Nordic Journal of Computing 1(1994), 157–171.

COMPLEXITY OF DOMINATION-TYPE PROBLEMS IN GRAPHS*

JAN ARNE TELLE
Department of Computer and Information Science
University of Oregon, Eugene, Oregon 97403, USA
telle@cs.uoregon.edu

Abstract. Many graph parameters are the optimal value of an objective function over selected subsets S of vertices with some constraint on how many selected neighbors vertices in S, and vertices not in S, can have. Classic examples are minimum dominating set and maximum independent set. We give a characterization of these graph parameters that unifies their definitions, facilitates their common algorithmic treatment and allows for their uniform complexity classification. This characterization provides the basis for a taxonomy of domination-type and independence-type problems. We investigate the computational complexity of problems within this taxonomy, identify classes of NP-complete problems and classes of problems solvable in polynomial time.

CR Classification: F.2.2, G.2.2

1. Introduction

If every vertex in a selected subset S of vertices of a graph has zero selected neighbors then S is an independent set, and similarly if every vertex not in S has at least one selected neighbor then S is a dominating set. This suggests a common characterization of independent sets and dominating sets based on the constraints imposed on the number of selected neighbors the vertices in S, and vertices not in S, can have. As we show in this paper, a large collection of well-known vertex subset properties admit such a characterization. For these vertex subset properties we consider the graph parameters definable by optimization over, or existence of, vertex subsets having the property. From the standard definitions of these parameters it is not obvious that they are related as described here. This characterization thus facilitates the common algorithmic treatment of the problems computing these parameters. In recent years, a variety of domination-type parameters in graphs have been introduced, and the number of papers devoted to this topic is steadily increasing [11, 12].

* This work supported by The Norwegian Research Council for Science and Humanities and by NSF grant CCR9213439.

Received July 1993. Accepted December 1993.

In the next section, we present our characterization and show that many of the graph parameters found in the literature admit such a characterization. We give a table cataloging the computational complexity of computing these parameters. In section 3, we investigate the computational complexity of further problems admitting the characterization. We identify several classes of NP-complete and of polynomial-time solvable problems. The NP-completeness results identify properties with interesting features, such as when both maximum and minimum versions are NP-complete (independent dominating sets), or when merely deciding whether a graph has a vertex subset with the property is NP-complete (perfect codes). A natural by-product of these results is the introduction of several new domination-type parameters in graphs.

2. Characterization of Domination-type Problems

We use standard graph terminology [3]. For a vertex $v \in V(G)$ of a graph G, let $N_G(v) = \{u : (u,v) \in E(G)\}$ be the set of neighbors of v and $deg_G(v) = |N_G(v)|$. For $S \subseteq V(G)$ let $G[S]$ denote the graph induced in G by S and let the symbols σ and ρ indicate membership in S and membership in $V(G) \setminus S = \{v \in V(G) : v \notin S\}$, respectively.

Definition 1. *Given a graph G and a set $S \subseteq V(G)$ of* selected *vertices*

- *The* state *of a vertex $v \in V(G)$ is*

$$state_S(v) \overset{df}{=} \begin{cases} \rho_i & \text{if } v \notin S \text{ and } |N_G(v) \cap S| = i \\ \sigma_i & \text{if } v \in S \text{ and } |N_G(v) \cap S| = i \end{cases}$$

- *Define syntactic abbreviations*

$$\rho_{\leq i} \equiv \rho_0, \rho_1, ..., \rho_i \qquad \rho_{\geq i} \equiv \rho_i, \rho_{i+1}, ...$$
$$\sigma_{\leq i} \equiv \sigma_0, \sigma_1, ..., \sigma_i \qquad \sigma_{\geq i} \equiv \sigma_i, \sigma_{i+1}, ...$$

Thus $\rho_{\geq i}$ and $\sigma_{\geq i}$ each represents an infinite set of states. Mnemonically, σ represents a vertex selected for S and ρ a vertex rejected from S, with the subscript indicating the number of neighbors the vertex has in S. A variety of vertex subset properties can be defined by allowing only a specific set L as *legal* states of vertices. For example, S is a dominating set if state ρ_0 is not allowed for any vertex, giving the legal states $L = \{\rho_{\geq 1}, \sigma_{\geq 0}\}$. Optimization problems over these sets often maximize or minimize the cardinality of the set of vertices with states in a given $M \subseteq L$. For instance, in the minimum dominating set problem, $M = \{\sigma_{\geq 0}\}$.

Our notation	Standard terminology	$\exists[L]$	$max[L]$	$min[L]$
$[\rho_{\geq 0}, \sigma_0]$-set	Independent set	P	NPC	P
$[\rho_{\geq 1}, \sigma_{\geq 0}]$-set	Dominating set	P	P	NPC
$[\rho_{\leq 1}, \sigma_0]$-set	Strong Stable set or 2-Packing	P	NPC	P
$[\rho_1, \sigma_0]$-set	Efficient Dominating set or Perfect Code	NPC	NPC	NPC
$[\rho_{\geq 1}, \sigma_0]$-set	Independent Dominating set	P	NPC	NPC
$[\rho_1, \sigma_{\geq 0}]$-set	Perfect Dominating set	P	P	NPC
$[\rho_{\geq 1}, \sigma_{\geq 1}]$-set	Total Dominating set	P	P	NPC
$[\rho_1, \sigma_1]$-set	Total Perfect Dominating set	NPC	NPC	NPC
$[\rho_{\leq 1}, \sigma_{\geq 0}]$-set	Nearly Perfect set	P	P	P
$[\rho_{\leq 1}, \sigma_{\leq 1}]$-set	Total Nearly Perfect set	P	NPC	P
$[\rho_1, \sigma_{\leq 1}]$-set	Weakly Perfect Dominating set	NPC	NPC	NPC
$[\rho_{\geq 0}, \sigma_{\leq q}]$-set	Induced Bounded-Degree subgraph ($q \geq 0$)	P	NPC	P
$[\rho_{\geq q}, \sigma_{\geq 0}]$-set	q-Dominating set ($q \geq 1$)	P	P	NPC
$[\rho_{\geq 0}, \sigma_q]$-set	Induced q-Regular subgraph ($q \geq 0$)	P	NPC	P

Table I: Some vertex subset properties and the complexity of derived problems.

Our notation	Standard terminology	Complexity
$\exists[\rho_1, \sigma_0]$	Perfect Code Problem	NPC
$min[\rho_{\geq 1}, \sigma_{\geq 0}]$	Minimum Dominating Set Problem	NPC
$max[\rho_{\geq 0}, \sigma_0]$	Maximum Independent Set Problem	NPC
$min\{\rho_{\geq 0}\}[\rho_{\geq 0}, \sigma_0]$	Minimum Vertex Cover Problem	NPC
$max\{\rho_1\}[\rho_{\geq 0}, \sigma_{\geq 0}]$	Efficiency Problem	NPC

Table II: Examples of graph problems.

Definition 2. *Given sets M and L of vertex states and a graph G:*

- $S \subseteq V(G)$ *is an* $[L]$*-set if* $\forall v \in V(G) : state_S(v) \in L$
- $\exists[L]$ *is the problem deciding whether there exists any* $[L]$*-set* $S \subseteq V(G)$
- $minM[L]$ *(or* $maxM[L]$*) is the problem minimizing (or maximizing)* $|\{v : state_S(v) \in M\}|$ *over all* $[L]$*-sets* $S \subseteq V(G)$
- $min[L]$ *(or* $max[L]$*) is shorthand for* $minM[L]$ *(or* $maxM[L]$*) when* M *consists of all* σ*-states in* L*, in effect optimizing the cardinality of the selected set of vertices.*

Thus, a dominating set is a $[\rho_{\geq 1}, \sigma_{\geq 0}]$-set, with the square brackets implying the set notation. Table I shows some of the classical vertex subset properties and also the complexity of derived problems, with P denoting Polynomial time and NPC denoting NP-Complete. Most of these complexity results are old [10, 5, 8, 1, 6, 9, 4, 13], and others are among the results given in the next section. Properties traditionally defined using closed neighborhoods are easily captured by the characterization. Table II shows examples of graph problems [2, 18] expressed using the characterization. Note that complementary problems, e.g. Maximum Independent Set and Minimum Vertex Cover, are both expressible. Table I can be used as a quick reference guide to the exact definitions of the various properties represented and to the complexity of the associated problems. The characterization may also be useful when introducing new parameters.

3. Complexity Results

We are mainly interested in classifying problems admitting the given characterization as NP-complete or as solvable in polynomial time. The objective functions most studied in the past involve minimizing or maximizing the cardinality of the set of selected vertices, and for each entry in Table I, except Nearly Perfect Sets, there is at least one NP-complete problem related to such a parameter. In this paper we continue this trend, and the optimization problems we concentrate on are of the form $min[L]$ and $max[L]$. For certain subset properties, such as Perfect Code, it is well known that even deciding whether a graph has *any* such set is an NP-complete problem. In the following Lemma we observe several consequences of NP-completeness of an $\exists[L]$ problem.

LEMMA 1. *If $\exists[L]$ is NP-complete on a class of graphs C then any decision problems of the form $max[L]$, $min[L]$, $maxM[L]$, $minM[L]$ or $maxL[Q]$, $L \subseteq Q$ are NP-complete on C. Conversely, if any of the latter problems have a polynomial time algorithm, then so does $\exists[L]$.*

PROOF. The decision version of $maxM[L]$ takes a graph G and an integer k as input, and asks if G has an $[L]$-set S with $|\{v : state_S(v) \in M\}| \geq k$. Thus, with an algorithm for the decision version of $maxM[L]$, we can decide $\exists[L]$ by a single call of that algorithm providing the integer $k = 0$ as the second part of the input. With an algorithm for $minM[L]$ or $maxL[Q]$ problems we decide $\exists[L]$ using $k = |V(G)|$ for the input graph G as the second part of the input. □

In particular, Theorems 1,2,3,4 and 7 can each be combined with Lemma 1 to yield corollaries of this kind. We will not state these corollaries explicitly. We observe from Table I that vertex subset properties attracting most interest in the past are characterizable by two syntactic states (using the abbreviations) in which vertices have zero, one, at least zero, or at least one selected neighbors. Our focus in this paper continues this trend.

We show NP-completeness of several infinite classes of problems by reducing from the NP-complete problem Exact 3-Cover.

DEFINITION 3. *Exact 3-Cover (X3C)*
Instance: Set U and $T \subseteq \binom{U}{3}$.
Question: $\exists T' \subseteq T$, where T' a partition of U?

We introduce each NP-completeness result by way of a short comparison with the complexity of some related problem from Table I. In contrast to the NP-complete problem of deciding existence of $[\rho_1, \sigma_0]$-sets (Perfect Codes) our first result shows the NP-completeness of $\exists[L]$ problems with L containing an infinite number of legal states.

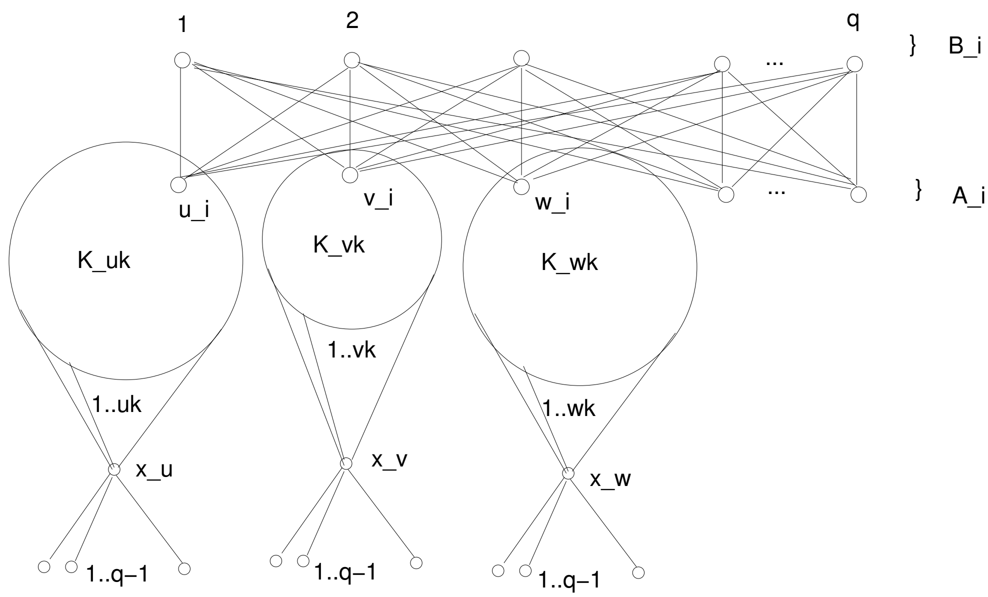

Fig. 1: Gadgets G_i, G_u, G_v, G_w for the triple $t_i = \{u, v, w\}$ with $q \geq 3$.

THEOREM 1. *The decision problems $\exists[\rho_{\geq q}, \sigma_0]$ are NP-complete for all $q \in \{2, 3, ...\}$.*

PROOF. We give a reduction from X3C to $\exists[\rho_{\geq q}, \sigma_0]$ for any $q \in \{2, 3, ...\}$. Given an instance (U, T) of X3C we construct a graph G such that $\exists T' \subseteq T$ with T' a partition of U if and only if G has a $[\rho_{\geq q}, \sigma_0]$-set S. Let $T = \{t_1, ..., t_{|T|}\}$. For each $u \in U$, let $T_u = \{t \in T : u \in t\} = \{t_{u1}, t_{u2}, ..., t_{uk}\}$ be the triples containing u. For each $u \in U$ the graph G will contain a subgraph G_u consisting of a complete graph on vertices $\{x_u, u_{u1}, u_{u2}, ..., u_{uk}\}$ and $q-1$ leaves L_u, each adjacent only to x_u. For each $t_i \in T$ with $t_i = \{u, v, w\}$ we construct a subgraph G_i sharing the vertices u_i, v_i, w_i with G_u, G_v, G_w, respectively, as follows: (case $q = 2$) G_i is a 6-cycle on vertices $A_i \cup B_i$ such that $A_i = \{u_i, v_i, w_i\}$ are mutually non-adjacent; (case $q \geq 3$) G_i is a complete bipartite graph $K_{q,q}$ with partition (A_i, B_i) and with $\{u_i, v_i, w_i\} \subseteq A_i$. This completes the description of G, see Figure 1.

Let S be a $[\rho_{\geq q}, \sigma_0]$-set of G. Note that every leaf in L_u must be in S, since ρ_0 and ρ_1 are not legal vertex states. In turn, their common neighbor x_u cannot be in S since σ_1 is not legal. Since $|L_u| = q-1$ and ρ_{q-1} is not legal at least one other neighbor of x_u, besides its L_u-neighbors, must be in S, i.e. $|\{u_{u1}, u_{u2}, ..., u_{uk}\} \cap S| \geq 1$. But $\{u_{u1}, u_{u2}, ..., u_{uk}\}$ induce a complete graph in G, and σ_0 is the only legal σ-state, so exactly one of these vertices must be in S. Let $u_i \in S$ with $t_i = \{u, v, w\}$. We would want t_i to cover u, v, w and show that indeed we must have $\{u_i, v_i, w_i\} \subseteq S$. Note that these three vertices are all in the same partition A_i of the bipartite graph G_i. We argue

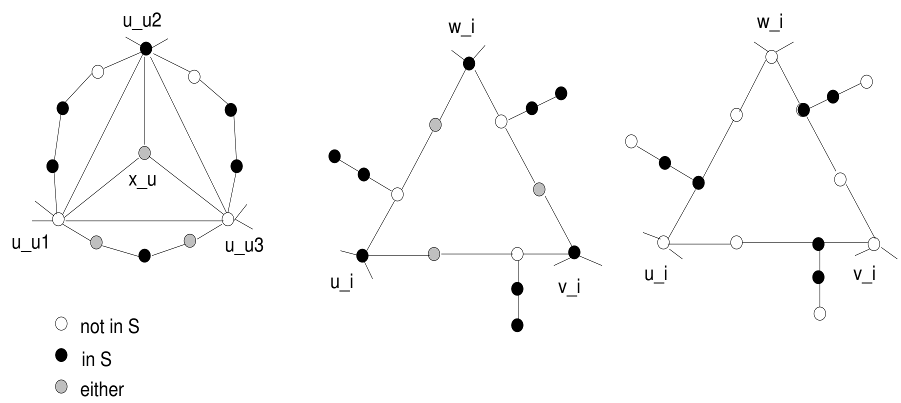

Fig. 2: NP-completeness of Dominating Induced Matchings. Left: Gadget G_u with $uk = 3$ and $u_{u2} \in S$. Middle: Gadget G_i for triple $t_i = \{u, v, w\}$ and $u_i, v_i, w_i \in S$. Right: The only other possibility for G_i is $u_i, v_i, w_i \notin S$.

first the case $q \geq 3$. No vertex in partition B_i can be in S since already $u_i \in S$ and σ_0 is the only legal σ-state. Moreover, since the neighborhood of any vertex in B_i is exactly A_i and $|A_i| = q$ we must have $A_i \subseteq S$ since ρ_k is not legal for any $k < q$. If $q = 2$ we have G_i a cycle and $u_i \in S$ again forces $A_i \subseteq S$. With this in mind, we have that $T' = \{t_i : A_i \subseteq S\}$ must be an exact 3-cover of U.

For the other direction, if $T' \subseteq T$ is an exact 3-cover of U, it is easy to check that $S = \{v : v \in L_u \wedge u \in U\} \cup \{v : v \in A_i \wedge t_i \in T'\} \cup \{v : v \in B_i \wedge t_i \notin T'\}$ is a $[\rho_{\geq q}, \sigma_0]$-set of G. NP-completeness of the $\exists[\rho_{\geq q}, \sigma_0]$ problem follows, since in polynomial time it is easy to verify a $[\rho_{\geq q}, \sigma_0]$-set and compute the transformation. □

In contrast to $[\rho_{\geq 1}, \sigma_0]$-sets (Independent Dominating sets) which are easily found using a greedy algorithm, our next result shows that $[\rho_{\geq 1}, \sigma_1]$-sets, which we call Dominating Induced Matchings, are difficult to find.

Theorem 2. *The decision problem* $\exists[\rho_{\geq 1}, \sigma_1]$ *(Dominating Induced Matching) is* NP*-complete.*

Proof. We again reduce from X3C and adopt all the notation from the proof of Theorem 1, constructing gadgets G_u and G_i sharing a vertex u_i if $u \in t_i \in T$. G_u will consist of a complete graph on the vertices $\{x_u, u_{u1}, u_{u2}, ..., u_{uk}\}$ and for each pair $u_i, u_j, i \neq j$ we add three new vertices and edges forming a 5-path from u_i through the new vertices to u_j. See Figure 2 which also shows the gadget G_i for $t_i = \{u, v, w\}$. Let S be a $[\rho_{\geq 1}, \sigma_1]$-set in the graph G thus constructed from an instance of X3C. We note right away that for any any vertex $v \in V(G)$ we have $N(v) \cap S \neq \emptyset$ since S is a dominating set and $v \in S \Leftrightarrow state(v) = \sigma_1$ so a neighbor of

v must be in S. Employing this argument to x_u of the gadget G_u shows that $|\{u_{u1}, u_{u2}, ..., u_{uk}\} \cap S| \geq 1$. Moreover, we cannot have $u_i, u_j \in S$ for $i \neq j$ since the middle vertex on the 5-path from u_i to u_j would have no S-neighbors. Hence, $|\{u_{u1}, u_{u2}, ..., u_{uk}\} \cap S| = 1$. The gadget G_i for a triple $t_i = \{u, v, w\}$ forces either $u_i, v_i, w_i \in S$ or $u_i, v_i, w_i \notin S$, see Figure 2. Thus, if we let T' be the triples t_i which have the shared vertices of G_i selected then T' must be an Exact 3-Cover of U. For the other direction of the proof, it is not hard to see from Figure 2 that an Exact 3-Cover of the instance (U, T) likewise gives rise to a $[\rho_{\geq 1}, \sigma_1]$-set in G. □

The $\exists[L]$ problem has trivially the affirmative answer if $\rho_0 \in L$. If L contains no ρ-states the $\exists[L]$-problem on G is solved by checking whether for each vertex v we have $\sigma_{deg_G(v)} \in L$. If L contains no σ-states the $\exists[L]$-problem on G is solved by checking whether $\rho_0 \in L$. In light of this, our next theorem gives a complete characterization, up to P vs. NP, of the complexity of $\exists[L]$ problems when L has a finite number of states. The reduction given is a generalization of a reduction used in [14].

THEOREM 3. *The $\exists[L]$ problem is NP-complete if $\rho_0 \notin L$ and L contains a finite positive number of both ρ-states and σ-states.*

PROOF. Let $L = \{\rho_{p_1}, \rho_{p_2}, ..., \rho_{p_m}, \sigma_{q_1}, \sigma_{q_2}, ..., \sigma_{q_n}\}$, where $n, m \geq 1$ and p_i, q_i non-negative integers satisfying $0 < p_1 < p_2 < ... < p_m$ and $q_1 < q_2 < ... < q_n$. We reduce from X3C. Given an instance (U, T) of X3C we want a graph G such that G has an $[L]$-set $S \subseteq V(G)$ if and only if $\exists T' \subseteq T$, a partition of U. The gadget for $u_i \in U$ is simply the vertex u_i, which will be shared by gadgets G_t for all triples t with $u_i \in t \in T$. The graph G will be defined by describing the gadgets G_t, one for each $t \in T$. For all $t = \{u_{t1}, u_{t2}, u_{t3}\} \in T$ we construct a graph G_t with private vertices P_t and shared vertices u_{t1}, u_{t2}, u_{t3}, i.e. $V(G_t) = P_t \cup \{u_{t1}, u_{t2}, u_{t3}\}$, having the property:

In the graph G_t, all $S \subseteq V(G_t)$ that assign $\forall v \in P_t$ a state $state_S(v) \in L$ assigns to u_{t1}, u_{t2}, u_{t3} either

(i) $state_S(u_{t1}) = state_S(u_{t2}) = state_S(u_{t3}) = \rho_0$ or

(ii) $state_S(u_{t1}) = state_S(u_{t2}) = state_S(u_{t3}) = \rho_{p_m}$.

Moreover, sets of type (i) and sets of type (ii) should exist for G_t.

Assuming we can construct such G_t, the theorem will follow:

Claim1: $G = \cup_{t \in T} G_t$ has $[L]$-set $S \subseteq V(G) \Leftrightarrow \exists T' \subseteq T$, a partition of U.

($\Leftarrow$:) Note the parts G_t of the graph G share only the vertices representing U. For each $t \in T'$ choose a set $S_t \subseteq V(G_t)$ of type (ii) for G_t. For each $t \notin T'$ choose a set $S_t \subseteq V(G_t)$ of type (i) for G_t. Let $S = \cup_{t \in T} S_t$.

($\Rightarrow$:) For any $[L]$-set S of G we must have $S \cap V(G_t)$ be either a set of type (i) or a set of type (ii) for G_t. This since only G_t contains the vertices P_t, and also $\{w : w \in N(v) \wedge v \in P_t\} \subseteq V(G_t)$. Since $\rho_0 \notin L$, and since a vertex $u \notin S$ can have at most p_m neighbors in S, we must have that $T' = \{t : V(G_t) \cap S$ is a set of type (ii) for $G_t\}$ is a partition of U.

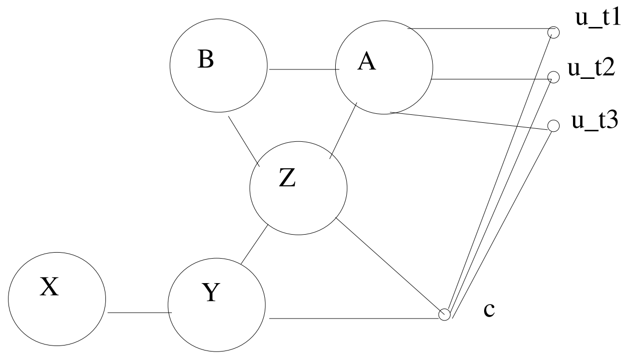

Fig. 3: A rough sketch of the components of G_t where the absence of a line between two components reflects the absence of an edge in G_t connecting any two vertices from those two components.

Construction of G_t: Let $V(G_t) = A \cup B \cup X \cup Y \cup Z \cup \{c\} \cup \{u_{t1}, u_{t2}, u_{t3}\}$. See Figure 3 for a rough sketch of how these components are connected together. As a preview, we mention that $\{A \cup Y\}$ will be a selected set of type (ii) and $\{B \cup Y\}$ a selected set of type (i) for G_t. X and Y will be such that a selected set must contain all vertices from Y but cannot contain any vertex from X. The vertex c, which cannot be selected, will be connected to enough vertices of Y so that none of its other neighbors, namely $Z \cup \{u_{t1}, u_{t2}, u_{t3}\}$, can be selected. The vertices Z will ensure that either all or none of the neighbors of u_{tk} are selected.

Let $A = A^1 \dot{\cup} ... \cup A^{p_m}$ and $B = B^1 \cup ... \cup B^{p_m}$ with $A^i = \{a^i_1, ..., a^i_{q_1+1}\}$ and $B^i = \{b^i_1, ..., b^i_{q_1+1}\}$, and let $G[A^i]$, $G[B^i], \forall i$ be complete graphs on $q_1 + 1$ vertices, with no other edges between As or between Bs. Edges connecting vertices of A with vertices of B are restricted to $(a^i_k, b^j_k), \forall i, j, k$. Edges incident with $\{u_{t1}, u_{t2}, u_{t3}\}$ in G_t are restricted to (c, u_{tk}) and $(a^i_1, u_{tk}), \forall i, k$.

We describe edges between X-vertices and Y-vertices. Let $\beta = \max\{p_m, q_n\} > 0$ and $\alpha = \lceil \frac{\beta}{(p_1(q_n+1))} \rceil > 0$. Let $Y = Y^1 \cup ... \cup Y^{p_1\alpha}$ and $G[Y^i], \forall i$, a complete graph on $q_n + 1$ vertices. Let $X = \{x_1, x_2, ..., x_{(q_n+1)(\beta+1)\alpha}\}$ with $G[X]$ containing no edges. We add edges connecting X-vertices with Y-vertices such that each vertex of X gets p_1 neighbors in Y and each vertex of Y gets $\beta + 1$ neighbors in X. This can be done since $|X| = \alpha(q_n + 1)(\beta + 1)$ and $|Y| = \alpha(q_n + 1)p_1$.

The vertex c is connected to p_m vertices of Y, note $|Y| \geq p_m > 0$, and c is also connected to every vertex of $Z \cup \{u_{t1}, u_{t2}, u_{t3}\}$.

It remains to describe the vertices and edges contributed by Z. Let $Z = Z^1 \cup Z^2 \cup Z^3 \cup \{z'\}$ with $Z^k = \{z^k_1, ..., z^k_{p_m}\}$ for $k \in \{1, 2, 3\}$. The vertex $z^1_i, \forall i$, is connected to a^1_1 and to b^i_1 and also has $p_1 - 1$ neighbors in Y. The vertex $z^2_i, \forall i$, is connected to a^1_1 and to b^i_1 and also has $p_m - 1$ neighbors in Y. The vertex $z^3_i, \forall i$, is connected to a^i_1 and to b^i_1 and also has $p_1 - 1$ neighbors in Y. The vertex z' is connected to $\{a^1_1, ..., a^{p_m}_1, b^1_1, ..., b^{p_m}_1\}$. This completes the description of G_t.

Claim2: $A \cup Y$ is a set of type (ii) and $B \cup Y$ is a set of type (i) for G_t.

Proof of claim: We consider $A \cup Y$ first. $G[A \cup Y]$ is a collection of p_m copies of K_{q_1+1} for the As and $p_1\alpha$ copies of K_{q_n+1} for the Ys, so $state_{A\cup Y}(a) = \sigma_{q_1}, \forall a \in A$ and $state_{A\cup Y}(y) = \sigma_{q_m}, \forall y \in Y$. Moreover, $\forall x \in X$ we have $N(x) \subseteq Y$ and $|N(x)| = p_1$ so $state_{A\cup Y}(x) = \rho_{p_1}$. For the vertex c we have $N(c) \subseteq Y \cup Z \cup \{u_{t1}, u_{t2}, u_{t3}\}$ and $|N(c) \cap Y| = p_m$, so $state_{A\cup Y}(c) = \rho_{p_m}$. The vertices $z \in Z^1 \cup Z^3$ have $|N(z) \cap \{A \cup Y\}| = p_1$, so $state_{A\cup Y}(z) = \rho_{p_1}$. Similarly, $\forall z \in Z^2$ we have $|N(z) \cap \{A \cup Y\}| = p_m$, so $state_{A\cup Y}(z) = \rho_{p_m}$. The vertex z' has $N(z') \subseteq A \cup B$ and $|N(z) \cap A| = p_m$, so $state_{A\cup Y}(z') = \rho_{p_m}$. So far, the argument for $B \cup Y$ being a set of type (i) can be obtained from the above by replacing B for A and vice-versa.

Since $\forall b \in B$, $N(b) \subseteq A \cup Z$ and $|N(b) \cap A| = p_m$, we have $state_{A\cup Y}(b) = \rho_{p_m}$. Similarly, $\forall a \in A$ we have $N(a) \subseteq B \cup Z \cup \{u_{t1}, u_{t2}, u_{t3}\}$ and $|N(a) \cap B| = p_m$, so $state_{B\cup Y}(a) = \rho_{p_m}$.

What remains is the argument for the vertices $\{u_{t1}, u_{t2}, u_{t3}\}$. We have for $k \in \{1, 2, 3\}$, $N(u_{tk}) = \{a_1^1, ..., a_1^{p_m}\}$, so $state_{A\cup Y}(u_{tk}) = \rho_{p_m}$ and $state_{B\cup Y}(u_{tk}) = \rho_0$ so that $A \cup Y$ is a set of type (ii) and $B \cup Y$ is a set of type (i), completing the proof of the claim.

Claim3: For any $S_t \subseteq V(G_t)$ which assigns $\forall w \in V(G_t) \setminus \{u_{t1}, u_{t2}, u_{t3}\}$ a state $state_{S_t}(w) \in L$, we have $Y \subseteq S_t$ and also $(Z \cup \{u_{t1}, u_{t2}, u_{t3}\}) \cap S_t = \emptyset$.

Proof of claim: $\forall y \in Y$ we have $|N(y) \cap X| = \beta + 1 > \max\{p_m, q_n\}$, so $\exists x \in N(y) : x \notin S_t$. But $|N(x) \cap Y| = p_1$, so $state_{S_t}(x) = \rho_{p_1}$ and $y \in S_t$. Since $|N(y) \cap Y| = q_n$ we must have $state_{S_t}(y) = \sigma_{q_n}$. Since $|N(c) \cap Y| = p_m$ we must have $state_{S_t}(c) = \rho_{p_m}$ and $(Z \cup \{u_{t1}, u_{t2}, u_{t3}\}) \cap S_t = (N(c) \setminus Y) \cap S_t = \emptyset$, completing the proof of the claim.

Claim4: For any $S_t \subseteq V(G_t)$ which assigns $\forall w \in V(G_t) \setminus \{u_{t1}, u_{t2}, u_{t3}\}$ a state $state_{S_t}(w) \in L$, we have either $a_1^i \in S_t, 1 \leq i \leq p_m$ or $a_1^i \notin S_t, 1 \leq i \leq p_m$.

Proof of claim: From Claim3 we have $Z \cap S_t = \emptyset$ and $Y \subseteq S_t$. In particular, $state_{S_t}(z_i^1) \in \{\rho_{p_1}, \rho_{p_1+1}\}$, similarly $state_{S_t}(z_i^2) \in \{\rho_{p_m-1}, \rho_{p_m}\}$ and $state_{S_t}(z_i^3) \in \{\rho_{p_1}, \rho_{p_1+1}\}$. In turn, we consider the two cases $a_1^1 \in S_t$ and $a_1^1 \notin S_t$. $a_1^1 \in S_t$ gives $state_{S_t}(z_i^2) = \rho_{p_m}, \forall i$, so $b_1^i \notin S_t, \forall i$. This in turn gives $state_{S_t}(z_i^3) = \rho_{p_1}$ so $a_1^i \in S_t, \forall i$, completing the first case. $a_1^1 \notin S_t$ implies $b_1^i \in S_t, \forall i$ so that $state_{S_t}(z_i^1) = \rho_{p_1}, \forall i$. This in turn gives $state_{S_t}(z') = \rho_{p_m}$ so $a_1^i \notin S_t, \forall i$, completing the proof of the claim.

Each of $\{u_{t1}, u_{t2}, u_{t3}\}$ is adjacent to exactly $\{a_1^1, ..., a_1^{p_m}\}$ and by Claim3 cannot be in S_t. Hence, Claim4 actually shows that any $S_t \subseteq V(G_t)$ which assigns $\forall w \in V(G_t) \setminus \{u_{t1}, u_{t2}, u_{t3}\}$ a state $state_{S_t}(w) \in L$ has either

(i) $state_{S_t}(u_{t1}) = state_{S_t}(u_{t2}) = state_{S_t}(u_{t3}) = \rho_0$, or

(ii) $state_{S_t}(u_{t1}) = state_{S_t}(u_{t2}) = state_{S_t}(u_{t3}) = \rho_{p_m}$.

Thus G_t has the claimed properties and the theorem follows. □

As our next theorem shows, some of these decision problems are NP-complete even for very restricted classes of graphs. The reduction we use is a simple special case of the one just given, and uses the NP-complete problem Planar 3-Dimensional Matching (P3DM). A similar reduction is used in [8].

DEFINITION 4. *3-Dimensional Matching (3DM)*
Instance: Disjoint sets U_1, U_2, U_3 *with* $U = U_1 \cup U_2 \cup U_3$ *and* $T \subseteq U_1 \times U_2 \times U_3$*.*
Question: $\exists T' \subseteq T$*, where* T' *a partition of* U*?*

With an instance I of 3DM, we associate the bipartite graph G_I where $V(G_I) = U \cup T$ and $E(G_I) = \{(u,t) : u \in U \wedge u \in t \in T\}$. In [7] it is shown that the Planar 3DM problem, 3DM restricted to instances where G_I is planar, is still NP-complete.

THEOREM 4. *The problem of deciding whether a planar bipartite graph of maximum degree three has any* $[\rho_1, \sigma_1]$*-set (Total Perfect Dominating Set) is* NP*-complete.*

PROOF. Given an instance I of P3DM, we construct a graph G having a $[\rho_1, \sigma_1]$-set if and only if $\exists T' \subseteq T$, a partition of U. Let G be the graph G_I augmented by adding, for each $t \in T$, the vertices a_t and b_t, and edges connecting a_t to both t and b_t. Since this reduction does not distinguish between the sets U_1, U_2, U_3, the instance I can be viewed as an instance of X3C, and the argument that G has a $[\rho_1, \sigma_1]$-set if and only if $\exists T' \subseteq T$, a partition of U, is left out since it is in easy analogy with the argument used for the previous theorem.

Note that G_I and G are both planar bipartite graphs. We next show an easy transformation of a graph G having a vertex of degree larger than three to a graph G' with the following properties:

(i) if G planar and bipartite then G' planar and bipartite,
(ii) $\Sigma_{\{v:deg_G(v)\geq 4\}} deg_G(v) > \Sigma_{\{v:deg_{G'}(v)\geq 4\}} deg_{G'}(v)$
(iii) G has a $[\rho_1, \sigma_1]$-set if and only if G' has a $[\rho_1, \sigma_1]$-set.

Hence, applying such a polytime transformation repeatedly, starting with G, until the resulting graph has no vertices of degree larger than three, yields a graph proving the theorem.

We define the transformation by describing the resulting graph G'. Let v be a distinguished vertex of G with $N_G(v) = \{v_1, v_2, ..., v_k\}$ and $k \geq 4$. Let G' have vertices $V(G') = V(G) \cup \{w, x, y, z\}$ and edges $E(G') = E(G) \setminus \{(v_1, v), (v_2, v)\} \cup \{(v_1, w), (v_2, w), (w, x), (x, y), (y, z), (z, v)\}$. See Figure 4. Note the transformation is local, with changes only to the neighborhoods of v_1, v_2 and v.

We prove the stated properties of the transformation:

(i) Planarity is obviously preserved. If A, B is an appropriate bipartition of $V(G)$ then w.l.o.g. we must have $v \in A$, $N(v) \subseteq B$ so that $A \cup \{w, y\}$ and $B \cup \{x, z\}$ forms an appropriate bipartition of $V(G')$. (ii) The new vertices all have degree less than 4, whereas the degree of v decreases to

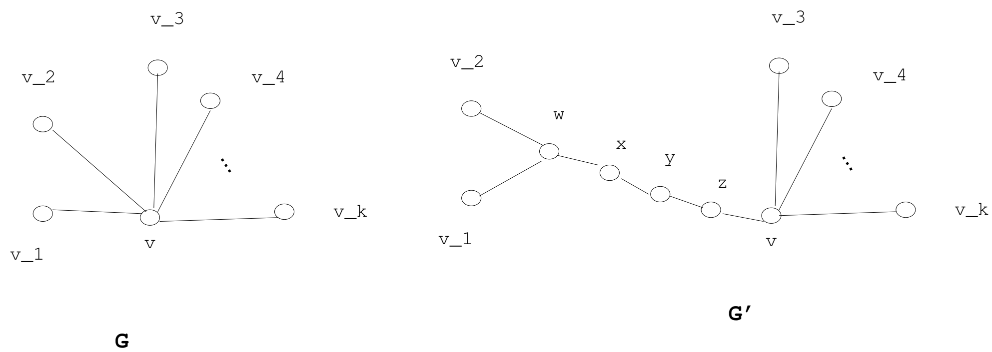

Fig. 4: Local transformation of G at vertex v to the graph G', used to construct a planar bipartite graph of maximum degree 3

$k-1$. (iii) Let S and S' be $[\rho_1, \sigma_1]$-sets in G and G', respectively. Note that $\{w, x, y, z, v\}$ induces a 5-path in G' so there are 4 possibilities for $\{w, x, y, z, v\} \cap S'$, namely $\{y, z\}, \{w, z, v\}, \{w, x, v\}$ and $\{x, y\}$. We similarly split the possibilities for choice of S into 4 classes, namely
$|\{v_1, v_2\} \cap S| = 1 \wedge v \notin S \wedge |\{v_3, ..., v_k\} \cap S| = 0$,
$|\{v_1, v_2\} \cap S| = 1 \wedge v \in S \wedge |\{v_3, ..., v_k\} \cap S| = 0$,
$|\{v_1, v_2\} \cap S| = 0 \wedge v \in S \wedge |\{v_3, ..., v_k\} \cap S| = 1$,
$|\{v_1, v_2\} \cap S| = 0 \wedge v \notin S \wedge |\{v_3, ..., v_k\} \cap S| = 1$.
It is easy to check that the 4 possibilities for choice of S' have, in the order given, characterizations in terms of effect on v and $N(v)$ which are identical to those just given for S, and indeed property (iii) holds. □

To our knowledge, the complexity of problems defined over Total Perfect Dominating Sets in graphs, had not been studied previously [5].

Combining Lemma 1 with Theorem 4 gives the NP-completeness on planar bipartite graphs of maximum degree three of the problem $max\{\rho_1, \sigma_1\}$ $[\rho_{\geq 0}, \sigma_{\geq 0}]$, which we call Total Efficiency. This problem arises in communication networks, if we assume that a communication round has two time-disjoint phases, *send* and *receive*, and that a processor receives a message whenever it has a single sending neighbor. The maximum number of processing elements that can receive a message in one communication round is the Total Efficiency of the graph underlying the network topology.

The following strong result is due to Kratochvíl.

THEOREM 5. *[14] The problem of deciding whether a planar 3-regular graph has a $[\rho_1, \sigma_0]$-set (perfect code) is NP-complete.*

We state the implications of this result for some other problems admitting our characterization.

COROLLARY 1. *Any decision problem of the form $min[L]$ with $\rho_0 \notin L$ and $\{\rho_1, \sigma_0\} \subseteq L$ is NP-complete on planar 3-regular graphs.*

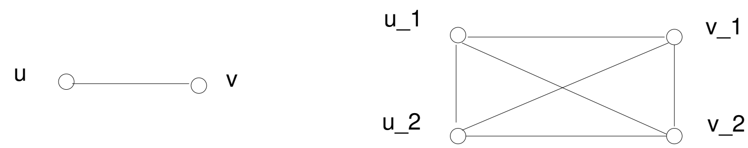

Fig. 5: Given G on the left, the reduction constructs G' on the right

Proof. Let G be a planar 3-regular graph. We show that G has a perfect code if and only if the value of $min[L]$ on G is $|V(G)|/4$. Since every vertex of G has degree 3, a perfect code of G has cardinality $|V(G)|/4$ and is clearly a dominating set. Moreover, a dominating set of G which is not a perfect code will have more than $|V(G)|/4$ vertices. An $[L]$-set in G is a dominating set since ρ_0 is not legal and it could be a perfect code since ρ_1 and σ_0 are legal. The corollary follows. □

While every graph has an Independent Dominating Set ($[\rho_{\geq 1}, \sigma_0]$-set), that can be easily found by a greedy algorithm, it is well-known that both minimizing and maximizing the cardinality of such a set is NP-hard. Our next result shows another vertex subset property with this complexity classification.

Theorem 6. *The decision problems* $min[\rho_{\geq 1}, \sigma_{\leq 1}]$ *and* $max[\rho_{\geq 1}, \sigma_{\leq 1}]$ *are both* NP*-complete, while* $\exists[\rho_{\geq 1}, \sigma_{\leq 1}]$ *is easy.*

Proof. Any graph has a $[\rho_{\geq 1}, \sigma_{\leq 1}]$-set, take for example a $[\rho_{\geq 1}, \sigma_0]$-set. NP-completeness of $min[\rho_{\geq 1}, \sigma_{\leq 1}]$ follows from Corollary 1. We show NP-completeness of $max[\rho_{\geq 1}, \sigma_{\leq 1}]$ by reduction from $max[\rho_{\geq 1}, \sigma_0]$. Given a graph G, construct the graph G' with $V(G') = \{u_1, u_2 : u \in V(G)\}$ and $E(G') = \{(u_1, u_2) : u \in V(G)\} \cup \{(u_1, v_1), (u_2, v_2), (u_1, v_2), (u_2, v_1) : (u, v) \in E(G)\}$, see Figure 5. Let S be a maximum-size $[\rho_{\geq 1}, \sigma_0]$-set in G and let S' be a maximum-size $[\rho_{\geq 1}, \sigma_{\leq 1}]$-set in G'. We show that $2|S| = |S'|$. Let A be $[\rho_{\geq 1}, \sigma_0]$ in G. Then $A' = \{u_1, u_2 : u \in S\}$ is $[\rho_{\geq 1}, \sigma_{\leq 1}]$ in G'. We have $2|A| = |A'|$, so this shows that $2|S| \leq |S'|$. Let B' be $[\rho_{\geq 1}, \sigma_{\leq 1}]$ in G', with $C = \{(u_i, v_j) \in E(G') : \{u_i, v_j\} \subseteq B'\}$, the edges of $G'[B']$. Choose one endpoint of each edge from C and call this set of vertices D. Define $B = \{v \in V(G) : state_{B'}(v_1) = \sigma_0 \vee state_{B'}(v_2) = \sigma_0 \vee v_1 \in D \vee v_2 \in D\}$. Since we have removed one endpoint from each edge of $G'[B']$ it is clear that B is an independent set in G and $2|B| \geq |B'|$. In our notation, B is $[\rho_{\geq 0}, \sigma_0]$ in G, and can be greedily augmented to a $[\rho_{\geq 1}, \sigma_0]$-set, which shows that $2|S| \geq |S'|$. The transformation is easily computed in polynomial time, and the theorem follows. □

We now turn to problems with an easy solution algorithm, and focus our attention on optimization problems. Based on Lemma 1 such results have as corollaries the polynomial-time solvability of the associated existence problems.

THEOREM 7. *The problem $max[L]$ is solvable in polynomial time by a greedy algorithm if $\sigma_{\geq k}$ is the only σ-state in L and either (i), (ii), (iii) or (iv) holds*
(i) $\rho_1, \rho_2 ..., \rho_{k-1} \in L$
(ii) $\rho_0, \rho_1, ..., \rho_{k-1} \notin L$
(iii) $\rho_{\geq h}$ *is the only* ρ*-state in* L*, for some* h
(iv) ρ_0 *and* $\rho_{\geq h}$ *are the only* ρ*-states in* L*, for some* h

PROOF. We give two greedy algorithms, named ALG1 and ALG2, with input a graph G and output a set achieving $max[L]$ for G, if any $[L]$-set exists. ALG2 is used in case (iv) when $h \geq 2$ in which case there is a crucial gap in the legal ρ-states while ALG1 is used in the remaining cases. The algorithms use data structures $B\sigma$,$B\rho$ of type *set*.

ALG1(G)
$B\sigma$, $B\rho := V(G)$, $\emptyset$;
while (I: $\exists v \in B\sigma : |N(v) \cap B\sigma| < k$) do $B\sigma$, $B\rho := B\sigma \setminus \{v\}$, $B\rho \cup \{v\}$;
if ($\exists v \in B\rho : state_{B\sigma}(v) \notin L$) then output($\nexists[L]$-set) else output($B\sigma$);

ALG2(G)
$B\sigma$, $B\rho := V(G)$, $\emptyset$;
while (I: $\exists v \in B\sigma : |N(v) \cap B\sigma| < k$) or (II: $\exists w \in B\rho : |N(w) \cap B\sigma| < h$) do
    Case I : $B\sigma$, $B\rho := B\sigma \setminus \{v\}$, $B\rho \cup \{v\}$;
    Case II: $B\sigma$, $B\rho := B\sigma \setminus \{N(w) \cap B\sigma\}$, $B\rho \cup \{N(w) \cap B\sigma\} \setminus \{w\}$;
output($B\sigma$);

We first prove correctness, for both algorithms, of the loop invariant: "A vertex $v \notin B\sigma$ cannot be a member of any $[L]$-set S of G." The loop invariant is true initially since $B\sigma = V(G)$. Let $B\sigma$ and $B\sigma'$ be the values before and after an iteration of the loop. ¿From the loop invariant we have $S \subseteq B\sigma$ and show that $S \subseteq B\sigma'$.

Case I (both algorithms): $B\sigma \setminus B\sigma' = \{v\}$ and $v \in B\sigma : |N(v) \cap B\sigma| < k$. Since $\sigma_{\geq k}$ is the only σ-state in L, $S \subseteq B\sigma$ cannot contain v.

Case II (ALG2 only, i.e. ρ_0 and $\rho_{\geq h}$ the only legal ρ-states): $v \in B\sigma \setminus B\sigma'$ and $\exists w : v \in N(w)$ where $w \in B\rho$ and $|N(w) \cap B\sigma| < h$. When a vertex is added to $B\rho$ it is also removed from the non-growing set $B\sigma$ so that $B\rho \cap B\sigma = \emptyset$ and in particular $w \notin S$. Since $\rho_1, \rho_2, ..., \rho_{h-1} \notin L$ we must have $state_S(w) = \rho_0$ so that $N(w) \cap S = \emptyset$. Since $v \in N(w)$ this completes the proof of the loop invariant.

At termination of both algorithms all vertices in $B\sigma = S$ have at least k neighbors in S. We first argue correctness of ALG1. At termination of ALG1 all vertices not in S (in $B\rho$) have less than k neighbors in S, so if for some $v \in B\rho$ we have $state_S(v) = \rho_i \notin L$ there cannot be any $[L]$-set in G since for no $j < i$ is $\rho_j \in L$. However, if such a vertex does not exist S is a maximum-size $[L]$-set, proving correctness of ALG1. At termination of ALG2 all vertices not in $B\sigma = S$ have either at least h neighbors in S

(these vertices are in $B\rho$) or no neighbors in S. Since ρ_0 and $\rho_{\geq h}$ are both legal ρ-states we have S a maximum $[L]$-set, and ALG2 is correct. □

Minimization problems of the form $min[L]$ have the empty vertex subset as solution if $\rho_0 \in L$. Similarly, if L has no σ-states then the empty vertex subset is the only possible solution. If L has no ρ-states then the only possible $[L]$-set in a graph G is $V(G)$ which is checked by degree computation as described earlier. A $min[L]$ problem where L does not satisfy any of the above is asking for a minimum-size dominating set S of a certain kind. We have reason to believe that finding such a set is, in general, NP-hard.

CONJECTURE 1. *Assuming $P \neq NP$ the decision problem $min[L]$ is NP-complete if and only if $\rho_0 \notin L$ and L contains both some ρ-state and some σ-state.*

4. Conclusions

We have given a characterization of domination-type problems and investigated their computational complexity. The results given are a step towards our goal of a complete complexity classification of the problems admitting the characterization.

The given characterization can be generalized in several ways. Vertex weighted versions of $maxM[L]$ problems optimize the sum of the weights of vertices with state in M, the cardinality corresponding to unit weights. For directed graphs we consider $N_G(v)$ as $\{u : \langle u, v\rangle \in \text{Arcs}(G)\}$ to obtain directed versions of these domination-type properties and parameters. An extension of this characterization will encompass also parameters related to irredundant vertex subsets, and also to maximal and minimal versions of the vertex subsets given here, see [15]. Another extension of the characterization will encompass parameters related to partitioning of the vertices into several $[L]$-sets, e.g. domatic number, chromatic number, H-covering, see [16].

In another paper [17], we give practical algorithms on partial k-trees (graphs of treewidth bounded by k) solving any problem admitting the given characterization. A measure of the complexity of the resulting algorithm solving a problem with legal states L is the set A_L (a superset of L) of states needed for algorithmic purposes and its syntactic size $|A_L|$. Suffice it to say that any problem derived from Table I (with $q \leq 2$) has $|A_L| \leq 4$.

THEOREM 8. *[17] For any problem admitting the given characterization having legal states L there is an algorithm which takes a graph G with n vertices and a width k tree-decomposition of G as input, and gives a solution for G in $\mathcal{O}(n|A_L|^{2k})$ steps.*

Certain vertex subset properties, such as Perfect codes, have the interesting feature that any such set in a graph has the same cardinality. In [15] we give a theorem characterizing exactly those vertex subset properties having this feature

THEOREM 9. *[15] The statement "For any graph G, all $[L]$-sets have the same cardinality" is true if and only if (i) or (ii) holds*
(i) $L = \{\rho_p, \sigma_q\}$ for some $p \in \{1, 2, ...\}, q \in \{0, 1, ...\}$
(ii) L has either no ρ-states or no σ-states

Acknowledgements

Thanks to my advisor Andrzej Proskurowski for many ideas and to the referees for their useful remarks.

References

[1] D.W.Bange, A.E.Barkauskas and P.J.Slater, Efficient dominating sets in graphs, in: Ringeisen and Roberts, eds., *Applications of Discrete Mathematics*, SIAM, (1988).
[2] P.J.Bernhard, S.T.Hedetniemi and D.P.Jacobs, Efficient sets in graphs, *Discrete Applied Mathematics* 44 (1993).
[3] J.A.Bondy and U.S.R.Murty, *Graph theory with applications*, 1976.
[4] K. Cameron, Induced Matchings, *Discrete Applied Mathematics* 24 (1989), 97-102.
[5] E.J.Cockayne, B.L.Hartnell, S.T.Hedetniemi and R.Laskar, Perfect domination in graphs, manuscript (1992), to appear in Special issue of JCISS.
[6] E.J.Cockayne and S.T.Hedetniemi, Towards a theory of domination in graphs, *Networks*, 7 (1977), 211-219.
[7] M.E.Dyer and A.M.Frieze, Planar 3DM is NP-complete, *Journal of Algorithms*, vol.7, 174-184, (1983).
[8] M.Fellows and M.Hoover, Perfect domination, *Australian J. Combinatorics* 3 (1991), 141-150.
[9] J.F.Finck and M.S.Jacobson, On n-domination, n-dependence and forbidden subgraphs, in *Graph Theory with Applications to Algorithms and Computer Science*, Wiley (1984) 301-312.
[10] M.Garey and D.Johnson, *Computers and Intractability*, Freeman, San Fransisco, 1979.
[11] S.T.Hedetniemi and R.Laskar, Recent results and open problems in domination theory, in: Ringeisen and Roberts, eds., *Applications of Discrete Mathematics* (SIAM, 1988).
[12] S.T.Hedetniemi and R.Laskar, Bibliography on domination in graphs and some basic definitions of domination parameters, *Discrete Mathematics* 86 (1990).
[13] M.S.Jacobson, K.Peters and D.R.Fall, On n-irredundance and n-domination, *Ars Combinatoria* 29B (1990) 151-160.
[14] J.Kratochvíl, Perfect codes in general graphs, monograph, Academia Praha (1991).
[15] J.A.Telle, Characterization of domination-type parameters in graphs, in *Proceedings 24th SouthEastern Conference on Combinatorics, Graph Theory and Computing, Congressus Numerantium vol. 94 (1993) 9-16.*
[16] J.A.Telle, Vertex Partitioning Problems: Characterization, Complexity and Algorithms on Partial k-Trees, PhD thesis, University of Oregon CIS Department Technical Report TR-94-18.
[17] J.A.Telle and A.Proskurowski, Practical algorithms on partial k-trees with an application to domination-type problems, in *Proceedings WADS'93*, LNCS vol. 709 (1993) 610-621.
[18] J.A.Telle and A.Proskurowski, Efficient sets in partial k-trees, *Discrete Applied Mathematics 44* (1993) 109-117.