge \ |A| \leq k-1 \\ \exists B \subseteq V \backslash S' \ \wedge \ |B| = |A| + 1 \\ S'' = (S' \backslash A) \cup B \end{array} \right\}.$$

By taking $\prec^G_k = (\subset^G_k)^*$ (the transitive closure of $\subset^G_k$), we obtain a partial order that we call $(k-1, k)$*-replacement*.

Intuitively, for $k \geq 1$ and two members S', S'' of $\mathcal{I}(G)$, $S' \subset^G_k S''$ if S'' can be obtained from S' by deleting a set A of $r-1$ elements from S' (where $r - 1 \leq k - 1$) and adding a set B of r elements from $V \backslash S'$. We note that, for $k \geq 2$, $\prec^G_k$ is a refinement of $\subset^G_{k-1}$. This is demonstrated by the following result, which follows by observing that $\subset^G_{i-1} \subseteq \subset^G_i$.

Proposition 2.2. *Let $\prec^G_k$ be the partial order of $(k-1, k)$-replacement. Then, for all $i \geq 2$, $\subset^G_{i-1}$ is contained in $\subset^G_i$.*

Corollary 2.3. *For all $k \geq 1$ and $1 \leq i \leq k$, $\subset^G_i$ is contained in $\prec^G_k$.*

[1] Mynhardt [19] and Cockayne et al. [5] refer to $\beta^-_{0,k}$ as β_k. However, for $k = 1$, this choice coincides with the maximum matching parameter (see Harary [13], for example), and for $k = 2$, this choice coincides with the total matching parameter of Alavi et al. [1]. In our notation, the subscript '0' of $\beta^-_{0,k}$ refers to *vertex independence* (as in Harary [13]), the subscript 'k' refers to k-maximality, and the superscript '-' refers to the minimum cardinality requirement.

We call an element S of $\mathcal{I}(G)$ $\subset_k^G$*-maximal* or k*-maximal* if there is no $S' \in \mathcal{I}(G)$ such that $S \subset_k^G S'$. Since independence is a *hereditary* property [14] (i.e., every subset of an independent set is independent), a 1-maximal member of $\mathcal{I}(G)$ is maximal. We may now formally define the parameters $\beta_{0,k}^-$ ($k \geq 1$) as follows:

$$\beta_{0,k}^-(G) = \min\{|S| : S \in \mathcal{I}(G) \wedge S \text{ is } k\text{-maximal}\}.$$

By the remarks in the preceding paragraph, $\beta_0^- = \beta_{0,1}^-$. By Proposition 2.2, $\beta_{0,k-1}^-(G) \leq \beta_{0,k}^-(G)$ for $k \geq 2$. Also, $\beta_{0,k}^-(G) \geq k$ for $k \geq 1$ [19]. In the remainder of this section, and in Sections 3 and 4, we restrict our attention to the case $k = 2$.

We have already noted that $\beta_0^-(G) \leq \beta_{0,2}^-(G)$. A simple example of where strict inequality can occur is provided by P_3, since $\beta_0^-(P_3) = 1$, whereas $\beta_{0,2}^-(P_3) = 2$.

We now present some elementary definitions relating to graphs, and use them to obtain a convenient criterion for an independent set to be 2-maximal. For a graph $G = (V, E)$ and vertex $v \in V$, define the *open neighbourhood* of v to be the set $N(v) = \{u \in V : \{u, v\} \in E\}$. Define the *closed neighbourhood* of v to be the set $N[v] = \{v\} \cup N(v)$. For a set of vertices $S \subseteq V$ and a vertex $v \in V$, the *private* S*-neighbours of* v are those vertices in the set $N[v] \backslash N[S \backslash \{v\}]$. It turns out that a maximal independent set S is 2-maximal if and only if S admits no augmenting P_3, as the following result, due to McRae [18], demonstrates. We include her proof for completeness.

Lemma 2.4 (McRae [18]). *Let $G = (V, E)$ be a graph. A maximal independent set of vertices $S \subseteq V$ is 2-maximal if and only if there do not exist vertices w, x and y of V such that $w \in S$, $x, y \notin S$, $\{x, y\} \notin E$ and x, y are private S-neighbours of w.*

Proof. Suppose $S \subseteq V$ is a maximal independent set. If there exist vertices w, x and y of V such that $w \in S$, $x, y \notin S$, $\{x, y\} \notin E$ and x, y are private S-neighbours of w, then it is clear that S is not 2-maximal. Conversely suppose that S is not 2-maximal. Then there exist vertices w, x and y of V such that $w \in S$, $x, y \notin S$ and $(S \backslash \{w\}) \cup \{x, y\}$ is independent. As S is maximal independent, $S \backslash \{w\}$ dominates $V \backslash N[w]$. Thus $x, y \in N(w)$, and x, y were private S-neighbours of w. Clearly also $\{x, y\} \notin E$. □

This result is utilised in our algorithm for trees, which constructs a minimum cardinality maximal independent set S of a tree T, with the added property that S admits no augmenting P_3.

3 Linear-time algorithm for trees

Before presenting the main result of this section, we make some further definitions relating to trees. Our algorithm is adapted from the one used by Beyer et al. [2] to calculate $\beta_0^-(T)$ for a tree T, and we hence use similar notation. For a rooted tree T and any vertex v of T, define T_v to be the subtree of T with root v. For a vertex v of T and a set of vertices $S \subseteq V(T)$, v is said to be *bad* with respect to S if $v \notin S$ and no child of v is in S. As a result of Lemma 2.4, the following necessary condition for an independent set S of a tree T to be 2-maximal can easily be verified.

Lemma 3.1. *Let $T = (V, E)$ be a tree and $S \subseteq V$ be independent. Then S is 2-maximal implies that every v in S has at most one bad child with respect to S.*

In view of this observation, we define the following five functions:

$INN(v)$: The smallest number of vertices $S \subseteq V(T_v)$ in a 2-maximal independent set for T_v that contains v, such that v has no bad children with respect to S.

$INBC(v)$: The smallest number of vertices $S \subseteq V(T_v)$ in a 2-maximal independent set for T_v that contains v, such that v has one bad child with respect to S.

$OUTO(v)$: The smallest number of vertices $S \subseteq V(T_v)$ in a 2-maximal independent set for T_v that does not contain v, such that v has exactly one child w in S, and w has one bad child with respect to S.

$OUTC(v)$: The smallest number of vertices $S \subseteq V(T_v)$ in a 2-maximal independent set for T_v that does not contain v, such that *either:*

1. v has more than one child in S *or*
2. v has exactly one child w in S, and w has no bad children with respect to S.

$OUTN(v)$: The smallest number of vertices $S \subseteq V(T_v)$ in a 2-maximal independent set for T_v that does not contain v nor any child of v.

Given the definitions of the five functions presented above, together with the fact that any 2-maximal independent set of a tree T rooted at u must include at least one of u and the children of u, it follows that

$$\beta^-_{0,2}(T) = \min\{INN(u), INBC(u), OUTO(u), OUTC(u)\}.$$

The algorithm for finding a minimum 2-maximal independent set of a tree T uses Lemma 3.1 and constructs a set of vertices S of T such that, for each vertex v with parent u in S, v has at most one bad child, and also v cannot be augmented by a P_3 whose vertices are u, v and a child of v. The dynamic programming approach is based on the following result, which demonstrates relationships between the above five functions for adjacent vertices in a tree, and which also proves the correctness of the algorithm.

Theorem 3.2. *Let T'_u, T''_v be two subtrees of T rooted at u and v respectively, for two vertices u and v of T. Let T be the tree with root u that is obtained by joining vertices u and v by an edge. Then*

1. $INN(u) = INN'(u) + \min\{OUTC''(v), OUTO''(v)\}$
2. $INBC(u) = \min \left\{ \begin{array}{l} INBC'(u) + OUTC''(v), \\ INBC'(u) + OUTO''(v), \\ INN'(u) + OUTN''(v) \end{array} \right\}$
3. $OUTO(u) = \min \left\{ \begin{array}{l} OUTO'(u) + OUTC''(v), \\ OUTN'(u) + INBC''(v) \end{array} \right\}$
4. $OUTC(u) = \min \left\{ \begin{array}{l} OUTC'(u) + OUTC''(v), \\ OUTN'(u) + INN''(v), \\ OUTC'(u) + INN''(v), \\ OUTC'(u) + INBC''(v), \\ OUTO'(u) + INN''(v), \\ OUTO'(u) + INBC''(v) \end{array} \right\}$
5. $OUTN(u) = OUTN'(u) + OUTC''(v)$

Proof. Let I be a minimum 2-maximal independent set for T and define $I' = I \cap V(T'_u)$ and $I'' = I \cap V(T''_v)$. We consider the following disjoint cases.

(1). Suppose that I contains u, and u has no bad children in T. Then u has no bad children in T'_u and v has at least one child in I. Hence $|I'| = INN'(u)$, and the existence of u implies that if v has only one child w in I, then w is allowed to have a bad child in T''_v. Thus $|I''| = \min\{OUTC''(v), OUTO''(v)\}$.

(2). Suppose that I contains u, and u has one bad child in T. Then u has one bad child, either (i) in T'_u, or (ii) v is bad.

In (i), $|I'| = INBC'(u)$, and v has at least one child in I. As in (1), the existence of u implies that if v has only one child w in I, then w is allowed to have a bad child in T''_v. Thus $|I''| = \min\{OUTC''(v), OUTO''(v)\}$.

In (ii), u has no bad child in T'_u, so $|I'| = INN'(u)$, and v has no children in I. Thus $|I''| = OUTN''(v)$.

(3). Suppose that I does not contain u, but is such that u has exactly one child w in I, and w has a bad child in T. Then either (i) $w \in I'$ or (ii) w is v.

In (i), $|I'| = OUTO'(u)$. Also, v must have a child in I, or else $I \cup \{v\}$ is independent, a contradiction. If v has only one child w which has a bad child, then I is not 2-maximal, since $u \notin I$. Hence $|I''| = OUTC''(v)$.

In (ii), $|I'| = OUTN'(u)$. Also, $|I''| = INBC''(v)$.

(4). Suppose that I does not contain u, but is such that either u has more than one child in I, or u has exactly one child w in I, and w has no bad children in T. Then either (i) u has a child in I', or (ii) $v \in I$, or (iii) both.

In (i), u cannot have a sole child w in I' such that w has a bad child, since $v \notin I$. Thus $|I'| = OUTC'(u)$. Also, v must have at least one child in I, or else $I \cup \{v\}$ is independent, a contradiction. If v has only one child w which has a bad child, then I is not 2-maximal, since $u \notin I$. Hence $|I''| = OUTC''(v)$.

In (ii), $|I'| = OUTN'(u)$. Also, v cannot have a bad child in T''_v, for then I would not be 2-maximal, as no child of u is in I'. Hence $|I''| = INN''(v)$.

In (iii), the existence of a child of u in I' means that v is permitted to have a bad child in T''_v. Hence $|I''| = \min\{INN''(v), INBC''(v)\}$. Also, the existence of $v \in I$ means that it is permissible for u to have a sole child w in I' such that w has a bad child. Hence $|I'| = \min\{OUTC'(u), OUTO'(u)\}$.

(5). Suppose that I does not contain u, nor any child of u. Then $|I'| = OUTN'(u)$. Also, v must have a child in I, or else $I \cup \{v\}$ is independent, a contradiction. If v has only one child w which has a bad child, then I is not 2-maximal, since $u \notin I$. Hence $|I''| = OUTC''(v)$. □

The above result forms the basis of the algorithm shown in Figure 1, which calculates $\beta^-_{0,2}(T)$ for a tree T.

Given a tree T with N vertices, the five functions $INN, INBC, OUTO, OUTC$ and $OUTN$ are initialised as follows. For a singleton subtree of T consisting only of some vertex v, it is clear that $INN(v) = 1$ and $OUTN(v) = 0$. The values of the other three functions $INBC(v), OUTO(v)$ and $OUTC(v)$ are however undefined. They are therefore given value N, which is large enough not to affect the remainder of the procedure of computing $\beta^-_{0,2}(T)$.

It may be verified that the algorithm in Figure 1 requires $O(N)$ time for execution. The initialization is clearly $O(N)$, and the main loop is also $O(N)$, since the values of the five functions may be computed in a constant number of steps, for each iteration. It is also clear that $O(N)$ space is required.

procedure *tree-min-2-max-ind-set* (*parent* : **array** [2..N] **of** [1..N]) **return** $\mathbb{N}$
-- Given a tree T with vertices $1, 2, \ldots, N$, calculate $\beta^-_{0,2}(T)$. T is rooted at vertex 1 and the
-- vertices are numbered breadth-first from the root. T is represented by a *parent* array, i.e.
-- $i = parent[j]$ if and only if i is the parent of j in T.

var $i, j : \mathbb{N}$;
  $INN, INBC, OUTC, OUTN, OUTO$: **array** [1..N] **of** $\mathbb{N}$;
begin
  for i **in** [1..N] **loop** -- Initialization
    $INN[i] := 1$;
    $INBC[i] := N$; -- Indicates value undefined in this case
    $OUTO[i] := N$; -- Indicates value undefined in this case
    $OUTC[i] := N$; -- Indicates value undefined in this case
    $OUTN[i] := 0$;
  end loop;
  for j **in reverse** [2..N] **loop** -- Propagate values towards root
    $i := parent[j]$;
    $INN[i] := INN[i] + \min\{OUTC[j], OUTO[j]\}$;

$$INBC[i] := \min \left\{ \begin{array}{l} INBC[i] + OUTC[j], \\ INBC[i] + OUTO[j], \\ INN[i] + OUTN[j] \end{array} \right\};$$

$$OUTO[i] := \min \left\{ \begin{array}{l} OUTO[i] + OUTC[j], \\ OUTN[i] + INBC[j] \end{array} \right\};$$

$$OUTC[i] := \min \left\{ \begin{array}{l} OUTC[i] + OUTC[j], \\ OUTN[i] + INN[j], \\ OUTC[i] + INN[j], \\ OUTC[i] + INBC[j], \\ OUTO[i] + INN[j], \\ OUTO[i] + INBC[j] \end{array} \right\};$$

    $OUTN[i] := OUTN[i] + OUTC[j]$;
  end loop;
  return $\min\{INN[1], INBC[1], OUTO[1], OUTC[1]\}$;
end *tree-min-2-max-ind-set*

Figure 1: Algorithm to find the minimum 2-maximal independence number of a tree.

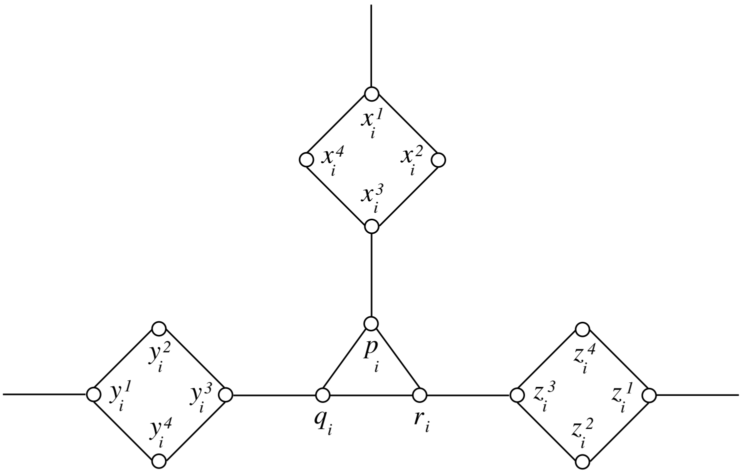

Figure 2: A typical vertex component from the constructed instance of MINIMUM 2-MAXIMAL INDEPENDENT SET.

4 NP-completeness in planar graphs of maximum degree 3

In this section, we prove that the MINIMUM 2-MAXIMAL INDEPENDENT SET problem is NP-complete, even for planar graphs of maximum degree 3. This decision problem takes a graph $G = (V, E)$ and integer $K \in \mathbb{Z}^+$ as input, and asks whether $\beta^-_{0,2}(G) \leq K$.

Theorem 4.1. MINIMUM 2-MAXIMAL INDEPENDENT SET *is NP-complete for planar graphs of maximum degree 3.*

Proof. Clearly MINIMUM 2-MAXIMAL INDEPENDENT SET belongs to NP. To show NP-hardness, we give a transformation from the INDEPENDENT SET problem for cubic planar graphs [11, problem GT20]. Hence let $G = (V, E)$ (a cubic planar graph) and K (a positive integer) be an instance of INDEPENDENT SET. Assume that $V = \{v_1, v_2, \ldots, v_n\}$. We construct an instance $G' = (V', E')$ (planar graph of maximum degree 3) and K' (positive integer) of MINIMUM 2-MAXIMAL INDEPENDENT SET.

Corresponding to every vertex $v_i \in V$ $(1 \leq i \leq n)$, construct a *vertex component* C_i of G', as illustrated in Figure 2, containing vertices $p_i, q_i, r_i, x_i^j, y_i^j, z_i^j$, for $1 \leq j \leq 4$, edges $\{x_i^4, x_i^1\}$, $\{x_i^j, x_i^{j+1}\}$, $\{y_i^4, y_i^1\}$, $\{y_i^j, y_i^{j+1}\}$, $\{z_i^4, z_i^1\}$, $\{z_i^j, z_i^{j+1}\}$, for $1 \leq j \leq 3$, and edges $\{x_i^3, u_i\}, \{y_i^3, v_i\}, \{z_i^3, w_i\}$, $\{p_i, q_i\}, \{p_i, r_i\}, \{q_i, r_i\}$. We denote by V_i the vertices in C_i.

For each $s_i \in \{x_i^1, y_i^1, z_i^1\}$, join s_i to a unique vertex $s_j \in \{x_j^1, y_j^1, z_j^1\}$ in C_j such that $\{v_i, v_j\} \in E$. There is obviously a degree of freedom involved in making such attachments, however the actual choice of assignment does not affect the planarity of G', nor the remainder of the proof. It is clear that the graph G' constructed is planar of maximum degree 3. Set $K' = 7n - K$. We now show that G has an independent set of cardinality at least K if and only if G' has a 2-maximal independent set with cardinality at most K'.

For, suppose that I is an independent set for G, where $|I| = k \geq K$. We construct a set S as follows. For each i $(1 \leq i \leq n)$, if $v_i \in I$, add the vertices $x_i^1, x_i^3, y_i^1, y_i^3, z_i^1, z_i^3$ to S. If $v_i \notin I$, add the vertices $p_i, x_i^2, x_i^4, y_i^2, y_i^4, z_i^2, z_i^4$ to S. S is independent in G', for if $\{s_i^1, t_j^1\} \in E'$, where s is x, y or z, and t is x, y or z, then $\{v_i, v_j\} \in E$. As I is independent in G then without loss of generality $v_i \notin I$, so that none of x_i^1, y_i^1 or z_i^1 is in S. Also, S is 2-maximal in G', for S is certainly maximal. Also, S admits no augmenting P_3 in G'. For, if $v_i \in I$ $(1 \leq i \leq n)$ then any P_3 in G' that augments s_i^1 or s_i^3 (where s is x, y or z) must include at least one of the vertices s_i^2, s_i^4, neither of which is available. If $v_i \notin I$ $(1 \leq i \leq n)$ then similarly any P_3 in G' that augments s_i^2 or s_i^4 (where s is x, y or z) must include at least one of the vertices s_i^1, s_i^3, neither of which is available. Also, it is clear that no P_3 in G' can augment p_i. Finally, $|S| = 6k+7(n-k) \leq 7n-K = K'$ as required.

Conversely suppose that S is a 2-maximal independent set for G', where $|S| \leq K'$. For a given i $(1 \leq i \leq n)$, we consider the elements of $S \cap V_i$. By the maximality of S, we see that the vertices s_i^2, s_i^4 must be dominated by vertices of V_i (where s is x, y or z). Since S is 2-maximal, we have that $s_i^1 \in S$ if and only if $s_i^3 \in S$, where s is x, y or z. Also, the maximality of S implies that $s_i^2 \in S$ if and only if $s_i^4 \in S$, where s is x, y or z. Thus $|S \cap V_i| \geq 6$.

It may be verified that $|S \cap V_i| = 6$ if and only if $S \cap V_i = W_i$, where

$$W_i = \{x_i^1, x_i^3, y_i^1, y_i^3, z_i^1, z_i^3\}.$$

Moreover, if $W_i \not\subseteq S \cap V_i$, then by the comments in the preceding paragraph, it is straightforward to check that $|S \cap V_i| = 7$. Define

$$I = \{v_i \in V : S \cap V_i = W_i\}.$$

We firstly claim that I is independent in G. For, if $\{v_i, v_j\} \in E$, then $\{s_i^1, t_j^1\} \in E'$, where s is x, y or z and t is x, y or z. If $v_i \in I$ then $x_i^1, y_i^1, z_i^1 \in S$. But S is independent in G', so that $t_j^1 \notin S$. Thus by construction of I, $v_j \notin I$ as required. Now let $k = |I|$ and suppose for a contradiction that $k < K$. Then $|S| = 6k + 7(n-k) > 7n - K = K'$ which is a contradiction. Hence $k \geq K$ as required. □

5 Conclusion and open problems

The complexity results for $\beta_{0,2}^-$ in trees and planar graphs presented here leave open the algorithmic complexity of $\beta_{0,2}^-$ in other classes of graphs, for example chordal graphs.

It is also interesting to consider the partial orders $\prec_k^G$ for $k > 2$, and the corresponding parameters $\beta_{0,k}^-$ for $k > 2$. We have already seen that, for $k = 1, 2$, the decision problem related to finding $\beta_{0,k}^-$ is NP-complete in bipartite and planar graphs, but polynomial-time solvable for trees, and we conjecture that this is the case for each fixed $k > 2$.

As a variation on the above hierarchy of parameters, consider the following. For a graph G and integer $k \leq \beta_0(G) - 1$, let $\beta_{0,\beta_0(G)-k}^-(G)$ denote the smallest order of a $(\beta_0(G) - k)$-maximal independent set of G. Now $\beta_0(G) - k \leq \beta_{0,\beta_0(G)-k}^-(G) \leq \beta_0(G)$ [19], so that $\beta_{0,\beta_0(G)}^-(G) = \beta_0(G)$, in the case $k = 0$. Thus finding $\beta_{0,\beta_0(G)}^-(G)$ is polynomial-time solvable for G a bipartite graph [13]. However, for general k, we conjecture that the associated decision problem is NP-complete for bipartite graphs.

We may also consider the parameters $\beta_{0,k}^-$ for line graphs $L(G)$ of general graphs G, for $k \geq 1$. The analagous parameter to $\beta_{0,k}^-$ in line graphs is $\beta_{1,k}^-$, the *minimum k-maximal matching* parameter. Thus $\beta_{1,k}^-(G) = \beta_{0,k}^-(L(G))$. For $k = 1, 2$, the decision problem related to finding $\beta_{1,k}^-$ in bipartite graphs is NP-complete [23, 18], and we conjecture that this is the case for each fixed $k > 2$. As above, we may consider, for a graph G and integer $k \leq \beta_1(G) - 1$, the parameter $\beta_{1,\beta_1(G)-k}^-(G)$. As before, $\beta_{1,\beta_1(G)}^-(G) = \beta_1(G)$, so finding $\beta_{1,\beta_1(G)}^-(G)$ is polynomial-time solvable for arbitrary graphs [9]. But for general k, we conjecture that the related decision problem is again NP-complete.

Acknowledgement

The author would like to thank Rob Irving for valuable comments on previous drafts of this paper.

References

[1] Y. Alavi, M. Behzad, L.M. Lesniak-Foster, and E.A. Nordhaus. Total matchings and total coverings of graphs. *Journal of Graph Theory*, 1:135–140, 1977.

[2] T. Beyer, A. Proskurowski, S.T. Hedetniemi, and S. Mitchell. Independent domination in trees. In *Proceedings of the 8th South-Eastern Conference on Combinatorics, Graph Theory and Computing*, pages 321–328. Utilitas Mathematica, 1977.

[3] B. Bollobás, E.J. Cockayne, and C.M. Mynhardt. On generalised minimal domination parameters for paths. *Discrete Mathematics*, 86:89–97, 1990.

[4] E.J. Cockayne and S.T. Hedetniemi. Independence graphs. In *Proceedings of the 5th South-Eastern Conference on Combinatorics, Graph Theory and Computing*, pages 471–491. Utilitas Mathematica, 1974.

[5] E.J. Cockayne, G. MacGillivray, and C.M. Mynhardt. Generalised maximal independence parameters for paths and cycles. *Quaestiones Mathematicae*, 13:123–139, 1990.

[6] E.J. Cockayne and C.M. Mynhardt. On the product of k-minimal domination numbers of a graph and its complement. *Journal of Combinatorial Mathematics and Combinatorial Computing*, 8:118–122, 1990.

[7] E.J. Cockayne and C.M. Mynhardt. Domination sequences of graphs. *Ars Combinatoria*, 33:257–275, 1992.

[8] D.G. Corneil and Y. Perl. Clustering and domination in perfect graphs. *Discrete Applied Mathematics*, 9(1):27–39, 1984.

[9] J. Edmonds. Paths, trees and flowers. *Canadian Journal of Mathematics*, 17:449–467, 1965.

[10] M. Farber. Independent domination in chordal graphs. *Operations Research Letters*, 1:134–138, 1982.

[11] M.R. Garey and D.S. Johnson. *Computers and Intractability.* Freeman, San Francisco, CA., 1979.

[12] M.M. Halldórsson. Approximating discrete collections via local improvements. In *Proceedings of SODA '95: the 6th Annual ACM-SIAM Symposium on Discrete Algorithms, San Francisco*, pages 160–169. ACM-SIAM, 1995.

[13] F. Harary. *Graph Theory.* Addison-Wesley, 1969.

[14] S.T. Hedetniemi. Hereditary properties of graphs. *Journal of Combinatorial Theory, Series B*, 14:94–99, 1973.

[15] D.S. Johnson. The NP-completeness column: an ongoing guide. *Journal of Algorithms*, 6:434–451, 1985.

[16] D.F. Manlove. *Minimaximal and maximinimal optimisation problems: a partial order-based approach.* PhD thesis, University of Glasgow, Department of Computing Science, 1998.

[17] D.F. Manlove. On the algorithmic complexity of twelve covering and independence parameters of graphs. *Discrete Applied Mathematics*, 91:155–175, 1999.

[18] A.A. McRae. *Generalizing NP-completeness proofs for bipartite and chordal graphs*. PhD thesis, Clemson University, Department of Computer Science, 1994.

[19] C.M. Mynhardt. Generalised maximal independence and clique numbers of graphs. *Quaestiones Mathematicae*, 11:383–398, 1988.

[20] C.M. Mynhardt and E.J. Cockayne. k-minimal domination numbers of cycles. *Ars Combinatoria*, 23A:195–206, 1987.

[21] M.D. Plummer. Well covered graphs: a survey. *Quaestiones Mathematicae*, 16:253–287, 1993.

[22] L.G. Valiant. The complexity of enumeration and reliability problems. *SIAM Journal on Computing*, 8(3):410–421, 1979.

[23] M. Yannakakis and F. Gavril. Edge dominating sets in graphs. *SIAM Journal on Applied Mathematics*, 18(1):364–372, 1980.