On the algorithmic complexity of k-tuple total domination

James K. Lan and Gerard Jennhwa Chang

May 29, 2013
On the algorithmic complexity of $k$-tuple total domination

James K. Lan$^{a,*}$, Gerard Jennhwa Chang$^{a,b,c}$

$^a$Department of Mathematics, National Taiwan University, Taipei 10617, Taiwan
$^b$Taida Institute for Mathematical Sciences, National Taiwan University, Taipei 10617, Taiwan
$^c$National Center for Theoretical Sciences, Taipei Office, Taiwan

Abstract

For a fixed positive integer $k$, a $k$-tuple total dominating set of a graph $G$ is a subset $D \subseteq V(G)$ such that every vertex of $G$ is adjacent to at least $k$ vertices in $D$. The $k$-tuple total domination problem is to determine a minimum $k$-tuple total dominating set of $G$. This paper studies $k$-tuple total domination from an algorithmic point of view. In particular, we present a linear-time algorithm for the $k$-tuple total domination problem for graphs in which each block is a clique, a cycle or a complete bipartite graph, which include trees, block graphs, cacti and block-cactus graphs. We also establish NP-completeness of the $k$-tuple total domination problem in undirected path graphs.

Keywords: $k$-Tuple total domination, Total domination, Block graph, Cactus, Algorithm, NP-complete, Undirected path graph

1. Introduction

All graphs in this paper are simple, i.e., finite, undirected, loopless and without multiple edges. Domination is a well studied subject in graph theory and combinatorial optimization as it has many applications in the real world such as location problems, sets of representatives, social network theory, etc.

$^*$This research was partially supported by the National Science Council of the Republic of China under grants NSC100-2811-M-002-146 and NSC101-2115-M-002-005-MY3.

$^*$Corresponding author.

Email addresses: drjamesblue@gmail.com (James K. Lan),
gjchang@math.ntu.edu.tw (Gerard Jennhwa Chang)

Preprint submitted to Elsevier May 29, 2013
and the literature on this topic has been surveyed and detailed in books
[2, 5, 6]. A dominating set of a graph $G$ is a subset $D \subseteq V(G)$ such that
every vertex not in $D$ has at least one neighbor in $D$. The domination number $\gamma(G)$ of $G$ is the minimum cardinality of a dominating set of $G$. The domination problem is to find a minimum dominating set of a graph.

The idea of dominating all vertices of the graph, rather than merely dominating vertices outside the set, is considered by Cockayne, Dawes and Hedetniemi [3]. A total dominating set of a graph $G$ is a subset $D \subseteq V(G)$ such that every vertex of $G$ has at least one neighbor in $D$. Every graph without isolated vertices has a total dominating set, since the vertex set $V(G)$ is such a set. The total domination number $\gamma_t(G)$ of $G$ is the minimum cardinality of a total dominating set of $G$. Total domination is now a well-studied topic in graph theory; see the recent survey paper [7] for more details.

Another variation of domination, the $k$-tuple domination was introduced by Harary and Haynes [4]. For a fixed positive integer $k$, a $k$-tuple dominating set of a graph $G$ is a subset $D \subseteq V(G)$ such that the closed neighborhood of every vertex of $G$ has at least $k$ vertices in $D$. The $k$-tuple domination number $\gamma_k(G)$ of $G$ is the minimum cardinality of a $k$-tuple dominating set of $G$. The $k$-tuple domination number is only defined for graphs with minimum degree at least $k - 1$. The special case when $k = 1$ is the usual domination.

Motivated by the concept of $k$-tuple domination, Henning and Kazemi [8] considered the following generalization of total domination. For a fixed positive integer $k$, a $k$-tuple total dominating set of a graph $G$ is a subset $D \subseteq V(G)$ such that every vertex of $G$ has at least $k$ neighbors in $D$. Every graph with minimum degree at least $k$ admits a $k$-tuple total dominating set. The $k$-tuple total domination number $\gamma_{x,k,t}(G)$ of $G$ is the minimum cardinality of a $k$-tuple total dominating set of $G$. The $k$-tuple total domination problem is to determine a minimum $k$-tuple total dominating set of a graph.

Since a $(k + 1)$-tuple dominating set is also a $k$-tuple total dominating set and a $k$-tuple total dominating set is also a $k$-tuple dominating set, we have $\gamma_{x,k}(G) \leq \gamma_{x,k,t}(G) \leq \gamma_{x,(k+1)}(G)$ for graphs with minimum degree at least $k$. The 1-tuple total domination is the well-studied total domination. The 2-tuple total domination is called the double total domination in the literature. Authors in [8, 9, 10] have established bounds on the number $\gamma_{x,k,t}(G)$ in terms of different graph invariants.

On the complexity side of the $k$-tuple total domination problem, Pradhan [14] showed that the $k$-tuple total domination problem is NP-complete
for bipartite graphs and for split graphs (and thus the chordal graphs). This problem remains NP-complete for doubly chordal graphs, another subclass of chordal graphs [14]. Apart from these, Pradhan also proposed some hardness results and approximation algorithms for this problem. On the other hand, Pradhan proved that the \(k\)-tuple total domination problem for chordal bipartite graphs is a subproblem of the \(k\)-tuple domination problem for strongly chordal graphs, which is solvable in polynomial time [13]. Therefore the \(k\)-tuple total domination problem for chordal bipartite graphs is polynomially solvable.

In this paper, we explore efficient algorithms for the \(k\)-tuple total domination problem in graphs. In particular, we present a linear-time algorithm for the \(k\)-tuple total domination problem in graphs in which each block is a clique, a cycle or a complete bipartite graph. This class of graphs include trees, block graphs, cacti and block-cactus graphs. Since the \(k\)-tuple total domination is a generalization of the total domination, our algorithms cover the partial results in [1, 7, 12]. Moreover, we also show that the \(k\)-tuple total domination problem remains NP-complete for undirected path graphs, another subclass of chordal graphs.

2. Preliminaries

Let \(G = (V, E)\) be a graph with vertex set \(V\) and edge set \(E\). For a vertex \(v\), the open neighborhood is the set \(N_G(v) = \{ u \in V : uv \in E \}\) and the closed neighborhood is \(N_G[v] = N_G(v) \cup \{ v \}\). The degree \(\text{deg}_G(v)\) of a vertex \(v\) in \(G\) is the number of edges incident to \(v\). When the graph \(G\) is clear from the context, it is dropped from notations. The minimum degree among the vertices of \(G\) is denoted by \(\delta(G)\). An isolated vertex is a vertex \(v\) with \(\text{deg}(v) = 0\). A leaf of a graph is a vertex with degree one. The subgraph of \(G\) induced by \(S \subseteq V\) is the graph \(G[S]\) with vertex set \(S\) and edge set \(\{ uv \in E : u, v \in S \}\). In a graph \(G = (V, E)\), the deletion of \(S \subseteq V\) from \(G\), denoted by \(G - S\), is the graph \(G[V \setminus S]\). For a vertex \(v\) in \(G\), we write \(G - v\) for \(G - \{ v \}\).

In a graph, an independent set is a set of pairwise nonadjacent vertices; a clique is a set of pairwise adjacent vertices. A tree is a connected graph without cycles. A vertex \(v\) is a cut-vertex if the number of connected components is increased after removing \(v\). A block of a graph is a maximal connected subgraph without any cut-vertex. An end-block of a graph is a block containing at most one cut-vertex of the graph. A block graph is a graph whose
blocks are cliques. A cactus is a connected graph whose blocks are either an edge or a cycle. A cactus is a tree if all the blocks are edges. A block-cactus graph is a graph whose blocks are cliques or cycles.

A graph is an intersection graph if there is a correspondence between its vertices and a family of sets (the intersection model) such that two distinct vertices are adjacent in the graph if and only if their two corresponding sets have a nonempty intersection. A graph $G$ is chordal if $G$ has no induced cycle with length greater than 3. It is well-known that a graph is chordal if and only if it is the intersection graph of some subtrees of a certain tree. If these subtrees are paths, this chordal graph is called an undirected path graph.

3. Block-wise approach for $k$-tuple total domination

The main result of this section is an algorithm for the $k$-tuple total domination problem in graphs. Actually our algorithm solves a slightly more general problem, which will be formulated as “$L$-domination”.

3.1. Labeling method for $k$-tuple total domination

Labeling techniques are widely used in the literatures for solving the domination problem and its variants [2, 11, 12, 13]. For $k$-tuple total domination, we employ the following labeling method which is similar to that in [11]. Given a graph $G$, labeling $L$ is a mapping that assigns each vertex $v$ in $G$ a two-tuple label $L(v) = (L_1(v), L_2(v))$, where $L_1(v) \in \{B, R\}$, and $L_2(v)$ is a nonnegative integer. Here a vertex $v$ with $L_1(v) = R$ is called a required vertex; a vertex $v$ with $L_1(v) = B$ is called a bound vertex. An $L$-dominating set of $G$ is a subset $D \subseteq V(G)$ such that

- if $L_1(v) = R$, then $v \in D$, and
- for each $v \in V(G)$, $|N_G(v) \cap D| \geq L_2(v)$.

That is, $D$ contains all required vertices, and for each vertex $v$ of $G$, $v$ is adjacent to at least $L_2(v)$ vertices in $D$. The $L$-domination number $\gamma_L(G)$ is the minimum cardinality of an $L$-dominating set in $G$, such set is called a $\gamma_L$-set of $G$. Clearly $G$ has an $L$-dominating set if and only if $L_2(v) \leq \deg_G(v)$ for all $v \in V(G)$, and we call such $L$ a proper labeling. Notice that if $L(v) = (B, k)$ for all $v \in V(G)$, then $\gamma_L(G) = \gamma_{\times k,t}(G)$. Thus an algorithm for $\gamma_L(G)$
gives $\gamma_{x,k,t}(G)$. In the following, we give a general approach (block-wise) to find a minimum $L$-dominating set in a graph.

Suppose $G$ is a graph with a proper labeling $L = (L_1, L_2)$. Let $C$ be an end-block of $G$ and $x$ be its unique cut-vertex. Let $G'$ denote the graph which results from $G$ by deleting all vertices only in $C$. Suppose $\ell$ is a nonnegative integer such that $\ell \leq \deg_C(x)$. The following notations will be used throughout the rest of this section.

- $L^\ell$: the restriction of $L$ on $C$ with the modification on $L^\ell_2(x) = \ell$.
- $L^{\ell R}$: the same as $L^\ell$ except for the modification on $L^{\ell R}_1(x) = R$.

Since the value $L_2(x)$ may be greater than the number of neighbors of $x$ in $G$, we need to set the $L_2$ value of $x$ as the minimum between $L_2(x)$ and $\deg_C(x)$ before evaluating the cardinality of a $\gamma_L$-set of $C$. Thus, for convenience, set

$$\alpha = \min\{L_2(x), \deg_C(x)\}.$$ 

Note that one always have $\gamma_{L^0}(C) \leq \gamma_{L^\ell}(C) \leq \gamma_{L^{\ell R}}(C)$. Especially, substituting $\alpha$ for $\ell$ gives

$$\gamma_{L^0}(C) \leq \gamma_{L^\alpha}(C) \leq \gamma_{L^{\alpha R}}(C).$$

(1)

For a vertex $v$ of $G$ with a proper labeling $L$, denote by $D^*_v(G, L)$ a minimum $L$-dominating set of $G$ such that $v$ has the most neighbors in this set among all minimum $L$-dominating sets of $G$, i.e., $|N_G(v) \cap D^*_v(G, L)| \geq |N_G(v) \cap D|$ for any minimum $L$-dominating set $D$ of $G$. Fig. 1 (b) illustrates an example of $D^*_v(G, L)$, while the minimum $L$-dominating set formed by the shaded vertices in Fig. 1 (a) cannot be selected as a $D^*_v(G, L)$. The construction and correctness of the algorithm is based on the following theorem.

Figure 1: Minimum $L$-dominating sets of a graph.
Theorem 1. Suppose $G$ is a graph with a proper labeling $L = (L_1, L_2)$. Suppose $C$ is an end-block of $G$ and $x$ is its unique cut-vertex. Let $G' = (G - C) \cup \{x\}$ and $L'$ be the restriction of $L$ on $G'$ with modifications as described below. Let $\alpha = \min\{L_2(x), \deg_G(x)\}$, $s = |N_G(x) \cap D_2^x(C, L^0)|$ and $t = \max\{L_2(x) - s - \deg_G(x), 0\}$.

1. If $\gamma_{L^0}(C) = \gamma_{L^0}(C)$, then $\gamma_L(G) = \gamma_L(G') + \gamma_{L^0}(C) - 1$, where $L'_1(x) = R, L'_2(x) = L_2(x) - \alpha$.

2. If $\gamma_{L^0}(C) < \gamma_{L^0}(C)$, then $\gamma_L(G) = \gamma_L(G') + \gamma_{L^0}(C)$, where $L'_1(x) = L_2(x) - \alpha$.

3. Suppose $\gamma_{L^0}(C) < \gamma_{L^0}(C)$.
   (3.1) If $L_1(x) = R$ or there exists $y \in N_G(x)$ such that $L_2(y) = \deg_G(y)$, then $\gamma_L(G) = \gamma_L(G') + \gamma_{L^0}(C) - 1$, where $L'_1(x) = R, L'_2(x) = L_2(x) - \ell$ and $\ell = s + t$.
   (3.2) If $L_2(y) < \deg_G(y)$ for all $y \in N_G(x)$, then $\gamma_L(G) = \gamma_L(G') + \gamma_{L^0}(C)$, where $L'_2(x) = L_2(x) - \ell$ and $\ell = s + t$.

The following observation is easily obtained.

Proposition 2. Suppose $G$ is a graph with a proper labeling $L$. Let $D$ be a $\gamma_L$-set of $G$. Then $|D \cap C| \geq \gamma_{L^0}(C)$.

Lemma 3. Suppose $G$ is a graph with a proper labeling $L$. Let $C$ be an end-block of $G$, $x$ be its unique cut-vertex and $G' = (G - C) \cup \{x\}$. Let $D$ be a $\gamma_L$-set of $G$. Let $\alpha = \min\{L_2(x), \deg_G(x)\}$. Let $D_C$ be a $\gamma_{L^0}$-set or a $\gamma_{L^0}$-set of $C$ and $\overline{D} = (D - C) \cup D_C$. Then $|N_G(x) \cap \overline{D}| \geq L_2(x) - \alpha$ and $|N_C(x) \cap \overline{D}| \geq \alpha$. In other words, $|N_G(x) \cap \overline{D}| \geq L_2(x)$.

Proof. By the definition of $L$-dominating set, $|N_G(x) \cap D| \geq L_2(x)$ and $|N_C(x) \cap \overline{D}| \geq \alpha$. If $L_2(x) \leq \deg_G(x)$, i.e., $\alpha = L_2(x)$, then $|N_G(x) \cap \overline{D}| \geq 0 = L_2(x) - \alpha$. If $L_2(x) > \deg_G(x)$, i.e., $\alpha = \deg_G(x)$, then $\alpha = \deg_G(x) \geq |N_C(x) \cap D|$, as $x$ can have at most $\deg_G(x)$ neighbors in $D$. Hence $|N_G(x) \cap \overline{D}| = |N_G(x) \cap D| = |N_G(x) \cap D| - |N_C(x) \cap D| \geq L_2(x) - \alpha$. The assertion holds clearly.

Proof of Theorem 1 (1). Let $D'$ be a $\gamma_{L'}$-set of $G'$ and $D_C$ be a $\gamma_{L^0}$-set of $C$. Since $L'_1(x) = L_1^{aR}(x) = R, x \in D' \cup D_C$. Also that $|N_G(x) \cap (D' \cup D_C)| = |N_G(x) \cap D' + |N_G(x) \cap D_C| \geq L'_2(x) + \alpha = L_2(x)$, by the assumption. We have $D' \cup D_C$ is an $L$-dominating set of $G$ and hence $\gamma_L(G) \leq |D' \cup D_C| = |D'| + |D_C| - 1 = \gamma_{L'}(G') + \gamma_{L^0}(C) - 1$. 

6
Conversely, suppose $D$ is a $\gamma_L$-set of $G$. Let $D_C$ be a $\gamma_{L^SNR}$-set of $C$ and let $\overline{D} = (D - C) \cup D_C$. By the assumption that $\gamma_{L^0}(C) = \gamma_{L^SNR}(C)$ and Eq. (1) and Proposition 2, $|D| \geq |\overline{D}|$. Note that no matter what $L_1(x)$ might be, $x$ is always be included in $\overline{D}$. By Lemma 3, $|N_G(x) \cap \overline{D}| \geq L_2(x)$ and thus $\overline{D}$ is also an $L$-dominating set of $G$. Moreover, $|N_G(x) \cap \overline{D}| \geq L_2(x) - \alpha = L'_2(x)$ and $|N_G(x) \cap \overline{D}| \geq \alpha$. Thus $\overline{D} \cap G'$ is an $L'$-dominating set of $G'$ and $\overline{D} \cap C$ is an $L^{SR}$-dominating set of $C$. Hence $\gamma_L(G) = |D| \geq |\overline{D}| = |\overline{D} \cap G'| + |\overline{D} \cap C| - 1 \geq \gamma_{L'^0}(G') + \gamma_{L^SNR}(C) - 1$. □

Note that under the assumptions of Theorem 1 (2), $x$ must be a bound vertex, i.e., $L_1(x) = B$. The argument is similar to that of Theorem 1 (1) and therefore is omitted.

**Proposition 4.** Suppose $G$ is a graph that is also a block. Suppose $L$ is a proper labeling on $G$. Let $x$ be a vertex of $G$. Let $s = |N_G(x) \cap D'_x(G, L^0)|$ and $j$ be a nonnegative integer such that $s + j \leq \deg_G(x)$. Then $\gamma_{L^*}(G) = \gamma_{L^*}(G) + j$.

**Proof.** Let $D_s$ be a $\gamma_{L^*}$-set of $G$ and $D = D_s \cup J$, where $J$ consists of $j$ vertices of $N_G(x) \setminus D_s$. Clearly $D$ is an $L^*+j$-dominating set of $G$ and $\gamma_{L^*}(G) \leq |D| = |D_s| + j = \gamma_{L^*}(G) + j$.

Conversely, suppose $D$ is a $\gamma_{L^*+j}$-set of $G$ and we have $|N_G(x) \cap D| \geq s + j$. Since $D$ must contain an $L^0$-dominating set of $G$, there exists a subset $J \subseteq V(G)$ such that $D \setminus J$ is an $L^0$-dominating set of $G$. Clearly, $|J| \leq \gamma_{L^*+j}(G) - \gamma_{L^0}(G)$. Also, by the definition of $s$, $\gamma_{L^0}(G) = \gamma_{L^*}(G)$ and $|N_G(x) \cap (D \setminus J)| \leq s$. This indicates that $|J| \geq j$. Therefore, $\gamma_{L^*+j}(G) \geq \gamma_{L^*}(G) + j = \gamma_{L^*}(G) + j$. □

**Proof of Theorem 1 (3).** We first consider (3.1) and assume that $L_1(x) = R$. It is easy to check that $L^R$ is a proper labeling on $V(C)$. Let $D'$ be a $\gamma_{L'}$-set of $G'$ and $D_C$ be a $\gamma_{L^SNR}$-set of $C$. Since $L'_1(x) = L'_1(x) = R$, $x \in D' \cup D_C$. Also by assumptions, we have $|N_G(x) \cap (D' \cup D_C)| = |N_G(x) \cap D'| + |N_G(x) \cap D_C| \geq L_2(x)$. Thus $D' \cup D_C$ is an $L$-dominating set of $G$ and hence $\gamma_L(G) \leq |D' \cup D_C| = |D'| + |D_C| - 1 = \gamma_{L^R}(G') + \gamma_{L^SNR}(C) - 1$.

Conversely, suppose $D$ is a $\gamma_L$-set of $G$. Let $D_C$ be a $\gamma_{L^SNR}$-set of $C$.

**Case 1:** $|N_G(x) \cap D| \leq \ell$. Let $\overline{D} = (D - C) \cup D_C$. We now claim that $|D \cap C| \geq \gamma_{L^SNR}(C)$. If this claim holds, then we have $|D| \geq |\overline{D}|$. First consider the case of $t = 0$. Since $\ell = s + t$, we have $\gamma_{L^SNR}(C) = \gamma_{L^SNR}(C)$. Since $x$ can have at most $s$ neighbors in a $\gamma_{L^SNR}$-set of $C$, $\gamma_{L^SNR}(C) = \gamma_{L^SNR}(C)$. Since
By Proposition 2 and the assumption that \(L_1(x) = R\), \(|D \cap C| \geq \gamma_{L_0}(C) = \gamma_{L_{1R}}(C)\) holds.

Now suppose \(t > 0\). From the definition of \(t\), we have \(L_2(x) = \ell + \deg_{G'}(x)\).
Since \(D\) contains at least \(L_2(x)\) vertices of \(N_G(x)\), \(D\) must contain at least \(\ell\) vertices of \(N_C(x)\), i.e., \(|N_C(x) \cap D| \geq \ell\), for otherwise we have \(|N_{G'}(x) \cap D| > L_2(x) - \ell = \deg_{G'}(x)\), a contradiction. By the assumption that \(L_1(x) = R\), \(D \cap C\) is also an \(L_{1R}\)-dominating set of \(C\). Hence \(|D \cap C| \geq \gamma_{L_{1R}}(C)\). We therefore have this claim.

By the choice of \(D\), \(x\) is included in \(\overline{D}\). Also that \(|N_C(x) \cap \overline{D}| \geq \ell\) and thus \(\overline{D} \cap C\) is an \(L_{1R}\)-dominating set of \(C\). By the assumption that \(|N_C(x) \cap D| \leq \ell\), we have \(|N_{G'}(x) \cap \overline{D}| = |N_G(x) \cap \overline{D}| - |N_C(x) \cap D| \geq L_2(x) - \ell = L_2'(x)\), and thus \(\overline{D} \cap G'\) is also an \(L'\)-dominating set of \(G'\). Hence \(\gamma_L(G) = |D| \geq |\overline{D}| = |\overline{D} \cap G'| + |\overline{D} \cap C| - 1 \geq \gamma_L(G') + \gamma_{L_{1R}}(C) - 1\).

Case 2: \(|N_C(x) \cap D| > \ell\). For easy writing, set \(r = |N_C(x) \cap D|\). Then \(D \cap C\) is an \(L_{1R}\)-dominating set of \(C\). Hence \(|D \cap C| \geq \gamma_{L_{1R}}(C)\). Let \(\overline{D} = (D - C) \cup D_C \cup Y\), where \(Y\) is a subset of \(N_{G'}(x)\) of cardinality \(r - \ell\). Note that \(Y\) may contain vertices that are already in \(D\). Clearly \(|D_C \cup Y| = \gamma_{L_{1R}}(C) + r - \ell\). We now show that \(|D \cap C| \geq |D_C \cup Y|\). Since \(r > \ell \geq s\), by Proposition 4, \(\gamma_{L_{1R}}(C) = \gamma_{L_{1R}}(C) + r - \ell\) and thus \(|D \cap C| \geq \gamma_{L_{1R}}(C) = \gamma_{L_{1R}}(C) + r - \ell = |D_C \cup Y|\) holds. Consequently, we have \(|D| \geq |D_C|\).

By the choice of \(\overline{D}\), \(x\) is included in \(\overline{D}\). Also that \(|N_C(x) \cap \overline{D}| \geq \ell\) and thus \(\overline{D} \cap C\) is an \(L_{1R}\)-dominating set of \(C\). Now we shall show that \(\overline{D} \cap G'\) is an \(L'\)-dominating set of \(G'\). From the definition of \(\ell\) and \(t\), if \(t = 0\) then \(L_2 - \ell = L_2 - s \leq \deg_{G'}(x)\); if \(t > 0\) then \(L_2 - \ell = \deg_{G'}(x)\). In both cases, we have \(L_2 - \ell \leq \deg_{G'}(x)\). Since \(D\) contains at least \(L_2(x)\) vertices of \(N_G(x)\), \(|N_{G'}(x) \cap D| \geq L_2(x) - r\). Let \(A\) be a vertex subset of \(N_{G'}(x) \cap D\) of cardinality \(L_2(x) - r\). Now we pick arbitrarily exactly \(r - \ell\) vertices from \(N_{G'}(x) \setminus A\) to form the vertex set \(Y\). Note that \(Y\) is well-defined as we have shown that \(L_2 - \ell \leq \deg_{G'}(x)\). Since \(N_{G'}(x) \cap \overline{D} = (N_{G'}(x) \cap D) \cup Y\), we have \(|N_{G'}(x) \cap \overline{D}| \geq |A| + |Y| = L_2(x) - \ell = L_2'(x)\). Consequently, \(\gamma_L(G) = |D| \geq |\overline{D}| = |\overline{D} \cap G'| + |\overline{D} \cap C| - 1 \geq \gamma_L(G') + \gamma_{L_{1R}}(C) - 1\).

Now we assume that \(L_1(x) = B\). If \(x\) has a neighbor \(y\) such that \(L_2(y) = \deg(y)\), then \(x\) must be included in any \(L\)-dominating set of \(G\). Let \(\overline{L}\) be the same as \(L\) except for the modification on \(\overline{L}_1(x) = R\). In this case, we have \(\gamma_L(G) = \gamma_{\overline{L}}(G)\). Thus the assertion holds clearly.

The argument of (3.2) is similar to (3.1) and therefore is omitted.
3.2. Algorithm

We are now in the position to present our algorithm, called kTTDom, to determine a minimum $L$-dominating set of a graph. In our algorithm, we assume kTTDomB is a subroutine that can find a minimum $L$-dominating set of each end-block $C$ of a graph.

Algorithm: kTTDom (A block-wise approach for finding a $\gamma_L$-set in graphs. kTTDomB is a subroutine we assume it can find a $\gamma_L$-set of each end-block of the graph)

Input: A graph $G$ with a proper labeling $L = (L_1, L_2)$.

Output: A minimum $L$-dominating set $D$ of $G$.

Method:

1. $G' \leftarrow G$;
2. $D \leftarrow \emptyset$;
3. while $G' \neq \emptyset$ do
   4. if $G'$ is a block then
      5. $D \leftarrow D \cup \text{kTTDomB}(G', L)$;
      6. $G' \leftarrow \emptyset$;
   7. else
      8. let $C$ be an end-block of $G'$ and $x$ be its unique cut-vertex;
      9. $\alpha \leftarrow \min \{L_2(x), \deg_C(x)\}$;
      10. $U_0 \leftarrow \text{kTTDomB}(C, L_0^0)$;
      11. $U_R \leftarrow \text{kTTDomB}(C, L_0^{\alpha_R})$;
      12. if $|U_0| = |U_R|$ then
         13. $D \leftarrow D \cup U_R$;
         14. $L_1(x) \leftarrow R$;
         15. $L_2(x) \leftarrow L_2(x) - \alpha$;
      16. else
         17. if $L_1(x) = B$ and $\exists y \in N_G(x)$ s.t. $L_2(y) = \deg_G(y)$ then
            18. $L_1(x) \leftarrow R$;
            19. $s \leftarrow |N_G(x) \cap D_2^*(C, L_0^0)|$;
            20. $\ell \leftarrow s + \max \{L_2(x) - s - \deg_G(x), 0\}$;
            21. $D \leftarrow D \cup \text{kTTDomB}(C, L^\ell)$;
            22. $L_2(x) \leftarrow L_2(x) - \ell$;
            23. $G' \leftarrow (G' - C) \cup \{x\}$;
      24. end
   25. end
end
Theorem 5. Algorithm kTTDom finds a minimum \( L \)-dominating set of a graph \( G \) in linear-time if \( kTTDomB \) and \( D^*_x(C, L^0) \) take linear time to compute for each end-block \( C \) of \( G \) with cut-vertex \( x \).

Proof. The correctness comes from Theorem 1. For the time complexity, since \( kTTDom \) calls at most three times of \( kTTDomB \) and computes \( D^*_x(C, L^0) \) at most once for each end-block \( C \) of the graph, it is clear that \( kTTDom \) is linear by the assumptions. \( \square \)

4. \( L \)-domination for some classes of graphs

In this section we present linear-time algorithms for finding a minimum \( L \)-dominating set for complete graphs, cycles and complete bipartite graphs. In addition, the computations of \( D^*_v(G, L) \) for the mentioned classes of graphs are also discussed. Combing with the algorithm presented in the previous section, we obtain a linear-time algorithm for the \( k \)-tuple total domination problem in graphs whose blocks are cliques, cycles or complete bipartite graphs. These include block graphs, cacti and block-cactus graphs.

Throughout the rest of this section, suppose \( G \) is a graph with a proper labeling \( L = (L_1, L_2) \). The aim is to find a minimum \( L \)-dominating set \( D \) of \( G \). Define

\[ \bar{R} = \{ v \in V : L_1(v) = R \text{ or } \exists u \in N_G(v) \text{ s.t. } L_2(u) = \deg_G(u) \}. \] (2)

And let \( |\bar{R}| = r \). By the definition of \( L \)-dominating set, all vertices of \( \bar{R} \) must be included in \( D \).

4.1. Complete graphs

Suppose \( G = (V, E) \) is a complete graph with \( n \) vertices. Let \( V \setminus \bar{R} = \{ v_1, v_2, \ldots, v_{n-r} \} \). If \( v_i \notin D \) and \( v_j \in D \) with \( L_2(v_i) < L_2(v_j) \) for some \( v_i, v_j \in V \setminus \bar{R} \), then \( (D - v_j) \cup \{ v_i \} \) is also a minimum \( L \)-dominating set of \( G \). This indicates that we shall choose vertices in \( V \setminus \bar{R} \) with \( L_2 \) value as small as possible. Suppose \( v^R_{\max} \) (resp. \( v^R_{\max} \)) has the maximum \( L_2 \) value among all vertices in \( \bar{R} \) (resp. \( V \setminus \bar{R} \)). It is the case that \( D = \bar{R} \cup S \), where \( S \) consists of the smallest \( s \) vertices of \( V \setminus \bar{R} \), where \( s = \max \{ L_2(v^R_{\max}) - r + 1, L_2(v^R_{\max}) - r, 0 \} \). However, some vertex in \( S \) may have \( L_2 \) value no less than \( r + s \). In that case, one can simply add another vertex of \( V \setminus \bar{R} \) to \( S \). In other words, choose the smallest \( s + 1 \) vertices of
$V \setminus \tilde{R}$ and $\tilde{R}$ to form a minimum $L$-dominating set of $G$. The process of the algorithm is described as follows.

**Algorithm:** kTTDomKn (Finding a $\gamma_L$-set of a complete graph)

**Input:** A complete graph $G = (V, E)$ with a proper labeling $L = (L_1, L_2)$.

**Output:** A minimum $L$-dominating set $D$ of $G$.

**Method:**

$$D \leftarrow \emptyset;$$
$$\tilde{R} \leftarrow \{ v \in V : L_1(v) = R \text{ or } \exists u \in N_G(v) \text{ s.t. } L_2(u) = \text{deg}_G(u) \};$$
$$r \leftarrow |\tilde{R}|;$$
$$v_0 \leftarrow \emptyset; // \text{ pseudo vertex}$$
$$L_2(v_0) \leftarrow 0;$$

let $v_1, v_2, \ldots, v_n$ be a vertex ordering of $V$ such that $v_i \in V \setminus \tilde{R}$ for $1 \leq i \leq n - r$, $L_2(v_1) \leq \cdots \leq L_2(v_{n-r})$ and $L_2(v_{n-r+1}) \leq \cdots \leq L_2(v_n);$  
$s \leftarrow \max \{ L_2(v_n) - r + 1, L_2(v_{n-r}) - r, 0 \};$

if $L_2(v_s) \geq r + s$ then 
$$s \leftarrow s + 1;$$
$$D \leftarrow \tilde{R} \cup \{ v_i \in V \setminus \tilde{R} : 0 \leq i \leq s \};$$

**Theorem 6.** Algorithm kTTDomKn finds a minimum $L$-dominating set for a complete graph in linear time.

**Proof.** The correctness is clear and is omitted. The time complexity is bound by the computation of the vertex ordering of $V \setminus \tilde{R}$. Note that $L$ is a proper labeling, $L_2(v) \leq \text{deg}_G(v) < n$ for all $v \in V \setminus \tilde{R}$. Since each $L_2(v)$ is an integer in the range 0 to $n$ and there are at most $n$ integers need to sort, one can use linear-time sorting algorithms, for examples, Counting sort, to obtain the vertex ordering.

Consider the computation of $D^*_x(G, L)$ of a complete graph $G$ for some fixed vertex $x$. Since each pair of vertices of $G$ is adjacent, any minimum $L$-dominating set $D$ of $G$ has the property that $|N_G(x) \cap D|$ is maximum, and can be selected as $D^*_x(G, L)$. Thus $D^*_x(G, L)$ can be found in linear time.

### 4.2. Trees

Since each end-block of a tree $T$ of order at least 2 is a clique of cardinality 2, a solution to find a minimum $L$-dominating set of a tree $T$ is to use kTTDom and kTTDomKn. However, since tree has a simple structure,
we provide an alternative algorithm to find a minimum $L$-dominating set in trees. The presented algorithm is actually the simplified version of the combination of $kTTDom$ and $kTTDomKn$ for the input graph to be trees, and will also be used later as a subroutine to find a minimum $L$-dominating set for cycles, cacti and block-cactus graphs.

Given a tree $T$ of $n$ vertices, it is well-known that $T$ has a vertex ordering $v_1, v_2, \ldots, v_n$ such that $v_i$ is a leaf of $G_i = G[v_i, v_{i+1}, \ldots, v_n]$ for $1 \leq i \leq n - 1$. This ordering can be found in linear-time by using, for example, the breadth-first-search (BFS) algorithm.

The algorithm visits vertices of $T$ along the tree ordering $v_1, v_2, \ldots, v_n$. In each iteration, processing a leaf $v_i$ of a tree $T$, which is adjacent to a unique vertex $u$. The label of $v_i$ is used to possibly relabel $u$. After $v$ is visited, $v_i$ is ignored from $T$ and a new tree $T'$ is obtained. A linear-time labeling algorithm for finding a $\gamma_L$-set in trees is shown as follows.

**Algorithm: kTTDomT** (Finding a $\gamma_L$-set of a tree)

**Input:** A tree $T$ of $n$ vertices with a tree ordering $v_1, v_2, \ldots, v_n$ and a proper labeling $L = (L_1, L_2)$.

**Output:** A minimum $L$-dominating set $D$ of $T$.

**Method:**

$D \leftarrow \emptyset$

for $i \leftarrow 1$ to $n$ do

$T' \leftarrow T[v_i, \ldots, v_n]$;

let $u$ be the parent of $v_i$ in $T'$ (regard $u$ as $v_i$ if $i = n$);

if $L_2(v_i) = 1$ then

$L_1(u) \leftarrow R$;

if $L_1(v_i) = R$ or $L_2(u) = \text{deg}_T(u)$ then // $v_i \in \tilde{R}$

$L_2(u) \leftarrow \max \{L_2(u) - 1, 0\}$;

$D \leftarrow D \cup \{v_i\}$;

end

The construction and correctness of the algorithm is based on the following lemma.

**Lemma 7.** Suppose $T$ is a tree of order at least 2 with a proper labeling $L = (L_1, L_2)$. Let $v$ be a leaf adjacent to $u$ in $T$.

(1) If $L_2(v) = 1$, then $\gamma_L(T) = \gamma_{L'}(T)$, where $L'$ is the same as $L$ except for the modification on $L'_1(u) = R$.  

12
(2) If $v \in \tilde{R}$, then $\gamma_L(T) = \gamma_{L'}(T - v) + 1$, where $L'$ is the restriction of $L$ on $T - v$ with the modification on $L'_2(u) = \max\{L_2(u) - 1, 0\}$

(3) If $L(v) = (B, 0)$ and $v \notin \tilde{R}$, then $\gamma_L(T) = \gamma_{L'}(T - v)$, where $L'$ is the restriction of $L$ on $T - v$.

Proof. (1) By the definition of $L$-domination, $\gamma_L(T) \leq \gamma_{L'}(T)$ holds clearly. Now suppose $D$ is a $\gamma_L$-set of $T$. Since $L_2(v) = 1$, $u$ must be included in $D$. Thus $D$ is also an $L'$-dominating set of $T$. Hence we have $\gamma_{L'}(T) \leq \gamma_L(T)$.

(2) By the Case (1), we can assume $L_1(u) = R$ if $L_2(v) = 1$ without loss of generality. Suppose $D'$ is a $\gamma_{L'}$-set of $T - v$. Set $D = D' \cup \{v\}$. Since $L'$ is the restriction of $L$ on $T - v$ with the modifications on $L'_2(u)$ and $L_2(u) \leq L'_2(u) + 1$, we have $|N_T(u) \cap D| \geq L_2(u)$. If $L_2(v) = 1$, then by assumption that $L_1(u) = R$ and thus $u \in D'$. Therefore $D$ is also an $L$-dominating set of $T$. Thus $\gamma_L(T) \leq |D| = |D'| + 1 = \gamma_{L'}(T - v) + 1$.

Conversely, suppose $D$ is a $\gamma_L$-set of $T$. By the assumption that $v \in \tilde{R}$, $v$ must be included in $D$. Set $D' = D \setminus \{v\}$. As $L'$ is the restriction of $L$ on $T - v$ with the modification on $L'_2(u)$, $|N_{T - v}(x) \cap D'| \geq L'_2(u) - 1 = L_2(u)$. Thus $D'$ is an $L'$-dominating set of $V(T')$. Hence $\gamma_{L'}(T - v) + 1 \leq |D'| + 1 = |D| = \gamma_L(T)$.

(3) Suppose $D'$ is a $\gamma_{L'}$-set of $T - v$. Since $L'$ is the restriction of $L$ on $T - v$ and $|N(v) \cap D'| \geq 0 = L_2(v)$, it is clear that $D'$ is an $L$-dominating set of $T$. Thus $\gamma_L(T) \leq |D'| = \gamma_{L'}(T - v)$.

Conversely, suppose $D$ is a $\gamma_L$-set of $T$. If $v \notin D$, then $D$ is also an $L'$-dominating set of $T - v$. Now suppose $v \in D$. If $|N_{T - v}(u) \cap (D - v)| \geq L'_2(u)$, then we are done. For otherwise, pick a vertex $w \in N_{T - v}(u) \setminus D$ and set $D' = (D - v) \cup \{w\}$. Since $v \notin \tilde{R}$, we have $L_2(u) < \deg_T(u)$ and thus $|N_{T - v}(u) \cap D'| \geq L'_2(u)$. Therefore $D'$ is also an $L'$-dominating set of $T - v$. Hence $\gamma_{L'}(T - v) \leq |D'| = |D| = \gamma_L(T)$.

\textbf{Theorem 8.} Algorithm $\texttt{kTTDomT}$ finds a minimum $L$-dominating set for a tree in linear-time.

4.3. Cycles

Suppose $G = (V, E)$ is a cycle. We will use $\texttt{kTTDomT}$ as a subroutine to find a minimum $L$-dominating set of $G$. Basically, the idea is to pick a particular vertex $v^*$ of $G$, cut the cycle at $v^*$ to get a path $P_{v^*}$, and then apply $\texttt{kTTDomT}$ to obtain a minimum $L'$-dominating set of $P_{v^*}$. The principle of picking is to choose a vertex that must be included in $D$ and is shown as
follows. If $R \neq \emptyset$, then we know that all vertices of $\tilde{R}$ must be included in $D$. Thus pick arbitrarily a vertex from $\tilde{R}$ as $v^*$. Now assume $\tilde{R} = \emptyset$, i.e., $L_1(v) = B$ and $L_2(v) < 2$ for all $v \in V$. If $L_2(v) = 0$ for all $v \in V$, then clearly $D = \emptyset$. Otherwise there must exist some vertex $v$ that has $L_2(v) = 1$. This indicates that at least one of the two neighbors of $v$, say $u$ and $w$, must be included in $D$. In this case, we can pick $v^*$ from $u$ or $w$ by testing the cardinality of the minimum $L'$-dominating set of $C$, where $L'$ is the same as $L$ except for the modification on $L_1(u) = R$ or $L_1(w) = R$.

After picking $v^*$, we need to modify the labels of its neighbors accordingly. Let $u^*$ and $w^*$ be the two neighbors of $v^*$ on $C$. Since $v^*$ must be included in $D$, the $L_2$ values of $u^*$ and $w^*$ should be decreased by 1. Let $P_{v^*}$ be the path of $C - v^*$ and $L^P$ be the restriction of $L$ on $P_{v^*}$ with the modifications as described below.

Case 1: $L_2(v^*) = 2$. Clearly $u^*$ and $w^*$ must be included in $D$. Thus just set the $L^P_2$ values of $u^*$ and $w^*$ as $R$.

Case 2: $L_2(v^*) = 1$. So at least one of $u^*$ and $w^*$ must be included in $D$. In this case, we test the cardinality of the minimum $L^Q$-dominating set of $P_{v^*}$ to decide which one of $u^*$ and $w^*$ should be chosen, where $L^Q$ is the same as $L^P$ except for the modifications on $L^Q_1(u^*) = R$ or $L^Q_1(w^*) = R$.

Case 3: $L_2(v^*) = 0$. Then none has to be changed in this case. The detailed algorithm is shown as in Algorithm $kTTDomCYC$.

The correctness is clear and therefore is omitted. The time complexity is clearly linear, since $kTTDomT$ is linear and $kTTDomCYC$ calls at most two times of $kTTDomT$.

**Theorem 9.** Algorithm $kTTDomCYC$ finds a minimum $L$-dominating set in a cycle in linear time.

Now consider the computation of $D^*_x(G, L)$ of a cycle $G$ for some fixed vertex $x$. Let $y$ and $z$ be the neighbors of $x$ in $G$. By the definition of $D^*_x(G, L)$, $|N_G(x) \cap D^*_x(G, L)| \leq 2$. If $|N_G(x) \cap D^*_x(G, L)| = 2$, then $y, z \in D^*_x(G, L)$ and $|D^*_x(G, L)| = \gamma_{L'}(G)$, where $L'$ is the same as $L$ with the modifications on $L_1(y) = R$ and $L_1(z) = R$. If $|N_G(x) \cap D^*_x(G, L)| = 1$ and suppose $y \in D^*_x(G, L)$, then $|D^*_x(G, L)| = \gamma_{L'}(G)$, where $L'$ is the same as $L$ with the modifications on $L_1(y) = R$. Thus one can find $D^*_x(G, L)$ by examining among all possible combinations of modifications on $L_1(v) = R$ for all neighbors $v$ of $x$ with the condition that $\gamma_{L'}(G) = \gamma_L(G)$. Since finding a $\gamma_L$-set of a cycle can be done in linear-time, the computation of $D^*_x(G, L)$ clearly can be done in linear time, too.
Algorithm: kTTDomCYC (Finding a $\gamma_L$-set of a cycle)
Input: A cycle $G = (V, E)$ with a proper labeling $L = (L_1, L_2)$.
Output: A minimum $L$-dominating set $D$ of $G$.
Method:
$D \leftarrow \emptyset$;
$\tilde{R} \leftarrow \{ v \in V : L_1(v) = R$ or $\exists u \in N_G(v)$ s.t. $L_2(u) = 2 \}$;
if $\tilde{R} \neq \emptyset$ then
   pick an arbitrarily vertex in $\tilde{R}$ as $v^*$;
else if $\exists v$ s.t. $L_2(v) = 1$ then
   let $u, w$ be the two neighbors of $v$;
   $U_u \leftarrow kTTDomCYC(G, L')$, where $L' \leftarrow L$ with $L'_1(u) \leftarrow R$;
   $U_w \leftarrow kTTDomCYC(G, L')$, where $L' \leftarrow L$ with $L'_1(w) \leftarrow R$;
   if $|U_u| \leq |U_w|$ then $D \leftarrow U_u$ else $D \leftarrow U_w$;
else stop;
let the two neighbors of $v^*$ be $u^*$ and $w^*$;
let $P_{v^*}$ be the path $G - v^*$ and $L^P$ be the restriction of $L$ on $P_{v^*}$;
$L^P_2(u^*) \leftarrow \max\{L_2(u^*) - 1, 0\}$; \label{eq:1}
$L^P_2(w^*) \leftarrow \max\{L_2(w^*) - 1, 0\}$; \label{eq:2}
if $L_2(v^*) = 2$ then
   $L^P_1(u^*) \leftarrow L^P_1(w^*) \leftarrow R$;
   $D \leftarrow kTTDomT(P_{v^*}, L^P) \cup \{v^*\}$;
else if $L_2(v^*) = 1$ then
   $U_{u^*} \leftarrow kTTDomT(P_{v^*}, L^Q)$, where $L^Q \leftarrow L^P$ with $L^Q_1(u^*) \leftarrow R$;
   $U_{w^*} \leftarrow kTTDomT(P_{v^*}, L^Q)$, where $L^Q \leftarrow L^P$ with $L^Q_1(w^*) \leftarrow R$;
   if $|U_{u^*}| \leq |U_{w^*}|$ then $D \leftarrow U_{u^*} \cup \{v^*\}$ else $D \leftarrow U_{w^*} \cup \{v^*\}$;
else
   $D \leftarrow kTTDomT(P_{v^*}, L^P) \cup \{v^*\}$;

4.4. Complete bipartite graphs
Suppose $G = (A \cup B, E)$ is a complete bipartite graph whose vertex set is a disjoint union of two independent sets $A$ and $B$. Let $r_1 = |A \cap \tilde{R}|$ and $r_2 = |B \cap \tilde{R}|$. The argument is similar to that of complete graphs. Let $a_{max}$ (resp. $b_{max}$) be a vertex in $A$ that has the maximum $L_2$ value among all vertices of $A$ (resp. $B$). Since $D$ must contain at least $L_2(a_{max})$ vertices of $B$, it is the case that we shall choose the smallest $\max\{L_2(a_{max}) - r_2, 0\}$ (resp. $\max\{L_2(b_{max}) - r_1, 0\}$) vertices of $B \setminus \tilde{R}$ (resp. $A \setminus \tilde{R}$). The process of the algorithm is described as follows.
Algorithm kTTDomKmn (Finding a $\gamma_L$-set of a complete bipartite graph)

**Input:** A complete bipartite graph $G$ whose vertex set is a disjoint union of two independent sets $A$ and $B$, and a proper labeling $L = (L_1, L_2)$.

**Output:** A minimum $L$-dominating set $D$ of $G$.

**Method:**

$D \leftarrow \emptyset$;

$R \leftarrow \{ v \in V : L_1(v) = R \text{ or } \exists u \in N_G(v) \text{ s.t. } L_2(u) = \deg_G(u) \}$;

$r_1 \leftarrow |A \cap R|$;

$r_2 \leftarrow |B \cap R|$;

let $a_1, a_2, \ldots, a_{|A|}$ and $b_1, b_2, \ldots, b_{|B|}$ be vertex orderings of $A$ and $B$, respectively, such that $a_i \in A \setminus R$ for $1 \leq i \leq |A| - r_1$, $b_i \in B \setminus R$ for $1 \leq i \leq |B| - r_2$, $L_2(a_1) \leq \cdots \leq L_2(a_{|A| - r_1})$, $L_2(a_{|A| - r_1 + 1}) \leq \cdots \leq L_2(a_{|A|})$, $L_2(b_1) \leq \cdots \leq L_2(b_{|B| - r_2})$, and $L_2(b_{|B| - r_2 + 1}) \leq \cdots \leq L_2(b_{|B|})$;

$a_0 \leftarrow \emptyset; \ b_0 \leftarrow \emptyset$; // pseudo vertices

$s_a \leftarrow \max\{L_2(a_{|A|}) - r_2, L_2(a_{|A| - r_1}) - r_2, 0\}$;

$s_b \leftarrow \max\{L_2(b_{|B|}) - r_1, L_2(b_{|B| - r_2}) - r_1, 0\}$;

$D \leftarrow R \cup \{a_i \in A \setminus R : 0 \leq i \leq s_a\} \cup \{b_i \in B \setminus R : 0 \leq i \leq s_b\}$;

**Theorem 10.** Algorithm kTTDomKmn finds a minimum $L$-dominating set in a complete bipartite graph in linear time.

**Proof.** The correctness is clear and is omitted. The time complexity is linear, since the vertex orderings of $A \setminus R$ and $B \setminus R$ can be found by using linear-time sorting algorithms.

The computation of $D^*_x(G, L)$ of a complete bipartite graph $G$ for some fixed vertex $x$ can be found in linear time as any minimum $L$-dominating set $D$ of $G$ has the property that $|N_G(x) \cap D|$ is maximum, and can be selected as $D^*_x(G, L)$.

It is well-known that block graphs, cacti, block-cactus graphs can be recognized in linear time. By Theorems 5, 6, 9, and 10, one can immediately have the following result.

**Theorem 11.** Algorithm kTTDom finds a minimum $L$-dominating set in linear time for graphs in which each block is a clique, a cycle or a complete bipartite graph, including block graphs, cacti and block-cactus graphs.
5. NP-completeness result

In this section, we study the complexity of the $k$-tuple total domination problem:

$k$-TUPLE TOTAL DOMINATION ($k$TTD)
INSTANCE: A graph $G = (V, E)$ and positive integers $k$ and $s$.
QUESTION: Does $G$ have a $k$-tuple total dominating set of size $\leq s$?

It has been proved that $k$TTD is NP-complete for bipartite graphs and split graphs [14], in which the reductions are mainly from the well-known vertex cover problem. Pradhan also showed that $k$TTD is NP-complete for doubly chordal graphs, a subclass of chordal graphs. In this section, we show that $k$TTD remains NP-complete for undirected path graphs, another subclass of chordal graphs. The argument is similar to that in [12], where the reduction is from another well-known NP-complete problem, the 3-dimensional matching problem. For the sake of completeness, we will describe the reduction completely.

3-DIMENSIONAL MATCHING (3DM)
INSTANCE: Disjoint sets $X$, $Y$ and $Z$, each of cardinality $q$, and a set $M \subseteq X \times Y \times Z$ of triples having cardinality $p$.
QUESTION: Is there a set of $q$ triples in $M$ such that each element of $X \cup Y \cup Z$ is contained in exactly one of these triples?

**Theorem 12.** For any fixed positive integer $k$, $k$TTD is NP-complete for undirected path graphs.

**Proof.** Obviously $k$TTD belongs to NP, since it is easy to verify a “yes” instance of $k$TTD in polynomial time. Consider an instance of 3DM. Let

$$X = \{ x_r : 1 \leq r \leq q \}, \quad Y = \{ y_s : 1 \leq s \leq q \}, \quad Z = \{ z_t : 1 \leq t \leq q \},$$

and a subset

$$M = \{ m_i = (x_r, y_s, z_t) : x_r \in X, y_s \in Y \text{ and } z_t \in Z \text{ for } 1 \leq i \leq p \}$$

of triples $X \times Y \times Z$. Now we construct a clique tree $T$ having $6p + 3q + 1$ cliques from which we will obtain an undirected path graph. The vertices of the tree $T$, which are represented by sets, are explained below.
For each triple $m_i \in M$, there are six vertices depend only upon the triple itself and not upon the elements within the triple:

\[
\{ A_{ij}, B_{ij}, C_{ij}, D_{ij} \} \\
\{ A_{ij}, B_{ij}, D_{ij}, E_{ij} \} \\
\{ C_{ij}, D_{ij}, F_{ij} \} \\
\{ A_{ij}, B_{ij}, D_{ij}, E_{ij} \} \\
\{ A_{ij}, E_{ij}, H_{ij} \} \\
\{ B_{ij}, E_{ij}, I_{ij} \} \text{ for } 1 \leq i \leq p, 1 \leq j \leq k.
\]

These six vertices form the subtree corresponding to $m_i$, which is illustrated in Fig. 2. Next there is a vertex for each element of $X, Y$ and $Z$ that depends upon the triples of $M$ to which each respective element belongs:

\[
\{ R_j \} \cup \{ A_{ij_1}, \ldots, A_{ij_k} : x_r \in m_i \} \text{ for all } x_r \in X, \\
\{ S_j \} \cup \{ B_{ij_1}, \ldots, B_{ij_k} : y_s \in m_i \} \text{ for all } y_s \in Y, \\
\{ T_j \} \cup \{ C_{ij_1}, \ldots, C_{ij_k} : z_t \in m_i \} \text{ for all } z_t \in Z.
\]

Finally, the last vertex, the root of the tree $T$, which contains

\[
\{ A_{ij}, B_{ij}, C_{ij} : 1 \leq i \leq p, 1 \leq j \leq k \}.
\]

The arrangement of these vertices in the tree $T$ is shown in Fig. 2. This then result in an undirected path graph $G$ with vertex set

\[
\{ A_{ij}, B_{ij}, C_{ij}, D_{ij}, E_{ij}, F_{ij}, G_{ij}, H_{ij}, I_{ij} : 1 \leq i \leq p, 1 \leq j \leq k \} \cup \\
\{ R_j, S_j, T_j : 1 \leq j \leq q \}
\]

of cardinality $9pk + 3q$, where the undirected path in $T$ corresponding to a vertex $v$ of $G$ consists of those vertices (sets) containing $v$ in the tree $T$.

Now we shall show that $3DM$ has a solution if and only if $G$ has a $k$-tuple total dominating set of cardinality $2kp + kq$. Suppose $3DM$ has a solution $M'$ of cardinality $q$. Let

\[
D = \{ A_{ij}, B_{ij}, C_{ij} : m_i \in M', 1 \leq j \leq k \} \cup \\
\{ D_{ij}, E_{ij} : m_i \in M \setminus M', 1 \leq j \leq k \}.
\]
It is straightforward to check that $D$ is a $k$-tuple total dominating set of $G$ of cardinality $3kq + 2k(p - q) = 2kp + kq$.

Conversely, suppose $D$ is a $k$-tuple total dominating set of $G$ of cardinality $2kp + kq$. Observe that for any $i$, the cardinality of a $k$-tuple total dominating set of the subgraph induced by the vertex set $\{A_{ij}, B_{ij}, C_{ij}, D_{ij}, E_{ij}, F_{ij}, G_{ij}, H_{ij}, I_{ij} : 1 \leq i \leq p, 1 \leq j \leq k\}$ corresponding to $m_i$ is at least $2k$, and the only way is to choose $D_{ij}$ and $E_{ij}$ for all $j, 1 \leq j \leq k$. Any larger $k$-tuple total dominating set might just as well consist of $A_{ij}, B_{ij}$ and $C_{ij}$ for all $j, 1 \leq j \leq k$, since none of the other possible vertices dominates any vertex outside of the subtree. Consequently, suppose $D$ contains $A_{ij}, B_{ij}, C_{ij}, 1 \leq j \leq k$ for $\ell m_i$'s, and $D_{ij}, E_{ij}, 1 \leq j \leq k$ for $p - \ell$ other $m_i$'s, and at least $\max\{3k(q - \ell), 0\}$ for $S_r, S_s, T_t$. Then we have

$$2kp + kq = \left| D \right| \geq 3k\ell + 2k(p - \ell) + 3k(q - \ell) = 2kp + 3kq - 2k\ell$$

and so $\ell \geq q$. Picking $q$ triples $m_i$ for which $A_{ij}, B_{ij}, C_{ij}, 1 \leq j \leq k$ are in $D$ form a matching $M'$ of cardinality $q$. \hfill \square

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{fig2.png}
\caption{A transformation to an undirected path graph.}
\end{figure}


