Algorithmic and complexity aspects of problems related to total restrained domination for graphs

Yu Yang[1] · Cai-Xia Wang[1] · Shou-Jun Xu[1]

Abstract

Let G be a graph with vertex set V and a subset $D \subseteq V$. D is a *total dominating set* of G if every vertex in V is adjacent to a vertex in D. D is a *restrained dominating set* of G if every vertex in $V \setminus D$ is adjacent to a vertex in D and another vertex in $V \setminus D$. D is a *total restrained dominating set* if D is both a total dominating set and a restrained dominating set. The minimum cardinality of total dominating sets (resp. restrained dominating sets, total restrained dominating sets) of G is called the *total domination number* (resp. *restrained domination number*, *total restrained domination number*) of G, denoted by $\gamma_t(G)$ (resp. $\gamma_r(G)$, $\gamma_{tr}(G)$). The MINIMUM TOTAL RESTRAINED DOMINATION (MTRD) problem for a graph G is to find a total restrained dominating set of minimum cardinality of G. The TOTAL RESTRAINED DOMINATION DECISION (TRDD) problem is the decision version of the MTRD problem. In this paper, firstly, we show that the TRDD problem is NP-complete for undirected path graphs, circle graphs, S-CB graphs and C-CB graphs, respectively, and that, for a S-CB graph or C-CB graph with n vertices, the MTRD problem cannot be approximated within a factor of $(1 - \epsilon)\ln n$ for any $\epsilon > 0$ unless $NP \subseteq DTIME(n^{O(\log\log n)})$. Secondly, for a graph G, we prove that the problem of deciding whether $\gamma_r(G) = \gamma_{tr}(G)$ is NP-hard even when G is restricted to planar graphs with maximum degree at most 4, and that the problem of deciding whether $\gamma_t(G) = \gamma_{tr}(G)$ is NP-hard even when G is restricted to planar bipartite graphs with maximum degree at most 5. Thirdly, we show that the MTRD problem is APX-complete for bipartite graphs with maximum degree at most 4. Finally, we design a linear-time algorithm for solving the MTRD problem for generalized series–parallel graphs.

Keywords Total domination number · Restrained domination number · Total restrained domination number · NP-complete · APX-complete · Linear algorithm

✉ Shou-Jun Xu
  shjxu@lzu.edu.cn

[1] School of Mathematics and Statistics, Gansu Center for Applied Mathematics, Lanzhou University, Lanzhou 730000, Gansu, China

1 Introduction

Domination is a well-researched topic in graph theory, the literature on this subject is well described in two books by Haynes et al. (1998a, b). And numerous researchers have thoroughly investigated the variations of domination in graphs.

Let G be a graph with vertex set V and edge set E. For vertex $v \in V$, the *open neighborhood* $N_G(v)$ of v is the set $\{u \in V | uv \in E\}$, its cardinality $|N_G(v)|$ is called the *degree* of v, and the *closed neighborhood* of v is the set $N_G[v] = N(v) \cup \{v\}$. Moreover, when no confusion can arise, $N_G(v)$ and $N_G[v]$ are simplified by $N(v)$ and $N[v]$, respectively. The maximum degree of G will be denoted by $\Delta(G)$. A *leaf* of G is a vertex of degree 1, a *support vertex* is a vertex adjacent to a leaf. For a subset $D \subseteq V$. D is a *total dominating set* (TDS), introduced by Cockayne et al. (1980), of G if every vertex in V is adjacent to a vertex in D. D is a *restrained dominating set* (RDS), introduced by Domke et al. (1999), of G if every vertex in $V \setminus D$ is adjacent to a vertex in D and another vertex in $V \setminus D$. D is a *total restrained dominating set* (TRDS), introduced by Ma et al. (2005), of G if every vertex in V is adjacent to a vertex in D and every vertex in $V \setminus D$ is adjacent to a vertex in $V \setminus D$. Obviously, a TRDS is exactly both a TDS and a RDS. By the definitions we easily get two observations:

Observation 1.1 Each RDS in graph G includes all its leaf vertices.

Observation 1.2 Each TRDS in graph G contains all its leaf vertices and support vertices.

The minimum cardinality of total dominating sets (resp. restrained dominating sets, total restrained dominating sets) of G is called the *total domination number* (resp. *restrained domination number*, *total restrained domination number*) of G, denoted by $\gamma_t(G)$ (resp. $\gamma_r(G)$, $\gamma_{tr}(G)$). The MINIMUM TOTAL RESTRAINED DOMINATION (MTRD) problem (resp. the MINIMUM TOTAL DOMINATION (MTD) problem) for a graph G is to find a TRDS (resp. a TDS) of minimum cardinality of G. Given a positive integer k and a graph G, the TOTAL RESTRAINED DOMINATION DECISION (TRDD) problem (resp. the TOTAL DOMINATION DECISION (TDD) problem) is to decide whether G has a TRDS (resp. a TDS) of cardinality at most k. For detail research on total domination, see a survey (Henning 2009).

Obviously, for a graph G with no isolated vertices, $\gamma_{tr}(G) \geq \gamma_t(G)$ and $\gamma_{tr}(G) \geq \gamma_r(G)$. Raczek and Cyman (2008) characterized the trees T with $\gamma_{tr}(T) = \gamma_t(T)$ and Raczek (2007) characterized the trees T with $\gamma_{tr}(T) = \gamma_r(T)$. Ma et al. (2005) proved that for bipartite graphs and chordal graphs, the TRDD problem is NP-complete. Chen et al. (2012) also proved that the TRDD problem is NP-complete for planar graphs and split graphs, respectively.

G is called an *undirected path graph* (*circle graph*) if there exists direct mapping between V and a family of paths of a tree (a set of chords of a circle) such that two vertices are adjacent if and only if the corresponding paths (chords) intersect. It is well known that undirected path graph is a subclass of chordal graphs.

A *comb* is a tree obtained from a path P (called *backbone*) by joining a pendant leaf (called a *tooth*) to each vertex of P. A *star* is defined as a tree with at most one non-leaf. A bipartite graph G with bipartition (X, Y) (roles of X and Y can be exchanged) is

called *tree-convex*, if there exists a tree T with vertex set X such that for every vertex $y \in Y$, the neighborhood of y induces a subtree of T. When T is a comb (resp. a star), then G is called *comb-convex* (resp. *star-convex*) bipartite graph. We abbreviate the two bipartite graphs above as C-CB graph and S-CB graph, respectively. Bao and Zhang (2012) demonstrated that the construction of the associated tree T can be completed in linear time, and tree-convex bipartite graphs can be efficiently identified in linear time.

As mentioned in Ma et al. (2005), an application of total restrained domination is that of guards and inmates. Each vertex in the TRDS represents a guard and each vertex not in the TRDS represents a inmate. For supervise inmates and ensure the safety of guards, each prisoner and each guard must be seen by some guards; However, to safeguard the rights of inmates, it is necessary to ensure that each innate is observed by another inmate; To ensure the lowest possible cost, it is preferable to deploy the fewest number of guards feasible.

The present paper is organized as follows. In Sect. 2, we prove that the TRDD problem is NP-complete for undirected path graphs, circle graphs, S-CB graphs and C-CB graphs, respectively, and that, for a S-CB graph or C-CB graph with n vertices, the MTRD problem cannot be approximated within a factor of $(1 - \epsilon) \ln n$ for any $\epsilon > 0$ unless $NP \subseteq DTIME(n^{O(\log \log n)})$. Next, for a graph G, we prove that the problem of deciding whether $\gamma_r(G) = \gamma_{tr}(G)$ is NP-hard even when G is restricted to planar graphs with maximum degree at most 4, and that the problem of deciding whether $\gamma_t(G) = \gamma_{tr}(G)$ is NP-hard even when G is restricted to planar bipartite graphs with maximum degree at most 5. Finally, we show that the MTRD problem is APX-complete for bipartite graphs with maximum degree at most 4. In Sect. 3, we design a linear-time algorithm for solving this problem for generalized series–parallel graphs.

2 Hardness results

In this section, we firstly show that the TRDD problem is NP-complete for undirected path graphs, circle graphs, S-CB graphs and C-CB graphs, respectively. Secondly, we discuss the lower bounds of approximation ratio for the MTRD problem in S-CB graphs and C-CB graphs, and the upper bound of approximation ratio for the MTRD problem in general graphs. Thirdly, we show that for a planar graph G with $\Delta(G) \leq 4$, it is NP-hard to decide whether $\gamma_r(G) = \gamma_{tr}(G)$, and that it is NP-hard to decide whether $\gamma_t(G) = \gamma_{tr}(G)$, even if G is a planar bipartite graph with $\Delta(G) \leq 5$. Finally, we show that the MTRD problem is APX-complete for bipartite graphs with maximum degree at most 4.

2.1 NP-completeness for the TRDD problem

To establish the NP-completeness of a problem P, it is enough to show that P belongs to the class NP and that a known NP-complete problem is reducible to P in polynomial
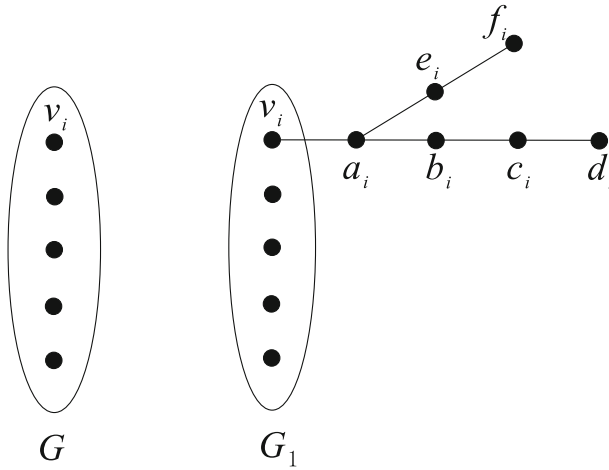
Fig. 1 Illustrations for Theorem 2.1: G and G_1 constructed from G

time. Here, We use the TDD problem, which is already known to be NP-complete for undirected path graphs (Laskar et al. 1984), circle graphs (Keil 1993), respectively.

Now we present a reduction from the TDD problem to the TRDD problem as follows. Let $G = (V, E)$ be a graph with $V = \{v_1, v_2, \ldots, v_n\}$, without isolated vertices, we construct a new graph $G_1 = (V_1, E_1)$, where $V_1 = V \cup \{a_i, b_i, c_i, d_i, e_i, f_i | 1 \leq i \leq n\}$ and $E_1 = E \cup \{v_i a_i, a_i b_i, b_i c_i, c_i d_i, a_i e_i, e_i f_i | 1 \leq i \leq n\}$ (see Fig. 1).

Theorem 2.1 *Let G and G_1 be graphs defined as above. Then G has a TDS of size at most k if and only if G_1 has a TRDS of size at most $k + 4n$.*

Proof **Sufficiency.** Let D be a TDS of G with $|D| \leq k$. It is obvious that $D_1 = D \cup \{c_i, d_i, f_i, e_i | 1 \leq i \leq n\}$ is a TRDS of G_1 with $|D_1| \leq k + 4n$.

Necessity. Let D' be a TRDS with $|D'| \leq k + 4n$ in G_1 such that $|D' \cap \{a_i, b_i | 1 \leq i \leq n\}|$ is as small as possible. For $1 \leq i \leq n$, since d_i, f_i are leaves of G_1, $\{c_i, d_i, f_i, e_i\} \subseteq D'$ by Observation 1.2. By the choice and definition of D', either $\{a_i, b_i\} \subseteq D'$ or $\{a_i, b_i\} \cap D' = \emptyset$. But we have

Claim. $D' \cap \{a_i, b_i\} = \emptyset$ for $1 \leq i \leq n$.

Let $I = \{i | a_i, b_i \in D'\}$ and $D'' = D' \cup \bigcup_{i \in I} \{v_i, v_{i'}\} \setminus \bigcup_{i \in I} \{a_i, b_i\}$, where $v_{i'} \in N_{G_1}(v_i) \cap V$. Now we show that D'' is a TRDS of G_1. For any $1 \leq i \leq n$, a_i is adjacent to $e_i \in D''$ and b_i is adjacent to $c_i \in D''$. Consider any $v_i \in V$. If $i \in I$, then there is $v_{i'} \in N_{G_1}(v_i) \cap V$ such that $v_{i'} \in D''$. Otherwise $i \notin I$, then there is $v_j \in N_{G_1}(v_i) \cap V \cap D'$. By the construction of D'', $v_j \in D''$. Any vertex $v_i \in V \setminus D''$ is adjacent to $a_i \notin D''$. Thus D'' is a TRDS of G_1 with $|D''| \leq |D'|$ and $D'' \cap \{a_i, b_i | 1 \leq i \leq n\} = \emptyset$, a contradiction with the choice of D'.

From Claim and $\{c_i, d_i, f_i, e_i\} \subseteq D'$ for $1 \leq i \leq n$, it follows immediately that $D' \cap V$ is a TDS with $|D' \cap V| \leq k$ in G. $\square$

From definitions of undirected path graphs and circle graphs, the following lemma emerges.

Lemma 2.1 *Assume that G_1 is constructed from a graph G as above, shown in Fig.* 1. *If G is an undirected path graph or circle graph, then so is G_1.*

Clearly, the TRDD problem is in NP for undirected path graphs and circle graphs, respectively. Combined with Theorem 2.1 and Lemma 2.1, the following theorem emerges, the following theorem emerges.

Theorem 2.2 *The* TRDD *problem is NP-complete for undirected path graphs and circle graphs, respectively.*

2.2 Lower bounds on the approximation ratio

To obtain a lower bound, we give an approximation preserving reduction from the MINIMUM SET COVER problem.

MINIMUM SET COVER (MSC) problem: Let U be any non-empty set and $\mathcal{S}$ be a collection of (non-empty) subsets of U such that $\cup_{S \in \mathcal{S}} S = U$ (and $(U, \mathcal{S})$ is called a *set system*). A subcollection $\mathcal{S}' \subseteq \mathcal{S}$ is called a *set cover* (SC) of U, if every element of U belongs to at least one element of $\mathcal{S}'$, i.e., $\cup_{S \in \mathcal{S}'} S = U$. The MSC problem is finding the smallest SC for a given set system $(U, \mathcal{S})$. The SET COVER DECISION problem is to decide whether U has a SC of cardinality at most k, for a given positive integer k and a set system $(U, \mathcal{S})$.

A lower bound of the approximation ratio for the MSC problem has been investigated in Feige (1998).

Theorem 2.3 (Feige 1998) *The MSC problem for the input instance $(U, \mathcal{S})$ does not admit a $(1 - \epsilon)\ln|U|$-approximation algorithm for any $\epsilon > 0$ unless $NP \subseteq DTIME(|U|^{O(\log\log|U|)})$. Furthermore, this inapproximability result holds for the case when the size of the input collection $\mathcal{S}$ is no more than the size of the base set U.*

Now, we show that a hardness result for the MTRD problem in S-CB graphs. First, the characterization of S-CB graphs was given as follows.

Lemma 2.2 (Pandey and Panda 2019) *Let $G = (X, Y, E)$ be a bipartite graph. G is a S-CB graph if, and only if, there is a vertex v in X, and every vertex in Y is either a pendant leaf or a neighbor of v.*

Theorem 2.4 *The* MTRD *problem for a S-CB graph G with n vertices does not admit a $(1 - \epsilon)\ln n$-approximation algorithm for any $\epsilon > 0$ unless $NP \subseteq DTIME(n^{O(\log\log n)})$.*

Proof It is sufficient to reduce the MSC problem to the MTRD problem. Let the set system $(U, \mathcal{S})$, where $U = \{u_1, u_2, \ldots, u_p\}$ and $\mathcal{S} = \{S_1, S_2, \ldots, S_m\}$, $m \leq p$ be an instance of the MSC problem. We construct the bipartite graph $G = (X, Y, E)$ as follows.

1. For $u_i \in U$, add a vertex x_i in X; add an additional vertex x_{p+1} in X; for $S_j \in \mathcal{S}$, add two vertices $y_{j,1}, y_{j,2}$ in Y.
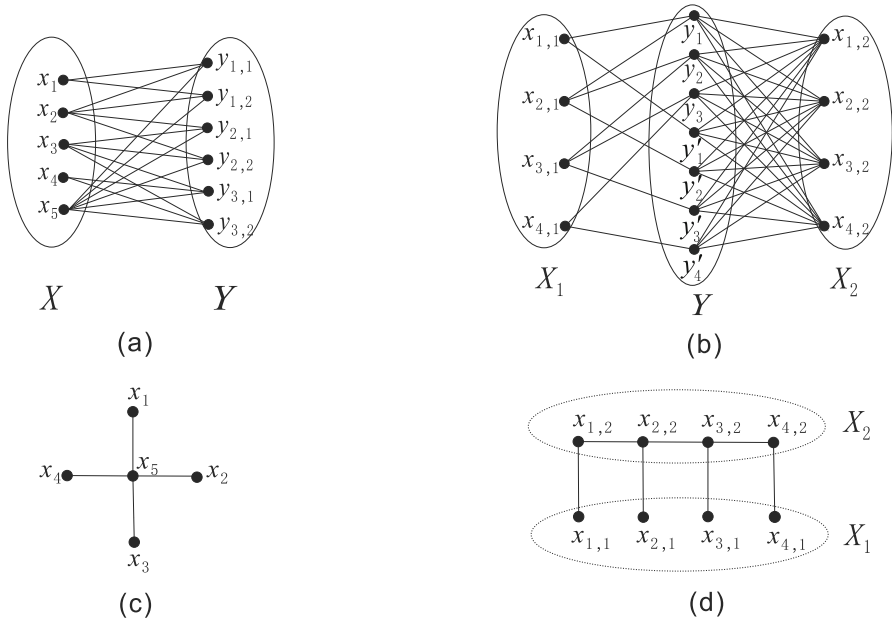
Fig. 2 **a** the S-CB graph and **b** the C-CB graph constructed from a set system $(U, \mathcal{S})$, where $U = \{u_1, u_2, u_3, u_4\}$, and $\mathcal{S} = \{\{u_1, u_2\}, \{u_2, u_3\}, \{u_3, u_4\}\}$; **(c)** the corresponding star and **(d)** the corresponding comb, respectively

2. For $u_i \in S_j$, add edges $x_i y_{j,1}, x_i y_{j,2}$ in E; add $\{x_{p+1} y_{j,1}, x_{p+1} y_{j,2} | 1 \leq j \leq m\}$ in E.

Clearly, in polynomial time, G can be constructed. By the construction of G, x_{p+1} is a neighbor of every vertex in Y and further G is a S-CB graph with $n = p + 2m + 1$ vertices by Lemma 2.2. See Fig. 2a for an example. Then we have

Construction ⋆: Let D be a TRDS of G defined as above. Construct a SC $\mathcal{S}'$ of U from D, where $\mathcal{S}' = \{S_j | y_{j,1} \text{ or } y_{j,2} \in D \cap Y, 1 \leq j \leq m\}$ with $|\mathcal{S}'| \leq |U| - 1$.

First, we show that $\mathcal{S}'$ is a SC, i.e., for any $1 \leq i \leq p$, there exists $S_j \in \mathcal{S}'$ such that $u_i \in S_j$. By the definition of D, there exists a neighbor y of x_i such that $y \in D$. Since G is bipartite, $y = y_{j,1}$ or $y_{j,2}$ for some $1 \leq j \leq m$, further $u_i \in S_j$ by the definition of G and $S_j \in \mathcal{S}'$ by the definition of $\mathcal{S}'$. Therefore, $\mathcal{S}'$ is a SC of U. Note that $|\mathcal{S}'| \leq |D \cap Y|$. Then we show that $|\mathcal{S}'| \leq |D| - 1$. Since D is a TRDS of bipartite graph $G = (X, Y, E)$, $D \cap X \neq \emptyset$. Thus $|\mathcal{S}'| \leq |D \cap Y| \leq |D \cap X| + |D \cap Y| - 1 = |D| - 1$.

Claim. For a positive integer k, U has a SC of size at most k if and only if G has a TRDS of size at most $k + 1$.

Let $\mathcal{S}'$ be a SC of U and $|\mathcal{S}'| \leq k$. Obviously, the set $D = \{x_{p+1}, y_{j,1} | S_j \in \mathcal{S}'\}$ is a TRDS of G and $|D| \leq k + 1$. Conversely, it can be obtained directly from Construction ⋆.

By Claim, if OPT_{sc} is a MSC of U and OPT_{trd} is a minimum TRDS of G, then $|OPT_{trd}| = |OPT_{sc}| + 1$. Suppose to the contrary that there exists a polynomial time algorithm, say Algorithm A, to approximate the MTRD problem within a ratio

of $(1 - \epsilon)\ln n$ for S-CB graphs with n vertices. Then Algorithm 1 (defined as follows) constructs a solution for the MSC problem.

Algorithm 1 APPROX-TOTAL-RESTRAINED-DOM-SET(G)

Input: An instance $(U, \mathcal{S})$.
Output: A SC $\mathcal{S}'$ of U.
    Construct a bipartite graph $G = (X, Y, E)$ from set system $(U, \mathcal{S})$ as above.
    Calculate a TRDS D for graph G by Algorithm A.
    Construct a SC $\mathcal{S}'$ of U by using Construction $\star$ from D.
    return $\mathcal{S}'$

Since each step of Algorithm 1 can be accomplished within polynomial time, Algorithm 1 can be computed in polynomial time. Let l be a positive integer such that $\frac{1}{l} < \epsilon$. If $|OPT_{sc}| \leq l$, then OPT_{sc} can be computed in polynomial time. Next we assume that $|OPT_{sc}| > l$. If $\mathcal{S}'$ is a SC of U computed by Algorithm 1, then $|\mathcal{S}'| < |D| \leq ((1 - \epsilon)\ln n)|OPT_{trd}| = ((1 - \epsilon)\ln n)(|OPT_{sc}| + 1) = ((1 - \epsilon)\ln n)(1 + \frac{1}{|OPT_{sc}|})|OPT_{sc}| < ((1 - \epsilon)\ln n)(1 + \frac{1}{l})|OPT_{sc}| < ((1 - \epsilon^2)\ln n)|OPT_{sc}|$. Since $m \leq p$, we have $n = p + 2m + 1 \leq 4p$. Then

$$|\mathcal{S}'| < ((1 - \epsilon^2)\ln n)|OPT_{sc}| \leq ((1 - \epsilon^2)\ln(4p))$$
$$|OPT_{sc}| = (1 - \epsilon^2)\left(1 + \frac{\ln 4}{\ln p}\right)(\ln p)|OPT_{sc}|.$$

For sufficiently large $p(= |U|)$, the term $1 + \frac{\ln 4}{\ln p}$ can be bounded by $1 + \epsilon^2$. Now we have

$$|\mathcal{S}'| < (1 - \epsilon^2)\left(1 + \frac{\ln 4}{\ln p}\right)(\ln p)|OPT_{sc}| \leq (1 - \epsilon')(\ln|U|)|OPT_{sc}|,$$

where $\epsilon' = \epsilon^4$.

By Theorem 2.3, if the MSC problem can be approximated within $(1 - \epsilon')\ln|U|$, then $NP \subseteq DTIME(|U|^{O(\log\log|U|)})$. It follows that if the MTRD problem can be approximated within $(1 - \epsilon)\ln n$ for any $\epsilon > 0$, then $NP \subseteq DTIME(n^{O(\log\log n)})$. Hence, the MTRD problem cannot be approximated within $(1 - \epsilon)\ln n$ unless $NP \subseteq DTIME(n^{O(\log\log n)})$. □

Theorem 2.5 (Karp 1972). *The* SET COVER DECISION *problem is NP-complete.*

Clearly, the TRDD problem is in NP for S-CB graphs. Combined Theorem 2.5 with Claim in Theorem 2.4, we get

Theorem 2.6 *The* TRDD *problem is NP-complete for S-CB graphs.*

Next, we show that the hardness result for the MTRD problem in C-CB graphs by giving the similar proof as in Theorem 2.4, but the construction of the bipartite graph is different.

Given a set system $(U, \mathcal{S})$, where $U = \{u_1, u_2, \ldots, u_p\}$ and $\mathcal{S} = \{S_1, S_2, \ldots, S_m\}$, $m \leq p$, we construct a C-CB graph $H = (X, Y, E)$ in the following way:

1. For $u_i \in U$, add two vertices $x_{i,1}$ and $x_{i,2}$ in X and a vertex y_i' in Y; for $S_j \in \mathcal{S}$, add a vertex y_j in Y; Denote by $X_k = \{x_{i,k} | 1 \leq i \leq p\}$, $k = 1, 2$, $Y' = \{y_i' | 1 \leq i \leq p\}$.
2. Add the edge $x_{i,1} y_i'$, $1 \leq i \leq p$;
3. If $u_i \in S_j$, then we add the edge $x_{i,1} y_j$, $1 \leq j \leq m$;
4. Form a complete bipartite graph between X_2 and Y. Note that H can be constructed in polynomial time. Figure 2b shows the construction of the C-CB graph H from the set system $(U, \mathcal{S})$, where $U = \{u_1, u_2, u_3, u_4\}$, and $\mathcal{S} = \{\{u_1, u_2\}, \{u_2, u_3\}, \{u_3, u_4\}\}$. It is easy to see that H is a C-CB graph if X_2 is taken as the backbone and X_1 is taken as the teeth of the comb. Next, we give a construction. *Construction* ⋆⋆: Let D be a TRDS of H defined as above, f a mapping from $D \cap Y'$ to $\mathcal{S}$ such that, for $y_i' \in D \cap Y'$ with $1 \leq i \leq p$, $u_i \in f(y_i')$. Construct a SC $|\mathcal{S}'| \leq |U| - 1$ of U from D by $\mathcal{S}' = \{S_j | y_j \in D, 1 \leq j \leq m\} \cup \{f(y_i') \in \mathcal{S} | y_i' \in D, 1 \leq i \leq p\}$.

    Firstly, we show that $\mathcal{S}'$ is a SC, i.e., for any $1 \leq i \leq p$, there exists $S_j \in \mathcal{S}'$ such that $u_i \in S_j$. By the definition of D, there exists a neighbor y of $x_{i,1}$ such that $y \in D$. Since H is bipartite, either $y = y_j$ for some $1 \leq j \leq m$, or $y = y_i'$. If $y = y_j$, then $u_i \in S_j$ by the definition of H and $(u_i \in) S_j \in \mathcal{S}'$ by the definition of $\mathcal{S}'$. Otherwise, $y = y_i'$. Then $u_i \in f(y_i')$ by the definition of f and $(u_i \in) f(y_i') \in \mathcal{S}'$ by the definition of $\mathcal{S}'$. Therefore, $\mathcal{S}'$ is a SC of U. Secondly, we show that $|\mathcal{S}'| \leq |D| - 1$. Since D is a TRDS of bipartite graph $H = (X, Y, E)$, $D \cap X \neq \emptyset$. Thus $|\mathcal{S}'| \leq |D \cap Y| \leq |D \cap X| + |D \cap Y| - 1 = |D| - 1$.

Lemma 2.3 *Let $(U, \mathcal{S})$ be a set system and H the bipartite graph associated with it, defined as above. Then $(U, \mathcal{S})$ has a SC of size at most k if and only if H has a TRDS of size at most $k+1$. In particular, the size of a MSC of U equals the size of a minimum TRDS of H minus 1.*

Proof Let $\mathcal{S}'$ be a SC of U with $|\mathcal{S}'| = k$. Evidently, the set $D = \{x_{1,2}\} \cup \{y_j | S_j \in \mathcal{S}'\}$ is a TRDS of H with $|D| = k + 1$. Conversely, it can be obtained directly from Construction ⋆⋆. □

Clearly, the TRDD problem is in NP for C-CB graphs. Combined Theorem 2.5 with Lemma 2.3, we get

Theorem 2.7 *The TRDD problem is NP-complete for C-CB graphs.*

Similar to Theorem 2.4 and its proof, we have

Theorem 2.8 *The MTRD problem for a C-CB graph G with n vertices does not admit a $(1 - \epsilon) \ln n$-approximation algorithm for any $\epsilon > 0$ unless $NP \subseteq DTIME(n^{O(\log \log n)})$.*

2.3 Upper bound on the approximation ratio

Let $G = (V, E)$ be a graph of maximum degree Δ with no isolated vertices, D be a minimum TRDS of G. Since for any $v \in D$ has a neighbor in D by the definition of TRDS, and $V = \cup_{v \in D} N_G(v)$, we have $|V| = |\cup_{v \in D} N_G(v)| \leq \sum_{v \in D} |N_G(v)| = \sum_{v \in D} d_G(v) \leq \sum_{v \in D} \Delta = \Delta \cdot |D|$. So $|V| \leq \Delta \cdot |D|$. It follows that (take the full vertex set as a TRDS).

Theorem 2.9 *For any graph $G = (V, E)$ with $\Delta(G) = \Delta$, the minimum TRDS can be approximated within an approximation ratio of Δ.*

2.4 Hardness results on $\gamma_r(G) = \gamma_{tr}(G)$ (resp. $\gamma_t(G) = \gamma_{tr}(G)$) problems for a graph *G*

It is evident that, for a graph G without isolated vertices, $\gamma_r(G) \leq \gamma_{tr}(G)$ and $\gamma_t(G) \leq \gamma_{tr}(G)$. A natural question arises: for which graphs does the respective equality hold? Raczek (2007) characterized the trees T with $\gamma_r(T) = \gamma_{tr}(T)$ and Raczek and Cyman (2008) characterized the trees T with $\gamma_t(T) = \gamma_{tr}(T)$, respectively. In this section, we show that it is NP-hard to decide whether $\gamma_r(G) = \gamma_{tr}(G)$ holds for planar graphs G with $\Delta(G) \leq 4$ and whether $\gamma_t(G) = \gamma_{tr}(G)$ holds for planar bipartite graphs G with $\Delta(G) \leq 5$, respectively. Here we use the following NP-complete problem, due to Dahlhaus et al. (1994), as the reduction.

PLANAR EXACTLY 3-BOUNDED 3-SAT *problem:* A formula Φ with variable set X and clause set C such that, firstly, each variable has exactly three literals with one of them occurring in two clauses and the other in one, and each clause in the disjunction of either two or three literals; secondly, the bipartite graph B with bipartition (X, C), obtained by connecting an edge between $x \in X$ and $c \in C$ if c contains one of the literals x and $\overline{x}$, is planar. Notice that $\Delta(B) \leq 3$. The PLANAR EXACTLY 3-BOUNDED 3-SAT is the problem of deciding whether a given Boolean formula satisfied the above conditions is satisfiable.

Theorem 2.10 *For a given planar graph G with $\Delta(G) \leq 4$, it is NP-hard to decide whether $\gamma_{tr}(G) = \gamma_r(G)$.*

Proof We reduce from the PLANAR EXACTLY 3-BOUNDED 3-SAT problem. Consider a set of clauses $C = \{c_1, c_2, \ldots, c_p\}$ with variables $X = \{x_1, x_2, \ldots, x_n\}$, as the input for the PLANAR EXACTLY 3-BOUNDED 3-SAT problem, say, B the planar bipartite graph associated with it, now we construct a graph $G = (V, E)$. For any variable x_i, we introduce a triangle G_{x_i} with two distinguished literal vertices x_i and $\overline{x_i}$, for any clause $c_j \in C$, we denoted by G_{c_j} the graph described in Fig. 3 (left). Finally, for $x_i \in c_j$ (resp. $\overline{x_i} \in c_j$), we add the edge $x_i a_j$ (resp. $\overline{x_i} a_j$). See Fig. 3 (right) for an example. In fact, G is obtained from planar B mainly by splitting each of vertices of degree 3 corresponding to variables into two vertices connecting an edge. Clearly, G is also planar and $\Delta(G) \leq 4$.

Claim. C is satisfiable if and only if $\gamma_{tr}(G) = \gamma_r(G)$.

First we show that $\gamma_r(G) = n + 5p$. Let D be a RDS of G. Then $|D \cap V(G_{x_i})| \geq 1$ for $1 \leq i \leq n$. For $1 \leq j \leq p$, $\{e_j, k_j\} \subseteq D$ by Observation 1.1, $|D \cap N[m_j]| \geq 1$, $|D \cap N[g_j]| \geq 1$ and $|D \cap N[b_j]| \geq 1$ by the definition of D. Note that $N[m_j]$, $N[g_j]$ and $N[b_j]$ are pairwise disjoint. Thus $|D \cap V(G_{c_j})| \geq 5$. Therefore, $\gamma_r(G) \geq n + 5p$. Moreover, $\{x_i, a_j, d_j, e_j, k_j, l_j | 1 \leq i \leq n, 1 \leq j \leq p\}$ is a RDS of G. Therefore, $\gamma_r(G) = n + 5p$.

Now, assuming that Φ is satisfiable. Given a satisfying truth assignment of Φ, next we will prove there exist a TRDS D of G with $|D| = n + 5p$. For $1 \leq i \leq n$, we add x_i to D, if x_i evaluate to true, or $\overline{x_i}$, otherwise. For $1 \leq j \leq p$, we add $\{a_j, d_j, e_j, k_j, l_j\}$
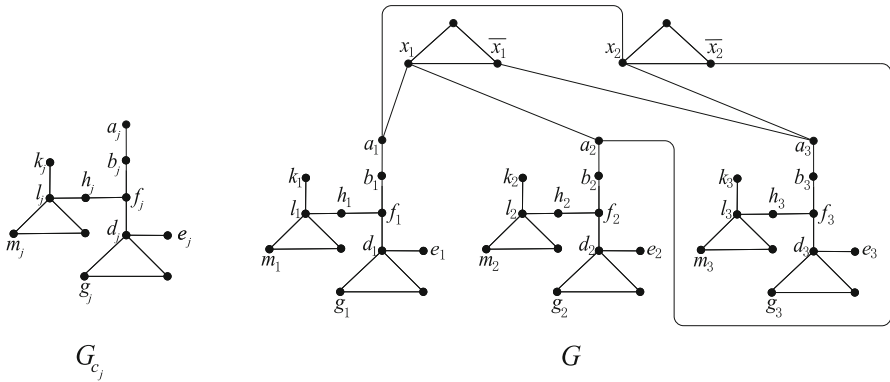
Fig. 3 G_{c_j} and G constructed from a formula $\Phi = (x_1 \vee x_2) \wedge (x_1 \vee \overline{x_2}) \wedge (\overline{x_1} \vee x_2)$

to D. Clearly, $|D| = n + 5p$. Next we show that D is a TRDS of G. By the construction of D, it is obvious that D is a dominating set and for any $v \in V \setminus D$, v has a neighbor in $V \setminus D$. It is sufficient to show that for any $v \in D$, v has a neighbor in D. Obviously, d_j is adjacent to e_j, l_j is adjacent to k_j for $1 \le j \le p$ and each literal vertex whose corresponding literal evaluates to true is adjacent to one of a_j for $1 \le j \le p$. Since Φ is satisfiable, each a_j has a neighbor that is a literal vertex whose corresponding literal is true, thus a_j is adjacent to a vertex in D.

Conversely, let's assume that $\gamma_{tr}(G) = \gamma_r(G)(= n + 5p)$ and let T be a TRDS of size $n + 5p$ in G. Similar to the proof above of $\gamma_r(G) = n + 5p$, we have $|G_{x_i} \cap T| = 1$ and $|T \cap G_{c_j}| = 5$. By Observation 1.2, $\{d_j, e_j, l_j, k_j\} \subseteq T$. If $b_j \in T$, then $|T \cap N[b_j]| \ge 2$, a contradiction with $|G_{c_j} \cap T| = 5$. If $b_j \notin T$, $f_j \in T$, then $h_j \in T$, otherwise h_j is an isolated vertex in $V \setminus T$. Thus $|G_{c_j} \cap T| \ge 6$, a contradiction with $|G_{c_j} \cap T| = 5$. So $a_j \in T$. Thus $T \cap V(G_{c_j}) = \{a_j, d_j, e_j, l_j, k_j\}$.

For $1 \le i \le n$, obviously, T contains exactly one of x_i and $\overline{x_i}$ in $V(G_{x_i})$. We establish a truth assignment for Φ in the following manner: set x_i to true if $x_i \in T$ and to false if $\overline{x_i} \in T$. Since each a_j belongs to T and $b_j \notin T$, by the definition of T, a_j has a neighbor that is a literal vertex in T, and the corresponding literal is true. Consequently, Φ satisfies all clauses. □

Theorem 2.11 *For a given planar bipartite graph G with $\Delta(G) \le 5$, it is NP-hard to decide whether $\gamma_{tr}(G) = \gamma_t(G)$.*

Proof We reduce from the PLANAR EXACTLY 3-BOUNDED 3-SAT problem. Consider a set of clauses $C = \{c_1, c_2, \ldots, c_p\}$ with variables $X = \{x_1, x_2, \ldots, x_n\}$, as the input for the PLANAR EXACTLY 3-BOUNDED 3-SAT problem, now we construct a graph $G = (V, E)$. For any variable x_i, we introduce a G_{x_i} with two special vertices x_i and $\overline{x_i}$, see Fig. 4 (left). For any clause $c_j \in C$, we introduce a G_{c_j} described in Fig. 4 (right). If literal x_i (resp. $\overline{x_i}$) in c_j, then we add the edge $x_i u_j^1$ (resp. $\overline{x_i} u_j^1$) in G. By simple analysis, we get G is planar bipartite graph and $\Delta(G) \le 5$.

Claim 1. $\gamma_t(G) = 4n + 5p$.

Let D be a TDS of G. By simple analysis, for each G_{c_j}, we have the subgraph induced by $\{v_j^1, v_j^2, \ldots, v_j^9\}$ (resp. by $\{u_j^1, u_j^2, \ldots, u_j^7\}$) contains at least three
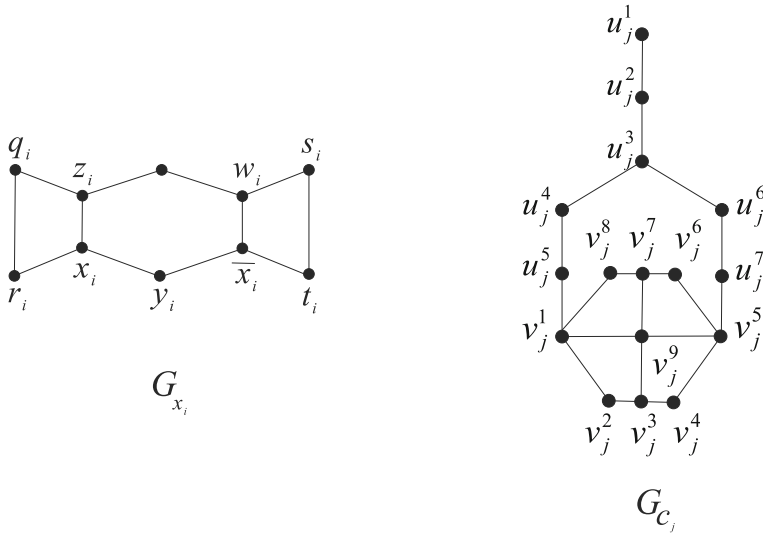
Fig. 4 Illustration for Theorem 2.11

(resp. two) vertices in D, thus $|D \cap V(G_{c_j})| \geq 5$; for each G_{x_i}, we have the subgraphs induced by $\{r_i, x_i, q_i, z_i\}$ and by $\{\overline{x_i}, t_i, s_i, w_i\}$ both contain at least two vertices in D, thus $|D \cap V(G_{x_i})| \geq 4$. Then, $\gamma_t(G) \geq 4n + 5p$. Moreover, $\{x_i, z_i, w_i, \overline{x_i}, u_j^2, u_j^3, v_j^1, v_j^5, v_j^9 | 1 \leq i \leq n, 1 \leq j \leq p\}$ of size $4n + 5p$ is a TDS of G. Therefore $\gamma_t(G) \leq 4n + 5p$, then $\gamma_t(G) = 4n + 5p$.

Next, we show

Claim 2. Φ is satisfiable if and only if $\gamma_{tr}(G) = \gamma_t(G)$.

Suppose that Φ is satisfiable. Given a satisfying truth assignment of Φ, next we construct a TRDS D of size $4n + 5p$ of G. In each G_{x_i}, we add $\{\overline{x_i}, w_i, q_i, r_i\}$ to D, if x_i evaluates to true, and add $\{x_i, z_i, s_i, t_i\}$ to D, otherwise. In each G_{c_j}, we add $\{u_j^2, u_j^3, v_j^1, v_j^5, v_j^9\}$ to D. Clearly, $|D| = 4n + 5p$. Now we show that D is a TRDS of G. Obviously, any $v \in V$ has a neighbor in D, every vertex in $V \setminus (D \cup \{u_j^1 | 1 \leq j \leq p\})$ has a neighbor in $V \setminus D$. It is sufficient to prove that u_j^1, $1 \leq j \leq p$, is adjacent to a vertex in $V \setminus D$. Since Φ is satisfiable, each u_j^1 has a neighbor that is a literal vertex whose corresponding literal is true, which is not in D by the definition of D, thus u_j^1 is adjacent to a vertex in $V \setminus D$.

Conversely, let's assume that $\gamma_{tr}(G) = \gamma_t(G)$ and let T be a TRDS of size $4n+5p$ in G. By the similar analysis as above, we have $|T \cap V(G_{x_i})| = 4$ and $|T \cap V(G_{c_j})| = 5$.

Fact 1. In each G_{x_i}, T contains exactly one of x_i and $\overline{x_i}$.

In G_{x_i}, since y_i is totally dominated by T, either $x_i \in T$ or $\overline{x_i} \in T$. If both of them are in T, then $y_i \in T$. Thus the remained one vertex in $T \cap V(G_{x_i})$ can not dominate q_i and s_i simultaneously, a contradiction.

Fact 2. In each G_{c_j}, $u_j^1 \notin T$, $u_j^2 \in T$.

Similar to the proof of Claim 1, $|T \cap \{u_j^1, u_j^2, \ldots, u_j^7\}| = 2$. Then $u_j^1 \notin T$ by the definition of T. If $u_j^2 \notin T$, then, by the definition of T, $T \cap \{u_j^1, u_j^2, \ldots, u_j^7\} = \{u_j^3, u_j^4\}$

or $\{u_j^3, u_j^6\}$. Assume that $T \cap \{u_j^1, u_j^2, \ldots, u_j^7\} = \{u_j^3, u_j^4\}$. To dominate u_j^7, then $v_j^5 \in T$. By simple analysis, we have $T \cap \{v_j^1, v_j^2, \ldots, v_j^9\} = \{v_j^1, v_j^5, v_j^9\}$. Then u_j^5 is isolated in $V \setminus T$, a contradiction with the definition of T. Therefore $u_j^2 \in T$.

We establish a truth assignment for Φ by Fact 1 in the following manner: set x_i to be true if $x_i \notin T$, to be false otherwise. By Fact 2 and the definition of T, u_j^1 is adjacent to a literal vertex which is not in T such that the corresponding literal is true. Consequently, Φ satisfies all clauses. □

2.5 APX-completeness

In this section we will prove the APX-completeness of the MTRD problem P for bipartite graphs with maximum degree at most 4. We need to prove that both P is in APX, that is, P can be approximated within a constant ratio, and P is APX-hard. To show the problem P is APX-hard, it is sufficient to show that there is an L-reduction from an APX-complete problem to P. We first recall the notation of L-reduction as follows.

L-reduction (Papadimitriou and Yannakakis 1991): Consider two NP-optimization problems F and G, along with a polynomial-time transformation f which maps instances from F and G. We can categorize f as an L-reduction if there are positive constants α and β, such that the following condition holds for every instance x of F,

1. $opt_G(f(x)) \leq \alpha \cdot opt_F(x)$,
2. For every feasible solution y of $f(x)$ with objective value c_2, in polynomial time one can find a solution y' of x with objective value c_1 such that $|opt_F(x) - c_1| \leq \beta \cdot |opt_G(f(x)) - c_2|$.
   Next we define the MINIMUM VERTEX COVER (MVC) problem. Given a graph $G = (V, E)$, a set $S \subseteq V$ of G is called a *vertex cover* (VC) of G if for every edge $uv \in E$, either $u \in S$ or $v \in S$. The MVC problem is to find a VC of G with minimum cardinality.

Theorem 2.12 *The* MTRD *problem is APX-complete for bipartite graphs with maximum degree at most 4.*

Proof By Theorem 2.9, the MTRD problem is in APX for bipartite graphs with maximum degree 4. The MVC problem for graphs with maximum degree at most 3 is APX-complete (Papadimitriou and Yannakakis 1991). Next, we use the MVC problem for graphs with maximum degree at most 3 to L-reduction the MTRD problem for bipartite graphs with maximum degree at most 4. Given a graph $G = (V, E)$ with $\Delta(G) \leq 3$, where $V = \{v_1, v_2, \ldots, v_n\}$ and $E = \{e_1, e_2, \ldots, e_m\}$, next we construct a graph $G^* = (V', E')$ (see Fig. 5 for an example):

1. For each edge $e_i = uv$ in G, we first subdivide the edge by introducing an additional vertex q_i (called *subdividing vertex* corresponding to e_i), connecting q_i and u, v, respectively; then we introduce three additional vertices r_i, s_i, t_i and add the edges $q_i r_i, r_i s_i, s_i t_i$.
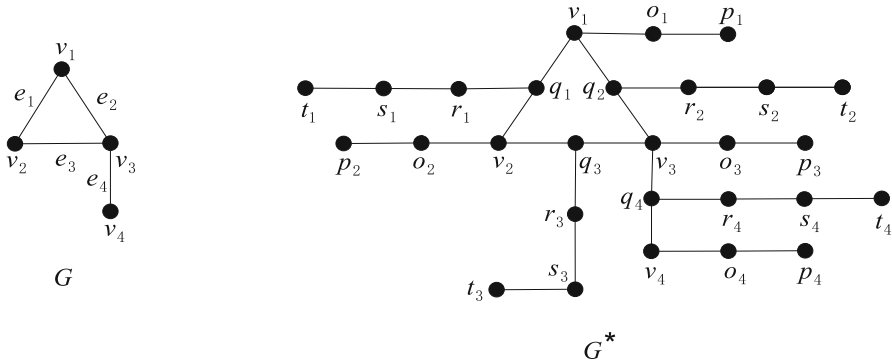
Fig. 5 Illustrations for the construction of G^* from G in the proof of Theorem 2.12
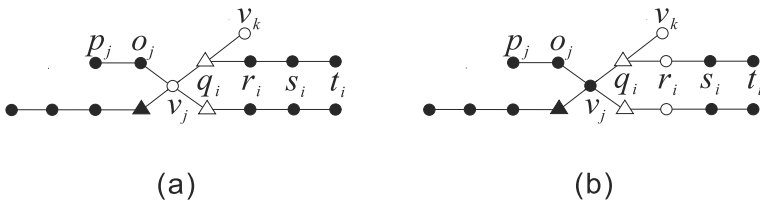

Fig. 6 Illustrations for the proof of Claim in Theorem 2.12. Vertices in Q are indicated as triangles, vertices in D (resp. D'') are indicated by solid vertices in (**a**) (resp. (**b**))

2. For each vertex v_i in G, take two additional vertices o_i, p_i and add the edges $v_i o_i$, $o_i p_i$. Obviously, G^* is bipartite and $\Delta(G^*) \leq 4$. For the sake of simplicity, we denote by $O = \{o_i | 1 \leq i \leq n\}$, $P = \{p_i | 1 \leq i \leq n\}$, $R = \{r_i | 1 \leq i \leq m\}$ and $Q = \{q_i | 1 \leq i \leq m\}$, $T = \{t_i | 1 \leq i \leq m\}$, $S = \{s_i | 1 \leq i \leq m\}$.

Claim. Let D be any TRDS of G^*. There exists a TRDS D' with $|D'| \leq |D|$ such that $(Q \cup R) \cap D' = \emptyset$. Furthermore, if C^* is a MVC of G and D^* is a minimum TRDS of G^*, then $|D^*| = |C^*| + 2n + 2m$.

First, $O \cup P \cup S \cup T \subseteq D$ by Observation 1.2. If $(Q \cup R) \cap D = \emptyset$, then we can choose $D' = D$, we have finished. Otherwise we can construct another TRDS D'' from D such that $|(Q \cup R) \cap D''| < |(Q \cup R) \cap D|$. Then we can continue to carry out the process repeatedly until we find D'.

Suppose there exists some $1 \leq i \leq m$ such that $\{q_i, r_i\} \cap D \neq \emptyset$ with $q_i \in Q$ corresponding to the edge $v_j v_k$ in G. If $q_i \in D$, then $r_i \in D$. Otherwise r_i is isolated in $V(G^*) \backslash D$, a contradiction with the definition of D. Next, we will break down the discussion into two cases.

Case 1. $\{q_i, r_i\} \subseteq D$.

If either v_j or v_k is in D, then, by the fact that $\{o_j, o_k\} \subseteq D$, $D'' = D \backslash \{q_i, r_i\}$ is as desired. Otherwise $D'' = D \backslash \{q_i\}$ is as desired.

Case 2. $q_i \notin D$ and $r_i \in D$.

If either v_j or v_k is in D, then $D'' = D \backslash \{r_i\}$ is as desired. Suppose $\{v_j, v_k\} \cap D = \emptyset$ in the following. Let $L = \{l | q_l \in N_{G^*}(v_j), q_l \notin D, r_l \in D\}$. Note that $i \in L$. Then $D'' = D \cup \{v_j\} \backslash \{r_l | l \in L\}$ is as desired (see Fig. 6 for the illustration).

If C^* is a MVC of G, then $C^* \cup O \cup P \cup S \cup T$ is a TRDS of G^*. Hence $|D^*| \le |C^*| + 2n + 2m$. Conversely, by the former part of Claim, we can assume that D^* is a minimum TRDS of G^* such that $D^* \cap (Q \cup R) = \emptyset$. Then for any vertex $q_i \in Q$, it is dominated by some v_j in D^*, which is an end of e_i in G by the definition of G^*. Thus $D^* \backslash (O \cup P \cup S \cup T)$ is a VC of G. Therefore, $|C^*| \le |D^*| - 2n - 2m$.

Finally, we prove that the previously introduced construction leads to an L-reduction. Let D^* be a minimum TRDS of G^* and C^* be a VC of G with the smallest cardinality. Since $\Delta(G) \le 3$, $m \le \frac{3}{2}n$ and $n \le 4|C|$ for any VC C. Hence $|D^*| = |C^*| + 2n + 2m \le 21|C^*|$. Moreover, given a TRDS D of G^*, there exists a TRDS D' in G^* with $|D'| \le |D|$ such that $D' \cap (Q \cup R) = \emptyset$ by Claim. Then $C = D' \backslash (O \cup P \cup S \cup T)$ is a VC of G by the same discuss above. Thus, $|C| - |C^*| = (|D'| - 2n - 2m) - |C^*| \le |D| - (2n + 2m + |C^*|) = |D| - |D^*|$. Therefore f is an L-reduction with $\alpha = 21, \beta = 1$.   □

3 A linear-time algorithm for generalized series–parallel graphs

In this section we study generalized series–parallel graphs, including outerplanar graphs and trees (Korneyenko 1994).

Generalized series–parallel (GSP) graphs with two distinct vertices u and v, called its *left* and *right terminals* respectively, and denoted by $G(u; v)$, are defined recursively as follows.

(1) A graph with only two vertices, denoted as u and v, connected by an edge represents a fundamental GSP graph with u and v as its terminals, and

(2) For two GSP graphs $G_1(u_1; v_1)$ and $G_2(u_2; v_2)$, performing any of the three operations below results in a GSP graph:

Series-1 (S_1) composition: sticking v_1 and u_2 while designating u_1 and v_2 as the left and right terminals, respectively, of the newly generated graph;

Series-2 (S_2) composition: sticking v_1 and u_2 (still named as v_1) while designating u_1 and v_1 as the left and right terminals, respectively, of the newly generated graph;

Parallel (P) composition: sticking u_1 and u_2 (denoted by u_1), v_1 and v_2 (denoted by v_1), respectively, while designating u_1 and v_1 as the left and right terminals, respectively, of the newly generated graph. We assume that this composition will not result in multiple edges.

For any GSP graph G, we introduce the definition of a parse tree $PT(G)$. A *parse tree $PT(G)$* is a rooted binary tree where every non-leaf vertex serves as a representation of a subgraph within G:

(1) Each leaf of $PT(G)$ exactly represents an edge in G;
(2) Each internal node u has a label $l(u)$ either S_1, S_2 or P, meaning that the subgraph represented by the node is constructed from two subgraphs represented by its two children by $l(u)$ composition;
(3) The root represents G itself.

The structure of any GSP graph G can be succinctly represented by its $PT(G)$. There are algorithms, with linear time complexity, to determine if a graph G is GSP and to construct a $PT(G)$ (Eppstein 1992; Ho et al. 1999). And note that parse trees of a
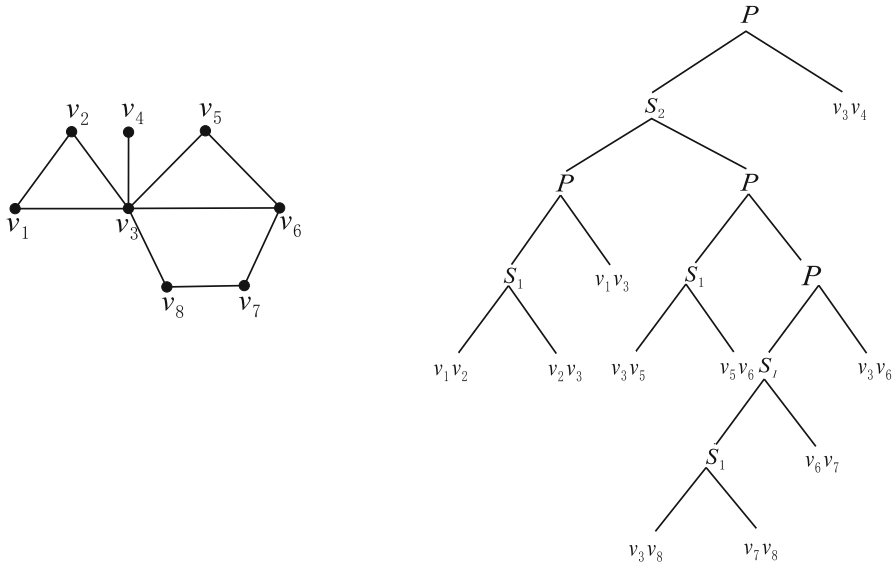
Fig. 7 A GSP graph (left), and one of its parse trees (right)

GSP graph are possibly not unique. An example for a GSP graph and one of its parse tree is shown in Fig. 7. In what follows, by using the structure of a parse tree, we will design an algorithm (linear-time) to solve the MTRD problem on GSP graphs.

First, for a GSP graph $G = G(u; v)$ and $i, j \in \{1, 2, \ldots, 5\}$, we define a subset, denoted by $T_{ij}(G)$, of vertex set $V(G)$ such that:

(1) $N(w) \cap T_{ij}(G) \neq \emptyset$ for every $w \in V(G) \backslash \{u, v\}$.

(2) $N(w) \cap (V(G) \setminus T_{ij}(G)) \neq \emptyset$ for every $w \in V(G) \setminus (\{u, v\} \cup T_{ij}(G))$.

(3) $u \in T_{1j}$ and $N(u) \cap T_{1j} \neq \emptyset$. (resp. $v \in T_{i1}$ and $N(v) \cap T_{i1} \neq \emptyset$).

(4) $u \in T_{2j}$ and $N(u) \cap T_{2j} = \emptyset$. (resp. $v \in T_{i2}$ and $N(v) \cap T_{i2} = \emptyset$).

(5) $u \notin T_{3j}$, $N(u) \cap T_{3j} \neq \emptyset$ and $N(u) \cap (V(G) \setminus T_{3j}) \neq \emptyset$. (resp. $v \notin T_{i3}$, $N(v) \cap T_{i3} \neq \emptyset$ and $N(v) \cap (V(G) \backslash T_{i3}) \neq \emptyset$).

(6) $u \notin T_{4j}$, $N(u) \cap T_{4j} \neq \emptyset$ and $N(u) \cap (V(G) \setminus T_{4j}) = \emptyset$. (resp. $v \notin T_{i4}$, $N(v) \cap T_{i4} \neq \emptyset$ and $N(v) \cap (V(G) \backslash T_{i4}) = \emptyset$).

(7) $u \notin T_{5j}$, $N(u) \cap T_{5j} = \emptyset$ and $N(u) \cap (V(G) \setminus T_{5j}) \neq \emptyset$. (resp. $v \notin T_{i5}$, $N(v) \cap T_{i5} = \emptyset$ and $N(v) \cap (V(G) \backslash T_{i3}) \neq \emptyset$). Let $MT_{ij}(G)$ be a minimum cardinality $T_{ij}(G)$ and $m_{ij}(G) = |MT_{ij}(G)|$. Note that $T_{ij}(G)$ may not exist for some i, j. If $T_{ij}(G)$ does not exist, denoted by Ω, where we let $S \cup \Omega = \Omega$ for any set S, then $m_{ij}(G)$ is infinity by convention. For example, $m_{11}(K_2) = 2$, $m_{24}(K_2) = 1$, $m_{42}(K_2) = 1$, $m_{55}(K_2) = 0$, $m_{ij}(K_2) = \infty$ otherwise.

Let $F_1, \ldots, F_s$ be subsets of $V(G)$. Define $\min\{F_t | 1 \leq t \leq s\}$ be a set which has minimum cardinality among all $\{F_t | 1 \leq t \leq s\}$, that is, $|\min\{F_t | 1 \leq t \leq s\}| = \min_{1 \leq t \leq s} |F_t|$. Obviously, $\min\{MT_{ij}(G) | i, j \in \{1, 3\}\}$ is a minimum TRDS of G.

Lemma 3.1 *Let $G(u_1; v_2)$ be a GSP graph obtained by applying a S_1 composition to GSP graphs $G_1(u_1; v_1)$ and $G_2(u_2; v_2)$. Then, for $i, j \in \{1, 2, 3, 4, 5\}$, the set*

$$
S_{ij} = \min \begin{cases} MT_{ip}(G_1) \cup MT_{qj}(G_2), & p, q \in \{1, 2\}, \ pq \neq 4; \\ MT_{ip}(G_1) \cup MT_{qj}(G_2), & p, q \in \{3, 4, 5\}, \ pq \neq 16, \ pq \neq 25. \end{cases}
$$

is a $MT_{ij}(G)$. Here v_1 in G_1 and u_2 in G_2 are considered as the same vertex.

Proof In the following, we just show the statement holds for the case of $i = 1$, $j = 1$, the other cases can be proven in similar ways. It is obvious that S_{11} defined in Lemma 3.1 is a $T_{11}(G)$. So $|S_{11}| \geq m_{11}(G)$. All that needs to be proven is that, for any chosen $MT_{11}(G)$, $|S_{11}| \leq |MT_{11}(G)|$. We divide $MT_{11}(G)$ into two cases to discuss.

Case 1. $v_1 \in MT_{11}(G)$.

Then by the definition of $MT_{11}(G)$, either $N_{G_1}(v_1) \cap MT_{11}(G) \neq \emptyset$ or $N_{G_2}(u_2) \cap MT_{11}(G) \neq \emptyset$, if, $N_{G_1}(v_1) \cap MT_{11}(G) \neq \emptyset$. Obviously, $G_1 \cap MT_{11}(G)$ is a $T_{11}(G_1)$, further $|MT_{11}(G_1)| \leq |G_1 \cap MT_{11}(G)|$. If $N_{G_2}(u_2) \cap MT_{11}(G) \neq \emptyset$, then $G_2 \cap MT_{11}(G)$ is a $T_{11}(G_2)$, further $|MT_{11}(G_2)| \leq |G_2 \cap MT_{11}(G)|$. Thus $|MT_{11}(G_1) \cup MT_{11}(G_2)| \leq |(G_1 \cap MT_{11}(G)) \cup (G_2 \cap MT_{11}(G))| = |MT_{11}(G)|$. Note that $MT_{11}(G_1) \cap MT_{11}(G_2) = (G_1 \cap MT_{11}(G)) \cap (G_2 \cap MT_{11}(G))$. If $N_{G_2}(u_2) \cap MT_{11}(G) = \emptyset$, then $G_2 \cap MT_{11}(G)$ is a $T_{21}(G_2)$ by the definition. Similarly, we have $|MT_{11}(G_1) \cup MT_{21}(G_2)| \leq |MT_{11}(G)|$. If $N_{G_1}(v_1) \cap MT_{11}(G) = \emptyset$, then $N_{G_2}(u_2) \cap MT_{11}(G) \neq \emptyset$, we can get $|MT_{12}(G_1) \cup MT_{11}(G_2)| \leq |MT_{11}(G)|$.

Case 2. $v_1 \notin MT_{11}(G)$.

Subcase 2.1. $N_{G_1}(v_1) \cap MT_{11}(G) \neq \emptyset$ and $N_{G_1}(v_1) \setminus MT_{11}(G) \neq \emptyset$.

Obviously, $G_1 \cap MT_{11}(G)$ is a $T_{13}(G_1)$. If $N_{G_2}(u_2) \cap MT_{11}(G) = \emptyset$, then $G_2 \cap MT_{11}(G)$ is a $T_{51}(G_2)$. Thus, $|MT_{13}(G_1) \cup MT_{51}(G_2)| \leq |MT_{11}(G)|$. Note that $T_{13}(G_1) \cap T_{51}(G_2) = (G_1 \cap MT_{11}(G)) \cap (G_2 \cap MT_{11}(G)) = \emptyset$. If $N_{G_2}(u_2) \subseteq MT_{11}(G)$, then $G_2 \cap MT_{11}(G)$ is a $T_{41}(G_2)$. Thus, $|MT_{13}(G_1) \cup MT_{41}(G_2)| \leq |MT_{13}(G)|$. Otherwise, $G_2 \cap MT_{11}(G)$ is a $T_{31}(G_2)$. Thus, $|MT_{13}(G_1) \cup MT_{31}(G_2)| \leq |MT_{13}(G)|$.

Subcase 2.2. $N_{G_1}(v_1) \subseteq MT_{11}(G)$.

Obviously, $G_1 \cap MT_{11}(G)$ is a $T_{14}(G_1)$ and $N_{G_2}(u_2) \setminus MT_{11}(G) \neq \emptyset$. If $N_{G_2}(u_2) \cap MT_{11}(G) = \emptyset$, then $G_2 \cap MT_{11}(G)$ is a $T_{51}(G_2)$. Thus $|MT_{14}(G_1) \cup MT_{51}(G_2)| \leq |MT_{11}(G)|$. Otherwise, $G_2 \cap MT_{11}(G)$ is a $T_{31}(G_2)$. Thus $|MT_{14}(G_1) \cup MT_{31}(G_2)| \leq |MT_{11}(G)|$.

Subcase 2.3. $N_{G_1}(v_1) \cap MT_{11}(G) = \emptyset$.

Obviously, $G_1 \cap MT_{11}(G)$ is a $T_{15}(G_1)$ and $N_{G_2}(u_2) \cap MT_{11}(G) \neq \emptyset$. If $N_{G_2}(u_2) \subseteq MT_{11}(G)$, then $G_2 \cap MT_{11}(G)$ is a $T_{41}(G_2)$. Thus $|MT_{15}(G_1) \cup MT_{41}(G_2)| \leq |MT_{11}(G)|$. Otherwise, $G_2 \cap MT_{11}(G)$ is a $T_{31}(G_2)$. Thus $|MT_{15}(G_1) \cup MT_{31}(G_2)| \leq |MT_{11}(G)|$. □

Lemma 3.2 *Let $G = G(u_1; v_1)$ be a GSP graph obtained by applying a S_2 composition to GSP graphs $G_1(u_1; v_1)$ and $G_2(u_2; v_2)$. Then, for any $i \in \{1, 2, 3, 4, 5\}$, $t \in \{1, 3\}$, $q \in \{2, 4, 5\}$, the sets*

$$
S_{i1} = \min \begin{cases} MT_{is}(G_1) \cup MT_{1t}(G_2), & s \in \{1, 2\}; \\ MT_{i1}(G_1) \cup MT_{2t}(G_2), \end{cases}
$$

$$S_{iq} = \min\{MT_{iq}(G_1) \cup MT_{qt}(G_2)\};$$

$$S_{i3} = \min \begin{cases} MT_{i3}(G_1) \cup MT_{st}(G_2), & s \in \{3, 4, 5\}; \\ MT_{ip}(G_1) \cup MT_{3t}(G_2), & p \in \{3, 4, 5\}; \\ MT_{i4}(G_1) \cup MT_{5t}(G_2), \\ MT_{i5}(G_1) \cup MT_{4t}(G_2). \end{cases}$$

are $MT_{i1}(G)$, $MT_{iq}(G)$, $MT_{i3}(G)$, respectively. Here v_1 in G_1 and u_2 in G_2 are considered as the same vertex.

Proof In the following, we just show the statement holds for the case of $i = 1$, $j = 1$, and $i = 1$, $j = 3$, the other cases can be proven in similar ways. It is obvious that S_{11} and S_{13} defined in Lemma 3.2 is a $T_{11}(G)$, $T_{13}(G)$, respectively. So $|S_{11}| \geq m_{11}(G)$, $|S_{13}| \geq m_{13}(G)$. All that needs to be proven is that, for any chosen $MT_{11}(G)$, $|S_{11}| \leq |MT_{11}(G)|$, for any chosen $MT_{13}(G)$, $|S_{13}| \leq |MT_{13}(G)|$. First we show that $|S_{11}| \leq |MT_{11}(G)|$. By the definition of $MT_{11}(G)$, $v_1 \in MT_{11}(G)$, and either $N_{G_1}(v_1) \cap MT_{11}(G) \neq \emptyset$ or $N_{G_2}(u_2) \cap MT_{11}(G) \neq \emptyset$. We divide $MT_{11}(G)$ into two cases to discuss.

Case 1. $v_2 \in MT_{11}(G)$.

Subcase 1.1. $N_{G_1}(v_1) \cap MT_{11}(G) \neq \emptyset$.

If $N_{G_2}(u_2) \cap MT_{11}(G) \neq \emptyset$, obviously, $G_1 \cap MT_{11}(G)$ is a $T_{11}(G_1)$, $G_2 \cap MT_{11}(G)$ is a $T_{11}(G_2)$. Thus $|MT_{11}(G_1) \cup MT_{11}(G_2)| \leq |MT_{11}(G)|$. Otherwise, $G_1 \cap MT_{11}(G)$ is a $T_{11}(G_1)$, $G_2 \cap MT_{11}(G)$ is a $T_{21}(G_2)$. Thus $|MT_{11}(G_1) \cup MT_{21}(G_2)| \leq |MT_{11}(G)|$.

Subcase 1.2. $N_{G_1}(v_1) \cap MT_{11}(G) = \emptyset$.

By the definition of $MT_{11}(G)$, $N_{G_2}(u_2) \cap MT_{11}(G) \neq \emptyset$, then $G_1 \cap MT_{11}(G)$ is a $T_{12}(G_1)$, $G_2 \cap MT_{11}(G)$ is a $T_{11}(G_2)$. Thus $|MT_{12}(G_1) \cup MT_{11}(G_2)| \leq |MT_{11}(G)|$.

Case 2. $v_2 \notin MT_{11}(G)$.

The proof is similar to Case 1, thus we have $|MT_{11}(G_1) \cup MT_{13}(G_2)| \leq |MT_{11}(G)|$, $|MT_{11}(G_1) \cup MT_{23}(G_2)| \leq |MT_{11}(G)|$, $|MT_{12}(G_1) \cup MT_{13}(G_2)| \leq |MT_{11}(G)|$.

Next we show that $|S_{13}| \leq |MT_{13}(G)|$. By the definition of $MT_{13}(G)$, $v_1 \notin MT_{13}(G)$, and either $N_{G_1}(v_1) \cap MT_{13}(G) \neq \emptyset$ or $N_{G_2}(u_2) \cap MT_{13}(G) \neq \emptyset$. Next, we divide $MT_{13}(G)$ into two cases to discuss.

Case 1. $v_2 \in MT_{13}(G)$.

Subcase 1.1. $N_{G_1}(v_1) \cap MT_{13}(G) \neq \emptyset$.

Subsubcase 1.1.1. $N_{G_2}(u_2) \cap MT_{13}(G) \neq \emptyset$.

If $N_{G_1}(v_1) \setminus MT_{13}(G) \neq \emptyset$ and $N_{G_2}(u_2) \setminus MT_{13}(G) \neq \emptyset$. Obviously, $G_1 \cap MT_{13}(G)$ is a $T_{13}(G_1)$, $G_2 \cap MT_{13}(G)$ is a $T_{31}(G_2)$. Thus $|MT_{13}(G_1) \cup MT_{31}(G_2)| \leq |MT_{13}(G)|$. If $N_{G_1}(v_1) \setminus MT_{13}(G) \neq \emptyset$ and $N_{G_2}(u_2) \setminus MT_{13}(G) = \emptyset$. Obviously, $G_1 \cap MT_{13}(G)$ is a $T_{13}(G_1)$, $G_2 \cap MT_{13}(G)$ is a $T_{41}(G_2)$. Thus $|MT_{13}(G_1) \cup MT_{41}(G_2)| \leq |MT_{13}(G)|$. If $N_{G_1}(v_1) \setminus MT_{13}(G) = \emptyset$ and $N_{G_2}(u_2) \setminus MT_{13}(G) \neq \emptyset$. Obviously, $G_1 \cap MT_{13}(G)$ is a $T_{14}(G_1)$, $G_2 \cap MT_{13}(G)$ is a $T_{31}(G_2)$. Thus $|MT_{14}(G_1) \cup MT_{31}(G_2)| \leq |MT_{13}(G)|$.

Subsubcase 1.1.2. $N_{G_2}(u_2) \cap MT_{13}(G) = \emptyset$.

If $N_{G_1}(v_1) \setminus MT_{13}(G) \neq \emptyset$, Obviously, $G_1 \cap MT_{13}(G)$ is a $T_{13}(G_1)$, $G_2 \cap MT_{13}(G)$ is a $T_{51}(G_2)$. Thus $|MT_{13}(G_1) \cup MT_{51}(G_2)| \leq |MT_{13}(G)|$. Otherwise, $G_1 \cap MT_{13}(G)$

is a $T_{14}(G_1)$, $G_2 \cap MT_{13}(G)$ is a $T_{51}(G_2)$. Thus $|MT_{14}(G_1) \cup MT_{51}(G_2)| \leq |MT_{13}(G)|$.

Subcase 1.2. $N_{G_1}(v_1) \cap MT_{13}(G) = \emptyset$.

By the definition of $MT_{13}(G)$, $N_{G_2}(u_2) \cap MT_{13}(G) \neq \emptyset$. If $N_{G_2}(u_2) \backslash MT_{13}(G) \neq \emptyset$. Obviously, $G_1 \cap MT_{13}(G)$ is a $T_{15}(G_1)$, $G_2 \cap MT_{13}(G)$ is a $T_{31}(G_2)$. Thus $|MT_{15}(G_1) \cup MT_{31}(G_2)| \leq |MT_{13}(G)|$. If $N_{G_2}(u_2) \backslash MT_{13}(G) = \emptyset$. Obviously, $G_1 \cap MT_{13}(G)$ is a $T_{15}(G_1)$, $G_2 \cap MT_{13}(G)$ is a $T_{41}(G_2)$. Thus $|MT_{15}(G_1) \cup MT_{41}(G_2)| \leq |MT_{13}(G)|$.

Case 2. $v_2 \notin MT_{13}(G)$.

The proof is similar to Case 1, thus we have $|MT_{13}(G_1) \cup MT_{33}(G_2)| \leq |MT_{13}(G)|$, $|MT_{13}(G_1) \cup MT_{43}(G_2)| \leq |MT_{13}(G)|$, $|MT_{14}(G_1) \cup MT_{33}(G_2)| \leq |MT_{13}(G)|$, $|MT_{13}(G_1) \cup MT_{53}(G_2)| \leq |MT_{13}(G)|$, $|MT_{14}(G_1) \cup MT_{53}(G_2)| \leq |MT_{13}(G)|$, $|MT_{15}(G_1) \cup MT_{33}(G_2)| \leq |MT_{13}(G)|$, $|MT_{15}(G_1) \cup MT_{43}(G_2)| \leq |MT_{13}(G)|$. □

Lemma 3.3 *Let $G(u_1; v_2)$ be a GSP graph obtained by applying a P composition to GSP graphs $G_1(u_1; v_1)$ and $G_2(u_2; v_2)$. Then, for any i, $j \in \{2, 4, 5\}$, the sets*

$S_{11} = \min\{MT_{sx}(G_1) \cup MT_{ty}(G_2) | s, t, x, y \in \{1, 2\}, st \neq 4, xy \neq 4\}$;

$S_{1j} = \min\{MT_{sj}(G_1) \cup MT_{tj}(G_2) | s, t \in \{1, 2\}, st \neq 4\}$;

$S_{13} = \min\{MT_{sp}(G_1) \cup MT_{tq}(G_2) | s, t \in \{1, 2\}, p, q \in \{3, 4, 5\}, st \neq 4, pq \neq 16, 25\}$;

$S_{i1} = \min\{MT_{is}(G_1) \cup MT_{it}(G_2) | s, t \in \{1, 2\}, st \neq 4\}$;

$S_{ij} = MT_{ij}(G_1) \cup MT_{ij}(G_2)$;

$S_{i3} = \min\{MT_{ip}(G_1) \cup MT_{iq}(G_2) | p, q \in \{3, 4, 5\}, pq \neq 16, 25\}$;

$S_{31} = \min\{MT_{ps}(G_1) \cup MT_{qt}(G_2) | s, t \in \{1, 2\}, p, q \in \{3, 4, 5\}, st \neq 4, pq \neq 16, 25\}$;

$S_{3j} = \min\{MT_{pj}(G_1) \cup MT_{qj}(G_2) | p, q \in \{3, 4, 5\}, pq \neq 16, 25\}$;

$S_{33} = \min\{MT_{pm}(G_1) \cup MT_{qn}(G_2) | p, q, m, n \in \{3, 4, 5\}, pq, mn \neq 16, 25\}$.

are $MT_{11}(G)$, $MT_{1j}(G)$, $MT_{13}(G)$, $MT_{i1}(G)$, $MT_{ij}(G), MT_{i3}(G)$, $MT_{31}(G)$, $MT_{3j}(G)$, $MT_{33}(G)$, respectively. Here u_1 in G_1 and u_2 in G_2 are considered as the same vertex, v_1 in G_1 and v_2 in G_2 are considered as the same vertex.

Based on Lemmas 3.1-3.3, an algorithm for finding a minimum TRDS of GSP graphs is designed by using dynamic programming technique.

Theorem 3.1 *Algorithm 2 produces a minimum TRDS of a GSP graph in linear-time.*

Proof The correctness of Algorithm 2 comes directly from Lemmas 3.1–3.3. Next, we analyze the time complexity. $PT(G)$ can be constructed in linear-time in Eppstein (1992); Ho et al. (1999); Hopcroft and Tarjan (1973). And since $PT(G)$ has $O(|E|)$ leaves, the initialization of lines 1–3 of Algorithm 2 takes $O(|E|)$ time. The iteration in lines 4–15 of Algorithm 2 is $O(|E|)$, because $PT(G)$ is a binary tree with $O(|E|)$ nodes and the calculation of each iteration in lines 4–15 requires a constant time. Therefore, the time complexity of Algorithm 2 is linear-time. □

Algorithm 2 TOTAL-RESTRAINED-DOM-GSP(G)

Require: A GSP graph G.
Ensure: A minimum TRDS D of G.
   Construct a parse tree $PT(G)$ of G with root r, each non-leaf node is labelled by S_1, S_2 or P, mark all leaves;
1: **for** each leaf $K_2(u; v)$ of $PT(G)$ **do**
2:    $MT_{11}(K_2) = \{u, v\}$; $MT_{24}(K_2) = \{u\}$; $MT_{42}(K_2) = \{v\}$; $MT_{55}(K_2) = \emptyset$; Others are Ω .
3: **end for**
4: **while** there are still unmarked nodes in $PT(G)$ **do**
5:    Select an unmarked node w in $PT(G)$ with marked children; Define G_w as the corresponding subgraph of node w;
6:    **if** $l(w) = S_1$ **then**
7:       Compute all $MT_{ij}(G_w)$ for $1 \leq i, j \leq 5$ by Lemma 3.1;
8:    **end if**
9:    **if** $l(w) = S_2$ **then**
10:       Compute all $MT_{ij}(G_w)$ for $1 \leq i, j \leq 5$ by Lemma 3.2;
11:    **end if**
12:    **if** $l(w) = P$ **then**
13:       Compute all $MT_{ij}(G_w)$ for $1 \leq i, j \leq 5$ by Lemma 3.3;
14:    **end if**
      Mark w;
15: **end while**
16: **return** $D = \min\{MT_{ij}(G_r)|i, j \in \{1, 3\}\}$

Data availibility Enquiries about data availability should be directed to the authors.

Declarations

Conflict of interest The authors have not disclosed any competing interests.

References

Bao FS, Zhang Y (2012) A review of tree convex sets test. Comput Intell 28(3):358–372

Chen L, Zeng W, Lu C (2012) NP-completeness and APX-completeness of restrained domination in graphs. Theor Comput Sci 448(24):1–8

Cockayne EJ, Dawes RM, Hedetniemi ST (1980) Total domination in graphs. Networks 10(3):211–219

Dahlhaus E, Johnson DS, Papadimitriou CH, Seymour PD, Yannakakis M (1994) The complexity of multiterminal cuts. SIAM J Comput 23(4):864–894

Domke GS, Hattingh JH, Hedetniemi ST, Laskar RC, Markus LR (1999) Restrained domination in graphs. Discret Math 203(1–3):61–69

Eppstein D (1992) Parallel recognition of series-parallel graphs. Inform Comput 98(1):41–55

Feige U (1998) A threshold of $\ln n$ for approximating set cover. JACM 45(4):634–652

Haynes TW, Hedetniemi ST, Slater PJ (1998) Fundamentals of domination in graphs. Marcel Dekker Inc, New York

Haynes TW, Hedetniemi ST, Slater PJ (1998) Domination in Graphs. Marcel Dekker, New York

Henning MA (2009) A survey of selected recent results on total domination in graphs. Discret Math 309(1):32–63

Hopcroft JE, Tarjan RE (1973) Dividing a graph into triconnected components. SIAM J Comput 2(3):135–158

Ho CW, Hsieh SY, Chen GH (1999) Parallel decomposition of generalized series-parallel graphs. J Inf Sci Eng 15(3):407–417

Karp RM (1972) Reducibility among combinatorial problems in complexity of computer computations. Springer, Boston, pp 85–103

Keil JM (1993) The complexity of domination problems in circle graphs. Discret Appl Math 42(1):51–63

Korneyenko NM (1994) Combinatorial algorithms on a class of graphs. Discret Appl Math 54(2–3):215–217

Laskar R, Pfaff J, Hedetniemi SM, Hedetniemi ST (1984) On the algorithmic complexity of total domination. SIAM J Algeb Discret Methods 5(3):420–425

Ma DX, Chen XG, Sun L (2005) On total restrained domination in graphs. Czechoslov Math J 55(1):165–173

Pandey A, Panda BS (2019) Domination in some subclasses of bipartite graphs. Discret Appl Math 252(10):51–66

Papadimitriou CH, Yannakakis M (1991) Optimization, approximation, and complexity classes. J Comput Syst Sci 43:425–440

Raczek J, Cyman J (2008) Total restrained domination numbers of trees. Discret Math 308(1):44–50

Raczek J (2007) Trees with equal restrained domination and total restrained domination numbers. Discuss Math Graph Theory 27:83–91