Mathematics

https://www.aimspress.com/journal/Math

AIMS Mathematics, 11(5): 14953–14983.
DOI: 10.3934/math.2026615
Received: 07 January 2026
Revised: 13 May 2026
Accepted: 15 May 2026
Published: 28 May 2026

Research article

Global total domination number: exact and approximate results

Ernesto Parra Inza[1], **Juan C. Hernández Gómez**[1], **José María Sigarreta Almira**[1,*] **and Nodari Vakhania**[2]

[1] Facultad de Matemáticas, Universidad Autónoma de Guerrero, Acapulco de Juárez, Guerrero, México

[2] Centro de Investigación en Ciencias, Universidad Autónoma del Estado de Morelos, Cuernavaca, Morelos, México

* **Correspondence:** Email: 14366@uagro.mx.

Abstract: A nonempty set $S \subseteq V$ is a *global total dominating set* (GTDS) in a graph $G = (V, E)$ if every vertex in both G and its complement $\overline{G}$ is adjacent to at least one vertex in S. The problem of finding a GTDS with the minimum cardinality $\gamma_t^g(G)$ is NP-hard, whereas no exact or approximation algorithm is known for the problem. We derive several new bounds and exact results for $\gamma_t^g(G)$, particularly for graphs of diameter 2 and triangle-free graphs. We propose an integer linear programming (ILP) model, as a natural adaptation of a previous model for the global dominating set problem. We build three heuristic algorithms that we evaluate against optimal solutions for benchmark instances with up to 10,700 vertices, the only benchmark instances which were solved by the ILP solver CPLEX using our ILP model. At least one of our heuristics generated an optimal solution for 41% of the instances for which CPLEX proved optimality, while for the remaining instances in this subset, the average absolute deviation from the optimum was only 1.58 vertices. In total, we report the behavior of the heuristics for over 1,100 instances with up to 25,000 vertices. In particular, Heuristics H2 and H3 run in time $O(n^3)$, and all proposed methods obtained solutions within 5 minutes in our computational experiments.

Keywords: graph; global total dominating set; lower and upper bounds; integer linear program; heuristic algorithm
Mathematics Subject Classification: 68R10, 05C69, 05C85

1. Introduction

Throughout this paper, we consider a graph G with the vertex set $V(G)$ and the edge set $E(G)$, of *order* $|V(G)| = n$ and *size* $|E(G)| = m$. The *subgraph induced* by the set $A \subseteq V(G)$, denoted $G[A]$, is the graph G without the vertices in $V(G) \setminus A$ and the edges with an endpoint in $V(G) \setminus A$. The *complement* of the graph G, denoted $\overline{G}$, is a graph with the vertex set $V(G)$ and an edge set consisting of all edges not in $E(G)$. Given a set $A \subseteq V(G)$ and a vertex $v \notin A$, a vertex $u \in A$ is a *neighbor* of vertex v in set A if and only if (iff) $(u, v) \in E(G)$. We denote the set of all neighbors of the vertex v in Set A as $N_A(v)$ and we let $N_A[v] = N_A(v) \cup \{v\}$. The *degree* of the vertex v in Set A is $\delta_A(v) = |N_A(v)|$. A vertex in Set A that is not a neighbor of the vertex v will be referred to as a *non-neighbor* of v. We denote the set of all non-neighbors of the vertex v in Set A as $\overline{N}_A(v)$, and we let $\overline{N}_A[v] = \overline{N}_A(v) \cup \{v\}$ and $\overline{\delta}_A(v) = |\overline{N}_A(v)|$. We may omit the argument $V(G)$ in $N_{V(G)}(v)$, $N_{V(G)}[v]$, $\delta_{V(G)}(v)$, $\overline{N}_{V(G)}(v)$, and $\overline{N}_{V(G)}[v]$. We denote the *minimum* and *maximum degrees* of the vertices in $V(G)$ by $\delta(G)$ and $\Delta(G)$, respectively. If X and Y are two subsets of Set $V(G)$, let $E(X, Y)$ be the set of all edges of graph G that join a vertex in X with a vertex in Y. The *length* of a $u - v$ path (a path connecting vertices u and v) is the number of edges on that path. The *distance* $d(u, v)$ between the vertices u and v is the length of a *shortest* $u - v$ path (one with the minimum number of edges). The *diameter* of G, denoted $\mathrm{diam}(G)$, is the maximum number of edges on the shortest path between any pair of vertices in G. The *girth* of a graph, denoted $g(G)$, is the length of the shortest cycle in the graph G.

A nonempty set $S \subseteq V(G)$ is called a *total dominating set* in a graph $G = (V, E)$ if every vertex in $V(G)$ is adjacent to at least one vertex in S. The *total domination number* of graph G, $\gamma_t(G)$, is the number of vertices in its smallest total dominating set. A total dominating set of the cardinality $\gamma_t(G)$ is referred to as a $\gamma_t(G)$-set. Notice that a graph G may not admit a global total dominating set (GTDS), and that whenever a total dominating set is concerned, the graph G cannot have an *isolated* vertex, i.e., a vertex with degree 0, and that $\gamma_t(G) \geq 2$. For a previous work on the total dominating set problem, the reader is referred, for example, to [1–3] and the references within them.

A subset S of vertices of a graph G is a GTDS if S is a total dominating set in both G and its complement $\overline{G}$. The *global total domination number* of the graph G, $\gamma_t^g(G)$, is the number of vertices in its smallest GTDS. We aim to find an *optimal solution*, a GTDS with cardinality $\gamma_t^g(G)$, that is, a $\gamma_t^g(G)$-set.

Notice that a graph G may have no GTDS. Indeed, whenever a GTDS $S \subseteq V(G)$ exists, every vertex in $V(G)$ must have at least one neighbor and at least one non-neighbor in S. Therefore, G cannot contain either an isolated vertex or a dominating vertex (that is, a vertex of degree $n - 1$). In particular, if G has a vertex of degree $n - 1$, then its complement $\overline{G}$ has an isolated vertex, and hence no total dominating set exists in $\overline{G}$. Thus, throughout this paper, we only consider graphs for which a GTDS exists or, equivalently, graphs with no isolated vertex and no dominating vertex. It follows that whenever $\gamma_t^g(G)$ is well-defined, $4 \leq \gamma_t^g(G) \leq n$ and $n \geq 4$.

The decision version of the GTDS problem is NP-complete [4], and it remains such even for bipartite and chordal graphs [5, 6]. It is also a known fact that if $G_1, G_2, \ldots, G_r$ ($r \geq 2$) are connected components of a graph G, then

$$\gamma_t^g(G) = \sum_{i=1}^{r} \gamma_t(G_i),$$

see [7]. Hence, without loss of generality, we assume that G is a connected graph.

The GTDS is a fundamental concept in graph theory. The dominating set problem and its variants or generalizations have significant applications in network design, facility location, and optimization problems in general [8–10]. One of the earliest significant papers on this topic is the work by Kulli and Janakiram [11], which introduces and studies some basic properties of GTDSs in graphs. Two years later, Haynes et al. [12] complemented this study by providing lower and upper bounds for the problem. More recently, alternative lower and upper bounds were proposed in [4, 13, 14]. Panda and Goyal [5] suggested a polynomial time algorithm for finding a minimum global total k-domination set (GTkDS) for chordal bipartite graphs. They also proposed an algorithm with an approximation ratio of $2(1 + \ln |V|)$ for the GTkDS problem in an arbitrary graph (without presenting any experimental study). In [4,7,15], some lower and upper bounds for the GTkDS problem were suggested. Parra et al., in [16], studied the global dominating set problem. Similar to this work, they proposed an integer linear programming (ILP) formulation, three heuristics, and a purification procedure. However, the problem investigated in [16] concerns global domination sets, while the problem studied in this paper concerns GTDSs.

To the best of our knowledge, no other results for the GTDS problem and its extensions are known. In particular, no exact or approximation algorithms have been previously suggested for the problem targeting any type of graphs. In the previous related work, the parameter $\gamma_t^g(G)$ is determined on the basis of the parameter $\gamma_t(G)$ for only some specific types of graphs (see Section 2 for details). Furthermore, these kinds of estimations are time-consuming and inefficient for synthetic data (see Section 4). In this paper, we propose alternative methods that create optimal and close to the optimal solutions directly for the GTDS problem. In addition, we present an ILP formulation for the problem. The model is solved using the LP-solver CPLEX to obtain optimal solutions. We used 1,130 benchmark instances for the classical domination problem and created six new larger-sized instances with up to 25,000 vertices. The CPLEX solver was able to obtain an optimal GTDS for 1,050 instances with $50 \le |V(G)| \le 10,700$. For approximately, 41% of the instances for which CPLEX succeeded in obtaining an optimal solution, at least one of the three heuristic algorithms proposed here also obtained an optimal solution. Moreover, the average absolute deviation from the optimum was about 1.58 vertices whenever the heuristics did not reach the optimum. For the remaining larger instances, our heuristics produced feasible solutions within 5 minutes in our computational experiments, while Heuristics H2 and H3 have time complexity $O(n^3)$.

The rest of the paper is structured as follows. Our bounds and the ILP model are described in Section 2. In the three consecutive subsections of Section 3, we describe our three heuristics and specify the graph families for which they give an optimal solution. We introduce the special purification procedure that discards redundant vertices in the solutions delivered by our heuristics in Subsection 4 of Section 3. Our experimental study is described in Section 4 (a detailed report is available in [17]), and the concluding notes are in Section 5.

2. Bounds and exact results for $\gamma_t^g(G)$

Throughout this section, we assume that G has no isolated vertices and no dominating vertex.

2.1. Preliminary bounds and known results

We start this subsection with some known results that we will use later on.

Lemma 1 ([4]). *Let G be a graph.*

(i) If $\gamma_t(G) > \Delta(G) + 1$*, then* $\gamma_t^g(G) = \gamma_t(G)$.
(ii) If $n > \Delta(G)(\Delta(G) + 1)$*, then* $\gamma_t^g(G) = \gamma_t(G)$.

Clearly, the definition of a GTDS gives us a direct lower bound for the parameter $\gamma_t^g(G)$.

Remark 1. *Let G be a graph, then*

(i) $\gamma_t^g(G) \geq \max\{\gamma_t(G), \gamma_t(\overline{G})\}$.
(ii) If G is a disconnected graph, then $\gamma_t^g(G) = \gamma_t(G)$.
(iii) $\gamma_t^g(G) \geq 4$.

Proposition 1 ([11]). *Let G be a graph.*

(i) If $diam(G) = 3$, *then* $\gamma_t^g(G) \leq \gamma_t(G) + 2$.
(ii) If $diam(G) = 4$, *then* $\gamma_t^g(G) \leq \gamma_t(G) + 1$.
(iii) If $diam(G) \geq 5$, *then* $\gamma_t^g(G) = \gamma_t(G)$.

Lemma 2. *For any graph G,* $\gamma_t(\overline{G}) = 2$ *if and only if* $diam(G) \geq 3$.

Proof. On the one hand, if $\gamma_t(\overline{G}) = 2$, then $S = \{u, v\}$ is a total dominating set in $\overline{G}$. Thus, u and v are nonadjacent in G, so $d_G(u, v) \geq 2$. Furthermore, if $d_G(u, v) = 2$, then u and v have a common neighbor in G, and hence S is not a total dominating set in $\overline{G}$, which is a contradiction. Therefore, $diam(G) \geq 3$.

On the other hand, if $u, v \in V(G)$ exist such that $d_G(u, v) \geq 3$, then u and v are nonadjacent in G, and every vertex $z \in V(G) \setminus \{u, v\}$ is nonadjacent to at least one of u and v. Hence, $\{u, v\}$ is a total dominating set in $\overline{G}$. □

Lemma 3. *For any graph G,* $\gamma_t^g(G) = \gamma_t(G)$ *if and only if a minimum total dominating set S exist such that, for every vertex* $v \in V$, $S \nsubseteq N(v)$.

Proof. If $\gamma_t^g(G) = \gamma_t(G)$, then a minimum total dominating set S exist that is also a GTDS. Hence, every vertex $v \in V(G)$ has at least one non-neighbor in S; that is, $S \nsubseteq N(v)$. Conversely, suppose that a minimum total dominating set S exist such that $S \nsubseteq N(v)$ for every vertex $v \in V(G)$. Then every vertex of G has a non-neighbor in S. Since S is already a total dominating set of G, it follows that S is a GTDS. Therefore, $\gamma_t^g(G) = \gamma_t(G)$. □

2.2. *Results for graphs G with* $diam(G) = 2$ *and for triangle-free graphs*

Proposition 2 ([4]). *For any* $k \geq 4$, *there is a graph G such that* $diam(G) = diam(\overline{G}) = 2$ *and* $\gamma_t^g(G) = k$.

Figure 1 shows another graph G that is different from the cycle C_5, such that $diam(G) = 2$, $diam(\overline{G}) = 2$, $\gamma_t^g(G) = 4$, $\gamma_t(G) = 3$, and $\delta(G) = 2$.

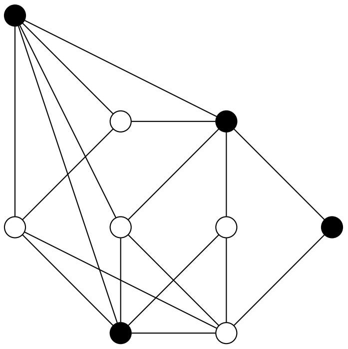


Figure 1. Graph G with $diam(G) = diam(\overline{G}) = 2$.

Proposition 3. *Let G be a graph with* $\delta(G) = 2$, $diam(G) = 2$, *and* $diam(\overline{G}) = 2$. *Then* $\gamma_t^g(G) = 4$.

Proof. Since $diam(\overline{G}) = 2$, it follows from Lemma 2 that $\gamma_t(G) \geq 3$. Let $v \in V(G)$ be a vertex with $N(v) = \{u_1, u_2\}$. Since $diam(G) = 2$, the set $\{u_1, u_2\}$ is a dominating set of G. In particular, u_1 and u_2 are nonadjacent. Now let $w \neq v$ be any neighbor of u_1. Then it is straightforward to verify that $\{v, u_1, u_2, w\}$ is a GTDS. Therefore, $\gamma_t^g(G) = 4$. □

Note that the Petersen graph and its complement have diameter 2, but the minimum degree is 3 and $\gamma_t^g(G) = 5$ (see Figure 2).

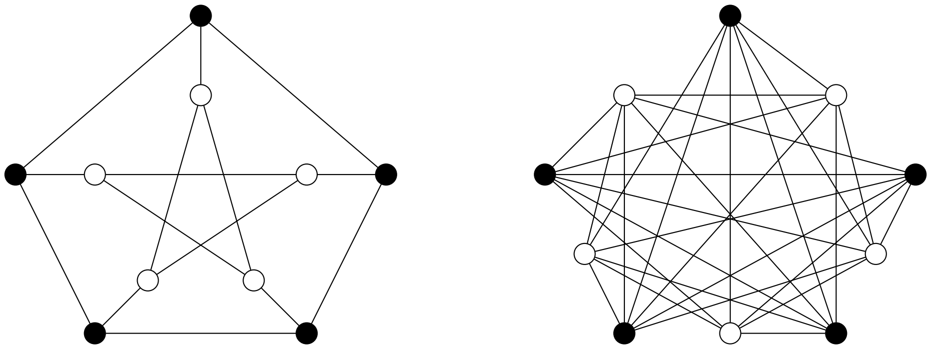


Figure 2. Petersen graph and its complement.

Proposition 4. *If G is a triangle-free graph and* $\gamma_t(G) = 2$, *then* $\gamma_t^g(G) = 4$.

Proof. If $S = \{v_1, v_2\}$ is a minimum total dominating set, since $g(G) \geq 4$, every vertex in $V(G) \setminus S$ has exactly one neighbor in S. Therefore, taking any vertex u_1 non adjacent to v_1 and any vertex u_2 non adjacent to v_2 we see that $S \cup \{u_1, u_2\}$ is a GTDS. □

Theorem 1. *Let G be a triangle-free graph such that* $\gamma_t(G) \geq 3$. *Then* $\gamma_t(G) \leq \gamma_t^g(G) \leq \gamma_t(G) + 1$.

Proof. The lower bound is immediate. To prove the upper bound, let S be a $\gamma_t(G)$-set. First, suppose that for every $v \in S$, we have $1 \leq \delta_S(v) < |S| - 1$. Since G is triangle-free and $|S| \geq 3$, every vertex $v \in S$ has at least one non-neighbor in S. Moreover, every vertex $x \in V(G) \setminus S$ satisfies $1 \leq \delta_S(x) < |S|$ because S is a total dominating set and no vertex outside S can be adjacent to all vertices of S (otherwise, G would contain a triangle). Hence, every vertex of G has both a neighbor and a non neighbor in S, and therefore S is a GTDS. Consequently, $\gamma_t^g(G) = |S| = \gamma_t(G)$.

Now suppose that $v \in S$ exists such that $\delta_S(v) = |S| - 1$. Observe that such a vertex is unique, since $G[S]$ is triangle-free. Let $w \in S \setminus \{v\}$. Since S is a minimum total dominating set, there is a vertex $u \in V(G) \setminus S$ adjacent to w. Moreover, u cannot be adjacent to v, since otherwise u, v, and w would form a triangle. Thus, for every $w \in S \setminus \{v\}$, a vertex $u \in V(G) \setminus S$ exists such that $u \sim w$ and $u \nsim v$. It follows that $S \cup \{u\}$ is a GTDS. Therefore, $\gamma_t^g(G) \leq |S| + 1 = \gamma_t(G) + 1$. In either case, $\gamma_t(G) \leq \gamma_t^g(G) \leq \gamma_t(G) + 1$. □

As an immediate consequence of Theorem 1, we obtain the exact value of $\gamma_t^g(G)$ when $\gamma_t(G) = 3$.

Corollary 1. *Let G be a triangle-free graph such that* $\gamma_t(G) = 3$. *Then* $\gamma_t^g(G) = 4$.

Proof. By Theorem 1, $\gamma_t(G) \leq \gamma_t^g(G) \leq \gamma_t(G) + 1$. Since $\gamma_t(G) = 3$, it follows that $3 \leq \gamma_t^g(G) \leq 4$. On the other hand, by Remark 1(iii), $\gamma_t^g(G) \geq 4$. Hence, $\gamma_t^g(G) = 4$. □

If G is the Petersen graph, then $g(G) = 5$, $\gamma_t(G) = 4$, and $\gamma_t^g(G) = 5$. If G is the Cartesian product $P_2 \Box P_4$ we have $g(G) = 4$, $\gamma_t(G) = 4 = \gamma_t^g(G)$. Moreover, we know that the Petersen graph diameter is 2.

Lemma 4 ([18]). *If G is a graph with $diam(G) = 2$, then $\gamma_t(G) \le \delta(G) + 1$.*

By Lemma 4 and Theorem 1, we obtain the following result.

Proposition 5. *Let G be a triangle-free graph with $diam(G) = 2$. Then $\gamma_t^g(G) \le \delta(G) + 2$.*

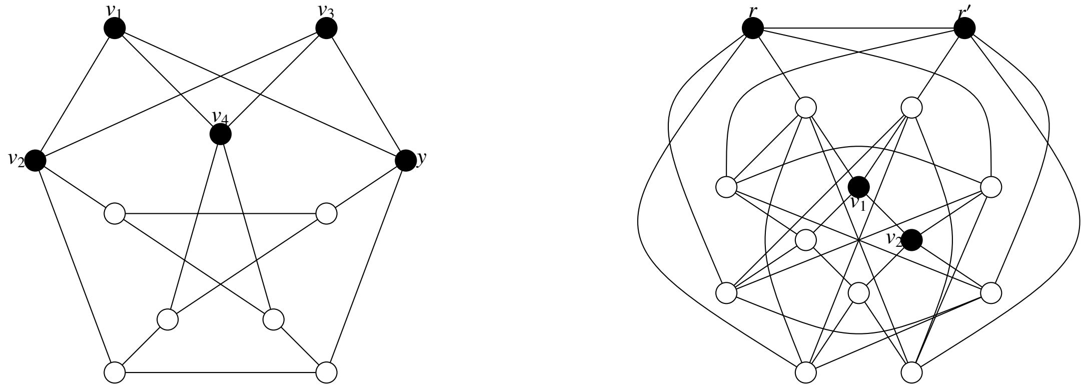
By Theorem 1 and Lemma 3, we obtain the following result.

Proposition 6. *Let G be a triangle-free graph. If, for every minimum total dominating set S of G, a vertex $v \in V(G)$ exists such that $S \subseteq N(v)$, then $\gamma_t^g(G) = \gamma_t(G) + 1$.*

If we consider a graph G with $diam(G) = 2$, it is clear that $g(G) \in \{3, 4, 5\}$. One might think that, for any graph G with $diam(G) = 2$, the value of $\gamma_t^g(G)$ is determined by the girth of G. However, this is not true in general. Indeed, even for graphs with $g(G) = 3$, the value of $\gamma_t^g(G)$ is not uniquely determined. For instance, the graphs shown in Figure 3 both satisfy $diam(G) = 2$ and $g(G) = 3$, but $\gamma_t^g(G) = 5$ and $\gamma_t^g(G') = 4$, respectively. We could think that for any graph G with $diam(G) = 2$ and $g(G) = 4$, we have $\gamma_t^g(G) = 4$. However, that is not true for $g(G) = 4$. For instance, in the graphs G and G' shown in Figure 4, we see that $\gamma_t^g(G) = 5$ and $\gamma_t^g(G') = 4$.

(a) $\gamma_t^g(G) = 5$ **(b)** $\gamma_t^g(G') = 4$

Figure 3. Graphs G and G' with diameter 2 and girth 3.



(a) G **(b)** G'

Figure 4. Graphs G and G' with diameter 2 and girth 4.

Lemma 5 ([18]). *If G is a diameter 2 graph of order n with girth 5, then* $\gamma_t(G) = 1 + \sqrt{n-1}$.

Proposition 7. *Let G be a graph with* $diam(G) = 2$ *and* $g(G) = 5$. *Then* $\gamma_t^g(G) = 4$ *if and only if G is a cycle with five vertices.*

Proof. We know that $\gamma_t^g(C_5) = 4$. If $diam(G) = 2$, $g(G) = 5$ and n is the order of G, by Lemma 5, we know that $\gamma_t(G) = 1 + \sqrt{n-1}$. If we suppose that $\gamma_t^g(G) = 4$, since $\gamma_t(G) \leq \gamma_t^g(G) \leq \gamma_t(G) + 1$, we have that $\gamma_t(G) \in \{3, 4\}$, so $n \in \{5, 10\}$; consequently, G is a cycle of five vertices or the Petersen graph. But if G is the Petersen graph, we have $\gamma_t^g(G) = 5$. □

Given a set of vertices S and $v \in S$, we say that u is a private neighbor of v with respect to S if $u \in V(G) \setminus S$, $u \in N(v)$, and $u \notin N(v')$ for any $v' \in S \setminus \{v\}$. The set of all private neighbors of v with respect to S is denoted by $pn[v, S]$. Thus, $pn[v, S]$ contains only vertices outside S.

Lemma 6. *Let* $G = (V, E)$ *be a graph of order n with* $\Delta(G) \leq n - 2$. *If a vertex* $v \in V$ *exists such that* $N(v)$ *is a dominating set, then there is a GTDS S of cardinality* $\delta(v) + 3$.

Proof. Let $A = N(v) = \{u_1, \ldots, u_t\}$, where $t = \delta(v)$. Since $\Delta(G) \leq n - 2$, there is at least one vertex outside $N[v]$. We distinguish cases.

Case 1. The values $u_i \in A$ and $w \in pn[u_i, A]$ exist. In this case, choose any vertex $z \notin N[u_i]$. Then one can verify that $A \cup \{v, w, z\}$ is a GTDS. Hence, G has a GTDS of cardinality $t + 3 = \delta(v) + 3$.

Case 2. No vertex in A has a private neighbor with respect to A. We now consider the structure of the subgraph induced by A. *Subcase 2.1.* Here, $\delta_A(u_i) < t - 1$ for every $i \in \{3, \ldots, t\}$. Choose the vertices $w_1 \notin N[u_1]$ and $w_2 \notin N[u_2]$. Then $A \cup \{v, w_1, w_2\}$ is a GTDS. Therefore, the result follows. *Subcase 2.2.* There are vertices in A adjacent to all other vertices of A. Assume, without loss of generality, that $\delta_A(u_i) = t - 1$ for $i \in \{1, \ldots, s\}$, where $s \geq 3$. For each $i \in \{1, \ldots, s\}$, choose a vertex $w_i \in V(G) \setminus N[v]$ such that $w_i \notin N(u_i)$. Then $S = A \cup \{v, w_1, \ldots, w_s\}$ is a GTDS.

Since no vertex in A has a private neighbor with respect to A, removing u_1 does not destroy domination inside A. Thus, $S_1 = S \setminus \{u_1\}$ is still a GTDS.

Now let $A_1 = A \setminus \{u_1\}$. If $u_i \in A_1$ exists such that $pn[u_i, A_1] \neq \emptyset$, choose $z \in pn[u_i, A_1]$ and $w_i \notin N[u_i]$. Then $A_1 \cup \{v, w_1, z, w_i\}$ is a GTDS of cardinality $t + 3$. If no such vertex exists, remove another vertex from A and repeat the same argument. Continuing in this way, we eventually obtain a GTDS of cardinality $t + 3 = \delta(v) + 3$. □

Proposition 8. *If G is a graph with* $diam(G) = 2$, *then* $\gamma_t^g(G) \leq \delta(G) + 3$.

Proof. Note that if G is a graph with $diam(G) = 2$ and v a vertex of minimum degree, then $N(v)$ is a dominating set of G. By Lemma 6, we have $\gamma_t^g(G) \leq \delta(G) + 3$. □

The bound in Proposition 8 is sharp. Indeed, the graph shown in Figure 3a satisfies $\delta(G) = 2$, $diam(G) = 2$, and $\gamma_t^g(G) = 5 = \delta(G) + 3$.

Proposition 9. *For every* $k \geq 2$, *there is a graph* $G = (V, E)$ *with a minimum degree k,* $diam(G) = 2$, $g(G) = 3$, *and* $\gamma_t^g(G) = k + 3$.

Proof. We consider a complete graph K_{k+1}, whose vertices are $v, u_1, u_2, \ldots, u_k$, and another $2k$ vertices $w_1, w_2, \ldots, w_k, z_1, z_2, \ldots, z_k$ with the following edges:

(1) $u_iw_j \in E$ for every $i \neq j$;
(2) $u_iz_j \in E$ for every i, j;
(3) $z_iw_j \in E$ for every i, j.

Since $N[u_i] = V \setminus \{w_i\}$ for every $i \in \{1, \ldots, k\}$, then w_i (the black vertices in Figure 5) must belong to any GTDS for every $i \in \{1, \ldots, k\}$. Finally, any vertex in $\{u_1, u_2, \ldots, u_k\}$ with any vertex in $\{z_1, z_2, \ldots, z_k\}$ would form a star with $\{w_1, w_2, \ldots, w_k\}$, so we would need another vertex to obtain a GTDS. □

For instance, $\{v, u_1, z_1, w_1, w_2, \ldots, w_{\delta(G)}\}$ (the black and gray vertices in Figure 5) is a minimum GTDS in G.

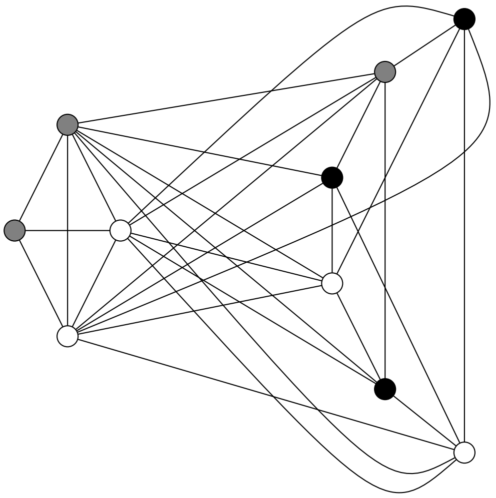


Figure 5. A graph G such that $\delta(G) = 3$ and $\gamma_t^g(G) = 6$.

Lemma 7 ([7]). *Let G be a graph. If* $\gamma_t(G) \leq \Delta(G) + 1$, *then* $\gamma_t^g(G) \leq \Delta(G) + 2$.

By Lemmas 4 and 7, we obtain the following result, which improves Proposition 8 for regular graphs. For completeness, we provide a new proof.

Proposition 10. *If G is a graph with* $diam(G) = 2$, *then* $\gamma_t^g(G) \leq \Delta(G) + 2$.

Proof. Let v be a vertex of G with degree $\Delta(G)$, and let $A = N_G(v)$. Then $|A| = \Delta(G)$. Since $diam(G) = 2$, there is a vertex $w \in V(G) \setminus N_G[v]$. Consider the set $S = A \cup \{v, w\}$. We claim that S is a GTDS of G. Indeed, both v and w have a neighbor in A, and they are nonadjacent. Moreover, every vertex in A has at least one neighbor and at least one non-neighbor in S. Finally, every vertex in $V(G) \setminus (N_G[v] \cup \{w\})$ also has at least one neighbor and at least one non-neighbor in S. Therefore, S is a GTDS of G, and hence $\gamma_t^g(G) \leq |S| = \Delta(G) + 2$. □

Proposition 11. *If G is a planar and triangle-free graph with* $diam(G) = 2$, *then* $\gamma_t^g(G) = 4$.

Proof. By Theorem 1, we directly see that $\gamma_t^g(G) \leq \gamma_t(G) + 1$. In [19], the following result is proved: If G is a planar graph with $diam(G) = 2$, then $\gamma_t(G) \leq 3$. It directly follows that $\gamma_t^g(G) \leq \gamma_t(G) + 1 \leq 4$. So, $\gamma_t^g(G) = 4$. □

2.3. Other exact results and graph operations

Proposition 12. *Let G be a graph with $\delta(G) = 1$, $\{u_1, u_2, \dots, u_r\}$ being the leaves of G and $\{v_1, v_2 \dots, v_t\}$ the corresponding supports. If $G' = G - \{u_1, u_2, \dots, u_r\}$ and $\gamma_t^g(G')$ exists, then $\gamma_t^g(G) \le \gamma_t^g(G') + t$.*

Proof. Let $G' = G - \{u_1, u_2, \dots, u_r\}$ and S' be a $\gamma_t^g(G')$-set. Define $S = S' \cup \{v_1, v_2, \dots, v_t\}$. First, since S' is a total dominating set of G', every vertex of G' is adjacent in G' (and hence in G) to a vertex of $S' \subseteq S$. Moreover, each leaf u_i is adjacent in G to its corresponding support vertex, which belongs to S. Thus, S is a total dominating set of G. Now, since S' is a GTDS of G', every vertex of G' has a neighbor in $\overline{G'}$ belonging to S', and therefore also one in $\overline{G}$ belonging to S. Moreover, each leaf u_i is adjacent in $\overline{G}$ to every vertex except its support vertex, so it also has a neighbor in S. Hence, S is a total dominating set of $\overline{G}$. Therefore, S is a GTDS of G, and $\gamma_t^g(G) \le \gamma_t^g(G') + t$. □

Figure 6 shows an example for which the previous upper bound is tight.

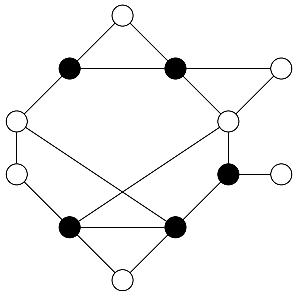


Figure 6. A graph G such that $diam(G) = 4$ and $\gamma_t^g(G) = \gamma_t^g(G') + 1$.

Proposition 13 ([7]). *Let P_n and C_n be the path and cycle graphs of order n, respectively, with $n \ge 4$. Then the following global total domination numbers hold:*

$$\gamma_t^g(P_n) = \gamma_t^g(C_n) = \begin{cases} 4 & if\ n = 4,\ 5 \\ \frac{n}{2} + 1 & if\ n \equiv 2 (mod\ 4) \\ \lceil \frac{n}{2} \rceil & otherwise. \end{cases}$$

Proposition 14 ([7]). *Let G be a cubic graph of order n, then*

$$\gamma_t^g(G) = \begin{cases} \gamma_t(G) & if\ \gamma_t(G) > 4 \\ 5 & if\ \gamma_t(G) = 4\ and\ for\ any\ \gamma_t(G)\text{-}set\ S\ of\ G\ is\ G[S] = K_{1,3} \\ 4 & otherwise. \end{cases}$$

Let $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$ be graphs. The union of graphs $G_1 + G_2$ results in $G'(V', E')$ where $V' = V_1 \cup V_2$ and $E' = E_1 \cup E_2 \cup \{(x, y) : x \in V_1 \wedge y \in V_2\}$.

Proposition 15. *Let $G' = G_1 + G_2$ be a joint graph, where $|V(G_1)|, |V(G_2)| \ge 3$, $\Delta(G_1) \le |V(G_1)| - 2$, and $\Delta(G_2) \le |V(G_2)| - 2$, then $\gamma_t^g(G') = \gamma_t(\overline{G_1}) + \gamma_t(\overline{G_2})$.*

Proof. Since $\Delta(G_1) \le |V(G_1)| - 2$ and $\Delta(G_2) \le |V(G_2)| - 2$, then $\overline{G_1}$ and $\overline{G_2}$ are the connected components of the disconnected graph $\overline{G'}$. By Remark 1 (ii), $\gamma_t^g(\overline{G'}) = \gamma_t(\overline{G_1}) + \gamma_t(\overline{G_2})$. Note that $\gamma_t^g(G') = \gamma_t^g(\overline{G'})$, then $\gamma_t^g(G') = \gamma_t(\overline{G_1}) + \gamma_t(\overline{G_2})$. □

As a direct consequence, the following corollary follows.

Corollary 2. *Let $K_{n,m}$ be a complete bipartite graph of order $n + m$ and $n, m \geq 2$, then $\gamma_t^g(K_{n,m}) = 4$.*

Proposition 16. *If G is a graph such that $\gamma_t^g(G) = \gamma_t(G) + \gamma_t(\overline{G})$, then $diam(G), diam(\overline{G}) \leq 3$.*

Proof. If $diam(G) = 4$, by Proposition 1, $\gamma_t^g(G) \leq \gamma_t(G) + 1 < \gamma_t(G) + 2$. By Lemma 2, $\gamma_t(\overline{G}) = 2$, then $\gamma_t^g(G) < \gamma_t(G) + \gamma_t(\overline{G})$, which is a contradiction. If $diam(G) \geq 5$, then $\gamma_t^g(G) = \gamma_t(G) < \gamma_t(G) + 2 \leq \gamma_t(G) + \gamma_t(\overline{G})$, which is also a contradiction. Since $\gamma_t^g(G) = \gamma_t^g(\overline{G})$, the same happens with $diam(\overline{G})$. □

Let G' and H be two graphs of order n' and m, respectively. The corona product $G' \odot H$ is defined as the graph obtained from G' and H by taking one copy of G' and n' copies of H and joining each vertex from the ith-copy of H with the ith-vertex of G by an edge.

Proposition 17. *Let G' be a graph of order $n' \geq 2$ (with no isolated vertices), let H be any graph, and consider the corona product $G = G' \odot H$. In this case, we have the following:*

(a) *If G' is a complete graph, then $\gamma_t^g(G) = n' + 2$.*
(b) *If G' is not a complete graph and $\Delta(G') = n' - 1$, then $\gamma_t^g(G) = n' + 1$.*
(c) *If $\Delta(G') < n' - 1$, then $\gamma_t^g(G) = n'$.*

Proof. The cases $n' = 2$ and $n' = 3$ can be checked directly. Thus, we may assume $n' \geq 4$. If S is a GTDS and we use H_v to denote the copy of H corresponding to the vertex $v \in V(G')$, then $|S \cap (\{v\} \cup V(H_v))| \geq 1$ for every $v \in V(G')$, and so $|S| \geq n'$. Moreover, if $v \in V(G') \setminus S$, then necessarily $|S \cap V(H_v)| \geq 2$. Hence, if replacing such choices is necessary, we may assume that $V(G') \subseteq S$.

(a) If G' is complete, then every vertex of $V(G')$ is adjacent to all other vertices of $V(G')$. Since every vertex in a GTDS must also have a non-neighbor in the set, at least two additional vertices from copies of H are required. Hence, $\gamma_t^g(G) = n' + 2$.
(b) If G' is not complete and $\Delta(G') = n' - 1$, then there is a vertex of G' adjacent to all other vertices of G'. Therefore, one additional vertex in a suitable copy of H is needed so that every vertex in the set has a non-neighbor in the set. Hence, $\gamma_t^g(G) = n' + 1$.
(c) If $\Delta(G') < n' - 1$, then every vertex of G' has a non-neighbor in $V(G')$. Since $V(G')$ totally dominates all vertices of G, it follows that $V(G')$ is a GTDS. Hence, $\gamma_t^g(G) = n'$.

□

2.4. *ILP formulation*

We next present an ILP formulation for the minimum GTDS problem. Our model can be seen as a natural adaptation of the formulation given in [16] for the global domination problem, modified here to capture the *total* domination requirement in both G and $\overline{G}$.

Given a graph $G = (V, E)$ of order n, let $A = (a_{ij})_{n\times n}$ be its adjacency matrix, where

$$a_{ij} = \begin{cases} 1, & \text{if } (i, j) \in E(G), \\ 0, & \text{otherwise.} \end{cases}$$

Moreover, let $B = (b_{ij})_{n\times n}$ be the adjacency matrix of $\overline{G}$. The introduction of B is only for notational convenience, allowing us to write the total domination constraints in G and $\overline{G}$ in a symmetric way.

Given a GTDS S of cardinality $\gamma_t^g(G)$ and $i \in V(G)$, we define our decision variables as follows:

$$x_i = \begin{cases} 1, & \text{if } i \in S \\ 0, & \text{otherwise.} \end{cases} \tag{2.1}$$

The *GTDS problem* can now be formulated as follows:

$$\min \sum_{i=1}^{n} x_i, \tag{2.2}$$

$$subject\ to: \quad \sum_{j=1}^{n} a_{ij}x_j \geq 1, \ \forall i \in V(G), \tag{2.3}$$

$$\sum_{j=1}^{n} b_{ij}x_j \geq 1, \ \forall i \in V(G), \tag{2.4}$$

$$x_i \in \{0, 1\}. \tag{2.5}$$

Here the objective function is $\sum_{i=1}^{n} x_i$, which we wish to minimize (2.2). We have $2n$ restrictions (2.3) and (2.4), guaranteeing that for each $j \in V(G)$ and each $j \in V(\overline{G})$, at least one of the vertices in S is adjacent to j. Since every feasible instance considered in this paper satisfies $\gamma_t^g(G) \geq 4$, this bound is implicit and does not need to be included explicitly in the formulation. Note that if we want to find a *minimum total dominating set*, it suffices to remove the set of constraints (2.4).

3. Heuristic algorithms

We need to introduce additional notations and definitions. A *partial GTDS* of a graph $G = (V, E)$ is a set $S \subseteq V(G)$ such that S is a GTDS of the induced subgraph $G[S]$. Our heuristics run in a number of iterations. We will use S_h for the partial GTDS already constructed by iteration h. That is, every vertex in S_h has at least one neighbor and at least one non-neighbor in S_h, although S_h may not yet be a GTDS of G.

For a set $S \subseteq V(G)$, let $A(S) = \{x \in V(G) \setminus S : N(x) \cap S = \emptyset\}$. The *active degree* of a vertex $x \in V(G) \setminus S$ with respect to S is $ad_S(x) = |N(x) \cap A(S)|$. In other words, $ad_S(x)$ is the number of vertices not yet dominated by S that are adjacent to x. When $S = S_h$, we simply write $ad_h(x)$.

At iteration h, we also define the following sets of vertices.

- Let D_h be the set consisting of all vertices $v \notin S_h$ such that $|N_{S_h}(v)| \geq 1$ and $|\overline{N_{S_h}}(v)| \geq 1$, i.e., each vertex from the set D_h has at least one neighbor and one non-neighbor in set S_h.
- Let A_h be the set of vertices from set $V(G)$ that have no neighbor in set S_h, i.e., $N(A_h) \cap S_h = \emptyset$.
- Let B_h be the set of vertices from set $V(G)$ such that every vertex from this set is a neighbor of all the vertices from set S_h; in other words, for every $v \in B_h$, $\overline{N_{S_h}}(v) = \emptyset$.

Informally, set A_h (set B_h, respectively), consists of vertices which are not dominated (globally dominated, respectively) by iteration h. Relying on $\gamma_t^g(G) \geq 4$ (see Section 1), our heuristics initiate by setting $S_0 := \{u, v, q, w\}$ (in the subsequent sections, we will specify how our heuristics select these vertices). Notice that since S_0 is a partial global total dominating set, each vertex u, v, q, and w has both a neighbor and a non-neighbor in S_0. Moreover, $B_0 = N(u) \cap N(v) \cap N(q) \cap N(w)$, $A_0 = V(G) \setminus (N[u] \cup N[v] \cup N[q] \cup N[w])$, and $D_0 = ((N(u) \cup N(v) \cup N(q) \cup N(w)) \setminus (N(u) \cap N(v) \cap N(q) \cap N(w))) \setminus S_0$. The next remark immediately follows from the definitions above.

Remark 2. *The sets A_h, B_h, D_h, and S_h form a partition of set $V(G)$, i.e., they are disjoint and $A_h \cup B_h \cup D_h \cup S_h = V(G)$.*

Lemma 8. *If $A_0 = B_0 = \emptyset$ and $S_0 = \{u, v, q, w\}$, then $D_0 = V(G) \setminus S_0$, and S_0 is an optimal solution.*

Proof. By Remark 2 and the fact that $A_0 = B_0 = \emptyset$, $D_0 = V(G) \setminus S_0$. Hence, D_0 is a globally dominant set. At the same time, set V is also totally dominated by set S_0 since the vertices u, v, q, and w have both a neighbor and a non-neighbor in set S_0. Hence, S_0 is a GTDS. It is also minimum by Remark 1. □

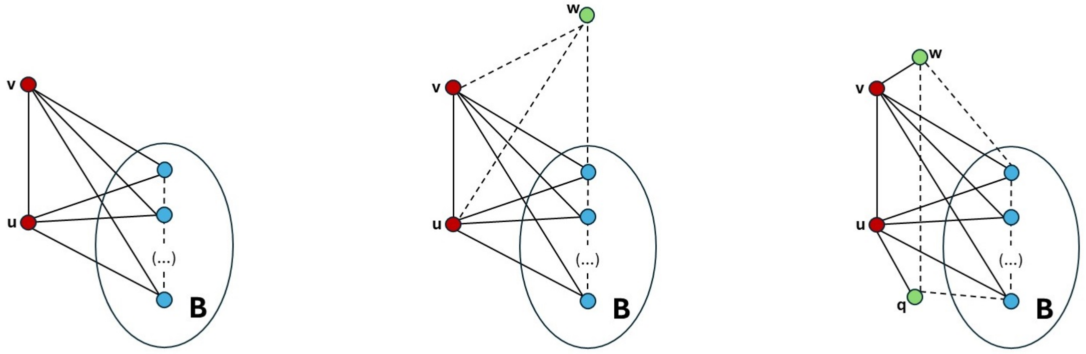
In the rest of this section, without loss of generality, assume $n > 4$ (by Remark 1) and that $\Delta(G) \leq n - 2$ since, by the definition of our domination problem, the graph $\overline{G}$ has no isolated vertex.

Lemma 9. *There is a partial GTDS $S_0 = \{u, v, q, w\}$ such that $D_0 \neq \emptyset$.*

Proof. We show the lemma first for Heuristics H2 and H3, which use the same selection rule for vertices u, v, q, and w. In particular, let u be a vertex of degree $\Delta(G)$, and let v be the neighbor of vertex u having the minimum degree, as defined in Heuristics H2 and H3.

With respect to the initial set $\{u, v\}$, Heuristic H2/H3 determines a vertex $w \in V(G) \setminus \{u, v\}$ with maximum active degree, that is, a vertex maximizing $ad_{\{u,v\}}(w)$, under the condition $\{u, v\} \not\subseteq N(w)$. Therefore, w is the vertex with the largest number of neighbors that are not yet dominated by either u or v (see Figure 7b).

Note that if w does not exist, any vertex in set $V(G) \setminus \{u, v\}$ must be a neighbor of both u and v and hence $\delta(u) = \delta(v) = n - 1$ (see Figure 7a). But this cannot happen by the definition of our domination problem. Thus, the vertex w exists.



(a) (b) (c)

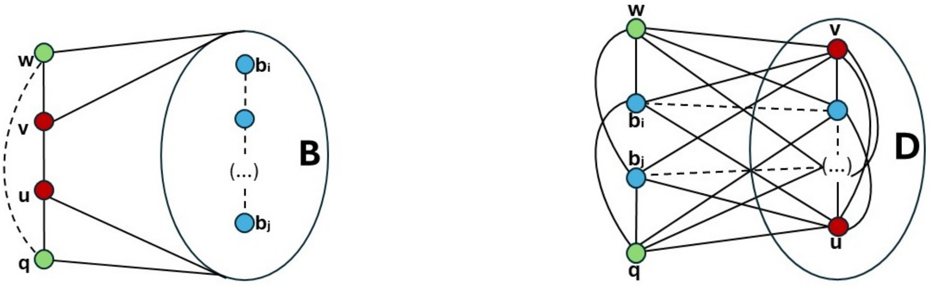
Figure 7. Construction of S_0.

Then the heuristics determine the last vertex q, which is one with the smallest degree having at least one neighbor and one non-neighbor in set $\{u, v, w\}$. Below, we show that this kind of a selection is possible. Consider the following cases.

Case 1: Vertex w is adjacent to v or u (note that it cannot be adjacent to both of them). If $(v, w) \in E(G)$ and q is a neighbor of both u and v, then $\delta(v) = n - 1$, which is not possible; see Figure 7b, where Set B represents the set of vertices in $V(G)$ which are neighbors of both u and v. Hence, $q \notin B$. Since $(v, w) \in E(G)$ and $\delta(u) \geq \delta(v)$, $(q, u) \in E(G)$. We showed that $S_0 = \{u, v, w, q\}$ exists; see Figure 7c.

It remains to show that $D_0 \neq \emptyset$. If $D_0 = \emptyset$, then also $A_0 = \emptyset$; see Figure 8a. Hence every vertex outside S_0 is already dominated by S_0, and therefore its active degree with respect to S_0 is 0. In particular, every vertex in set $B_0 = V(G) \setminus S_0$ has an active degree 0.

We claim that $\Delta(G[B_0]) \leq |B_0| - 2$. Indeed, recall that $\delta(u) = |B| + 2$ and that it is a vertex with a maximum degree in set $V(G)$. At the same time, every vertex in Set B_0 has four neighbors in Set S_0, and it may have one or more neighbor in set B_0 itself. Now it easily follows that the former vertex cannot have more than $|B_0| - 2$ neighbors in set B_0. It follows that every vertex $b_i \in B_0$ must have a neighbor and a non-neighbor b_j in set B_0. Then S_0 can be updated as $S_0 = \{q, b_j, b_i, w\}$; see Figure 8b. Thus D_0 contains the vertices u and v and hence $D_0 \neq \emptyset$. The case where $(u, w) \in E(G)$ is similarly treated.



(a) **(b)**

Figure 8. Update of S_0 if $D_0 = \emptyset$ and $A_0 = \emptyset$.

Case 2: Similar to Case 1, we first define the fourth vertex q to be included in set S_0. We have $(w, u) \notin E(G)$ and $(w, v) \notin E(G)$. Consider two possible cases when the active degree of vertex w is 0 and when it is positive. In the former case, any neighbor of the vertex w must be dominated by the vertices u or v and hence a neighbor of vertex w is either in set B_0 or it is in set D_0. If now the active degree of vertex w is positive, a vertex $q \notin B$ exists such that $(q, u) \notin E(G)$ and $(q, v) \notin E(G)$. It follows that in both cases, $S_0 = \{u, v, q, w\}$ (see Figure 9a).

(a) **(b)**

Figure 9. Update of S_0 if $D_0 = \emptyset$ and $A_0 = \emptyset$.

It remains to show that $D_0 \neq \emptyset$. If $D_0 = \emptyset$, then also $A_0 = \emptyset$; see Figure 9a. Hence every vertex outside S_0 is already dominated by S_0, and therefore, its active degree with respect to S_0 is 0. In particular, every vertex in set $B_0 = V \setminus S_0$ has an active degree 0.

We claim that $\Delta(G[B_0]) \leq |B_0| - 3$. Indeed, recall that $\delta(u) = |B| + 1$ and that it is a vertex with a maximum degree in set $V(G)$. At the same time, every vertex in set B_0 has four neighbors in set S_0, and it may have one or more neighbor in set B_0 itself. Now it easily follows that the former vertex cannot have more than $|B_0| - 3$ neighbors in set B_0. It follows that every vertex $b_i \in B_0$ must have a neighbor and a non neighbor b_j in set B_0. Then S_0 can be updated as $S_0 = \{u, b_j, b_i, w\}$; see Figure 9b. Thus D_0 contains the vertices q and v and hence $D_0 \neq \emptyset$.

We showed the lemma for Heuristics H2 and H3. Now, Heuristic H1 considers all possible combinations of the four vertices of set $V(G)$ and verifies the required condition for set S_0. For each created set S_0, it forms the corresponding set D_0 and selects a generated set S_0 to which the maximum cardinality set D_0 corresponds. In particular, sets S_0 and D_0, as formed for Heuristics H2 and H3, will be created. The lemma is proved. □

Although the heuristics proposed in this paper are inspired by the greedy framework introduced in [16], and particularly by the use of a dynamic partition of the vertex set together with an active degree type selection rule, the present problem imposes substantially stronger structural constraints. In the global total domination setting, a candidate vertex cannot be chosen only according to its domination gain in G and $\overline{G}$; it must also preserve the fact that the current partial solution remains a partial GTDS. For this reason, the heuristics developed here require a different feasible initialization, we restrict the admissible candidates to vertices in D_h and use the quantity $GTAD_{A_h,B_h}(v)$ only within this stricter feasibility framework. Thus, while the general greedy philosophy is inherited from [16], the algorithmic construction is specifically redesigned for global total domination.

On the basis of Lemma 9, suppose that $D_h \neq \emptyset$, and let us consider the vertex $v \in D_h$ with $N(v) \cap S_h \neq \emptyset$ and $|S_h \setminus N(v)| \geq 1$, a candidate to be included in the set S_h in iteration $h + 1$ (note that the first and second conditions guarantee the total and global dominance of the vertices in set S_{h+1}).

Analogous to the notion of global active degree introduced in [16], the *global total active degree* of vertex v in iteration h is defined as

$$GTAD_{A_h,B_h}(v) = |N_{A_h}[v] \cup \overline{N}_{B_h}[v]|.$$

In our heuristics, we proceed at iteration $h > 0$ if $A_h \neq \emptyset$ or $B_h \neq \emptyset$. We halt when set S_h is a GTDS, i.e., $A_h = \emptyset$ and $B_h = \emptyset$.

Lemma 10. *Let $h > 0$. Suppose that $A_h \neq \emptyset$ or $B_h \neq \emptyset$, and assume that $GTAD_{A_h,B_h}(v) = 0$ for every $v \in D_h$. Then there is a set $X \subseteq A_h \cup B_h$ with $2 \leq |X| \leq 3$ such that $S_{h+1} := S_h \cup X$ is a partial GTDS.*

Proof. By the definition of $GTAD_{A_h,B_h}(v)$, the assumption $GTAD_{A_h,B_h}(v) = 0$ for every $v \in D_h$ implies that $N(v) \cap A_h = \emptyset$ and $B_h \subseteq N(v)$ for every $v \in D_h$. In other words, no vertex of D_h is adjacent to a vertex of A_h, and every vertex of D_h is adjacent to every vertex of B_h. We distinguish three cases.

Case 1: $A_h \neq \emptyset$ and $B_h = \emptyset$. Let $a \in A_h$. Since $a \in A_h$, we have $N(a) \cap S_h = \emptyset$. Moreover, by the observation above, a has no neighbors in D_h. Hence, every neighbor of a must belong to A_h, which means that there are no edges between A_h and $V(G) \setminus A_h$. This contradicts the connectedness of G. Therefore, this case cannot occur.

Case 2: $A_h = \emptyset$ and $B_h \neq \emptyset$. If $G[B_h]$ is not complete, then there are two non adjacent vertices $b, b' \in B_h$. Since every vertex of B_h is adjacent to every vertex of S_h, both b and b' have a neighbor in S_h and a non-neighbor in $\{b, b'\}$. Hence $S_{h+1} := S_h \cup \{b, b'\}$ is a partial GTDS. Suppose now that $G[B_h]$ is complete, and let $b \in B_h$. Then b is adjacent to every vertex of S_h, every vertex of D_h, and every vertex of $B_h \setminus \{b\}$. Since $A_h = \emptyset$, it follows that b is adjacent to every other vertex of G (that is, $\delta(b) = n - 1$), contradicting the assumption $\Delta(G) \leq n - 2$. Thus this situation is impossible.

Case 3: $A_h \neq \emptyset$ and $B_h \neq \emptyset$. We divide this case into four subcases.

Subcase 3.1: $|A_h| = 1$ and $|B_h| = 1$. Let $A_h = \{a\}$ and $B_h = \{b\}$. Since $a \in A_h$, we have $N(a) \cap S_h = \emptyset$, and by the observation above, a has no neighbors in D_h. As G is connected, a must be adjacent to b. But then b is adjacent to every vertex of S_h, every vertex of D_h, and also to a, so $\delta(b) = n - 1$, again contradicting $\Delta(G) \leq n - 2$. Therefore, this subcase cannot occur.

Subcase 3.2: $|A_h| = 1$ and $|B_h| > 1$. Let $A_h = \{a\}$. Since a has no neighbors in $S_h \cup D_h$ and G is connected, there is a vertex $b \in B_h$ such that $ab \in E(G)$. Since $\Delta(G) \leq n - 2$, vertex b cannot be adjacent to all vertices of G. As b is adjacent to every vertex of S_h, every vertex of D_h, and also to a, it follows that a vertex $b' \in B_h$ exists such that $bb' \notin E(G)$. Now, in the set $S_h \cup \{a, b, b'\}$, the vertex a has a neighbor (b) and a non-neighbor (b'), the vertex b has a neighbor in S_h and a non-neighbor (b'), and the vertex b' has a neighbor in S_h and a non-neighbor (b). Therefore, $S_{h+1} := S_h \cup \{a, b, b'\}$ is a partial GTDS.

Subcase 3.3: $|A_h| > 1$ and $|B_h| = 1$. Let $B_h = \{b\}$. Since $\Delta(G) \leq n - 2$, the vertex b cannot be adjacent to all vertices of A_h; otherwise, as it is also adjacent to all vertices of S_h and D_h, we would have $\delta(b) = n - 1$. Hence, a vertex $a \in A_h$ exists such that $ab \notin E(G)$. Because G is connected and a has no neighbors in $S_h \cup D_h$, the vertex a must have a neighbor $a' \in A_h$. Then, in the set $S_h \cup \{a, a', b\}$, the vertex a has a neighbor (a') and a non-neighbor (b), the vertex a' has a neighbor (a) and a non-neighbor in S_h (since $a' \in A_h$), and the vertex b has a neighbor in S_h and a non-neighbor (a). Hence $S_{h+1} := S_h \cup \{a, a', b\}$ is a partial GTDS.

Subcase 3.4: $|A_h| > 1$ and $|B_h| > 1$. If $G[B_h]$ is not complete, then there are two non-adjacent vertices $b, b' \in B_h$ and, exactly as in Case 2, $S_{h+1} := S_h \cup \{b, b'\}$ is a partial GTDS. Assume now that $G[B_h]$ is complete. Let $b \in B_h$. Since $\Delta(G) \leq n - 2$, the vertex b must have a non-neighbor in A_h; otherwise, being adjacent to all vertices of S_h, D_h, and B_h, it would follow that $\delta(b) = n - 1$. Thus $a \in A_h$ exists such that $ab \notin E(G)$. Since a has no neighbors in $S_h \cup D_h$ and G is connected, the vertex a must have a neighbor $a' \in A_h \cup B_h$. If $a' \in B_h$, then $a'b \in E(G)$ because $G[B_h]$ is complete. If $a' \in A_h$, then a' is non-adjacent to every vertex of S_h, and hence a' has a non-neighbor in the enlarged set. In either case, in the set $S_h \cup \{a, a', b\}$, the vertex a has a neighbor (a') and a non-neighbor (b), the vertex b has a neighbor in S_h and a non-neighbor (a), and the vertex a' has a neighbor (a) and a non-neighbor in the enlarged set. Therefore, $S_{h+1} := S_h \cup \{a, a', b\}$ is a partial GTDS.

In all possible cases, a set $X \subseteq A_h \cup B_h$ with $2 \leq |X| \leq 3$ exists such that $S_{h+1} := S_h \cup X$ is a partial GTDS. □

Lemma 10 guarantees that whenever the greedy rule cannot make progress, the construction can still be continued by adding a small set of two or three vertices from $A_h \cup B_h$ in such a way that the enlarged set remains a partial GTDS. This prevents the heuristic from stalling before reaching a feasible solution.

Next, we describe our heuristics in detail specifying, in particular, how set S_0 in each of the heuristics is formed and how each current set S_h is extended at iteration h.

3.1. Heuristic H1

Heuristic H1 forms the initial set S_0 with the vertices u, v, q, and w such that the cardinality of the set $(N[u] \cup N[v] \cup N[q] \cup N[w]) \setminus (N[u] \cap N[v] \cap N[q] \cap N[w])$ is maximized (and also that the vertices u, v, q, and w have both a neighbor and a non-neighbor in the formed set $\{u, v, q, w\}$, which is a common requirement specified earlier). This goal is reached by a mere enumeration of all possible subsets of vertices (see Lemma 9(existence) and the pseudo-code below of Algorithm 1). At every succeeding iteration $h > 0$, a vertex $v_h \in D_{h-1}$ with the maximum global total active degree with respect to the sets A_{h-1} and B_{h-1} is determined. Then the vertex v_h is moved to the set S_{h-1}, resulting in the extended set S_h, and the sets A_h, B_h, and D_h are updated as specified earlier.

Algorithm 1 Heuristic H1.

```
Input: A graph $G = (V, E)$.
Output: A GTDS $S_h$.
$h := 0$;
$\{u, v, q, w\}$ := four vertex with the largest possible cardinality of the set
            $(N[u] \cup N[v] \cup N[q] \cup N[w]) \setminus (N[u] \cap N[v] \cap N[q] \cap N[w])$,
            where the vertices $u$, $v$, $q$, and $w$ have a neighbor and a non-
            neighbor in the set $\{u, v, q, w\}$;
$S_0 := \{u, v, q, w\}$;
$D_0 := ((N(u) \cup N(v) \cup N(q) \cup N(w)) \setminus (N(u) \cap N(v) \cap N(q) \cap N(w))) \setminus S_0$;
$A_0 := V(G) \setminus (N[u] \cup N[v] \cup N[q] \cup N[w])$;
$B_0 := (N(u) \cap N(v) \cap N(q) \cap N(w)) \setminus S_0$;
   { iterative step }
while $A_h \neq \emptyset \vee B_h \neq \emptyset$ do
   if $v \in D_h$ exists such that $GTAD_{A_h,B_h}(v) > 0$ then
      choose $v_h \in D_h$ with maximum $GTAD_{A_h,B_h}(v_h)$;
      $S_{h+1} := S_h \cup \{v_h\}$;
   else
      choose a set $X_h \subseteq A_h \cup B_h$, with $2 \leq |X_h| \leq 3$, such that $S_h \cup X_h$ is a partial GTDS;
      $S_{h+1} := S_h \cup X_h$;
   end if
   Compute $A_{h+1}$, $B_{h+1}$, and $D_{h+1}$ by definition;
   $h := h + 1$;
end while
```

Theorem 2. *Heuristic H1 works in at most* $n - |D_0| - 4$ *iterations and hence it runs in time* $O(n^5)$.

Proof. First, we bound the number of iterations of the external loop. By Remark 2, the sets A_h, B_h, D_h, and S_h form a partition of $V(G)$. In particular, at iteration $h = 0$, we have $|A_0| + |B_0| + |D_0| + |S_0| = n$. Since $|S_0| = 4$, it follows that $|A_0| + |B_0| = n - |D_0| - 4$.

Now consider an arbitrary iteration $h \geq 0$ such that $A_h \neq \emptyset$ or $B_h \neq \emptyset$. If a vertex $v_h \in D_h$ exists such that $GTAD_{A_h,B_h}(v_h) > 0$, then Heuristic H1 selects such a vertex and adds it to S_h. By the definition of $GTAD_{A_h,B_h}(v_h)$, at least one vertex from $A_h \cup B_h$ becomes dominated in G or in $\overline{G}$ and therefore leaves the set $A_h \cup B_h$. On the other hand, if $GTAD_{A_h,B_h}(v) = 0$ for every $v \in D_h$, then, by Lemma 10, there is a set $X_h \subseteq A_h \cup B_h$, $2 \leq |X_h| \leq 3$, such that $S_{h+1} := S_h \cup X_h$ is a partial GTDS. In this case, at least the vertices of X_h leave the set $A_h \cup B_h$. Therefore, in every iteration of Heuristic H1, the quantity $|A_h| + |B_h|$ decreases by at least one. Hence, the total number of iterations is at most $|A_0| + |B_0| = n - |D_0| - 4$.

In iteration $h = 0$, for every set $\{u, v, q, w\}$, the set $(N[u] \cup N[v] \cup N[q] \cup N[w]) \setminus (N[u] \cap N[v] \cap N[q] \cap N[w])$ is formed by summing up all the 1 entries in the rows corresponding to the vertices $\{u, v, q, w\}$ if the four entries are not all 1s. Since each row contains n entries and there are $\frac{n(n-1)(n-2)(n-3)}{24}$ combinations of vertices to verify, the calculation of this set for all possible subsets $\{u, v, q, w\}$ and the selection of the set with maximum cardinality takes time $O(n^5)$.

In each subsequent iteration $h \geq 1$, the global total active degree is calculated for every candidate vertex in the current set D_h by counting the 1 and 0 entries in the row corresponding to that vertex. Vertices corresponding to entries of 1 belong to A_h, and those corresponding to entries of 0 belong to B_h. Since each row contains n entries and there are fewer than n rows to verify, the calculation of the global total active degree for all elements in D_h and the selection of the maximum take a time of $O(n^2)$.

In the exceptional case where all vertices in D_h satisfy $GTAD_{A_h,B_h}(v) = 0$, the escape step described in Lemma 10 requires only the inspection of adjacencies among vertices of $A_h \cup B_h$, which can also be done in time $O(n^2)$.

Since the number of iterations is at most $n - |D_0| - 4 < n$, the total cost of these iterations is $O(n^3)$. Therefore, heuristic H1 runs in time $O(n^5)$. □

Theorem 3. *Heuristic H1 finds an optimal feasible solution if it halts at iteration h with* $|S_h| \leq 5$, $h = 0, 1$.

Proof. We need to show that $|S_h| = \gamma_t^g(G)$ whenever $|S_h| \leq 5$. Recall that $|S_h| \geq 4$; assume $|S_h| = 4$, i.e., $h = 0$ and $S_0 = \{u, v, q, w\}$. Since the heuristic halted with the set S_0, $A_0 = B_0 = \emptyset$ and hence S_0 is a GTDS (Lemma 8).

If now $|S_h| = 5$, i.e., $h = 1$, then by the construction, there is no set $\{u, v, q, w\}$ providing $D_0 = V(G) \setminus S_0$. Hence, $\gamma_t^g(G) \geq 5$. Additionally, since the algorithm has halted at iteration 1, we have $A_1 = B_1 = \emptyset$, and by Remark 2, $D_1 = V \setminus S_1$. Therefore, S_1 is a feasible solution. □

We summarize Propositions 5, 8, 10, and 11 in the next proposition.

Proposition 18. *Assume that G is a graph with* $diam(G) = 2$ *and satisfies one of the following conditions:*

- *It is triangle-free and* $\delta(G) \leq 3$;
- $\delta(G) \leq 2$;
- $\Delta(G) \leq 3$;

- *It is planar and triangle-free.*

In this case, $\gamma_t^g(G) \leq 5$. Moreover, if Heuristic H1 halts with a set S_h such that $|S_h| \leq 5$, then it finds an optimal solution by Theorem 3.

3.2. Heuristic H2

This heuristic initiates by selecting two vertices $\{u, v\}$ such that u is a vertex of degree $\Delta(G)$ and v is the neighbor of vertex u with the minimum degree. The heuristic then determines the vertex w with the largest active degree satisfying $\{u, v\} \not\subseteq N(w)$ (so w is a vertex with the largest number of neighbors that are not yet dominated by either u or v). The last vertex q with the smallest degree and with at least one neighbor and one non-neighbor in the set $\{u, v, w\}$ is determined. This completes the set S_0.

Although constructed in this way, set $S_0 = \{u, v, w, q\}$ exists, there are two cases when $D_0 = \emptyset$ (see the proof of Lemma 9). In these cases, we update the set S_0 by replacing two vertices of S_0 with two non adjacent vertices from B_0, resulting in a new set S_0 (see the formal description of Algorithm 2). As a result, the newly formed set $D_0 \neq \emptyset$.

Remark 3. *Let $h \geq 0$ and let $v \in D_h$. By definition, $GTAD_{A_h,B_h}(v) = |N_{A_h}[v] \cup \overline{N}_{B_h}[v]|$. Since $v \in D_h$, we have $v \notin A_h$ and $v \notin B_h$. Therefore, $v \notin N_{A_h}[v]$ and $v \notin \overline{N}_{B_h}[v]$, and hence $GTAD_{A_h,B_h}(v) = 0$ if and only if $N_{A_h}(v) = \emptyset$ and $\overline{N}_{B_h}(v) = \emptyset$. In other words, a vertex $v \in D_h$ has zero global total active degree precisely when it cannot reduce either A_h or B_h by itself.*

Corollary 3. *Let $h > 0$. Suppose that $A_h \neq \emptyset$ or $B_h \neq \emptyset$. If, for every vertex $v \in D_h$, $N_{A_h}(v) = \emptyset$ and $\overline{N}_{B_h}(v) = \emptyset$, then there is a set $X \subseteq A_h \cup B_h$ with $2 \leq |X| \leq 3$ such that $S_{h+1} := S_h \cup X$ is a partial GTDS.*

Proof. By Remark 3, the assumption is equivalent to $GTAD_{A_h,B_h}(v) = 0$ for every $v \in D_h$. Hence, the result follows directly from Lemma 10. □

Each following iteration $h > 0$ consists of up to two greedy selection steps. If $A_{h-1} \neq \emptyset$, we first determine a vertex $v_h \in D_{h-1}$ such that $|N_{A_{h-1}}(v_h)|$ is as large as possible, provided that $N(v_h) \cap A_{h-1} \neq \emptyset$, and we update $S_h := S_{h-1} \cup \{v_h\}$. This step is intended to reduce the set of vertices that are not yet dominated by S_{h-1}.

Next, if $B_h \neq \emptyset$, we determine a vertex $u_h \in D_h$ such that $|\overline{N}_{B_h}(u_h)|$ is as large as possible, provided that $\overline{N}(u_h) \cap B_h \neq \emptyset$, and we update $S_h := S_h \cup \{u_h\}$. This step is intended to reduce the set of vertices that are not yet globally dominated by the current partial solution.

If at some stage no such vertex exists in D_h, then by Corollary 3, a set $X \subseteq A_h \cup B_h$ with $2 \leq |X| \leq 3$ exists such that $S_{h+1} := S_h \cup X$ is again a partial GTDS. In this case, we perform this escape update and continue the procedure. At each step, the sets A_h, B_h, and D_h are updated correspondingly; see the formal description of Algorithm 2 below.

Algorithm 2 Heuristic H2.

```
Input: A graph $G = (V, E)$.
Output: A GTDS $S_h$.
$h := 0$;
$\{u, v\} :=$ $u$ is the vertex with the highest degree in $V(G)$ and $v$ is its neighbor
        with the lowest degree;
$w :=$ vertex in $V(G) \setminus \{u, v\}$ with the highest active degree such that $\{u, v\} \not\subseteq N(w)$;
$q :=$ vertex in $V(G) \setminus \{u, v, w\}$ with the lowest degree such that for all $x \in \{u, v, w, q\}$,
    $1 \leq |N_{\{u,v,w,q\}}(x)| < 3$;
$S_0 := \{u, v, q, w\}$;
$D_0 := ((N(u) \cup N(v) \cup N(q) \cup N(w)) \setminus (N(u) \cap N(v) \cap N(q) \cap N(w))) \setminus S_0$;
$B_0 := N(u) \cap N(v) \cap N(q) \cap N(w)$;
if $D_0 = \emptyset$ then
    $\{b_1, b_2\} :=$ two vertices in $B_0$ that are not adjacent, such that
            $|(\overline{N}(b_1) \cup \overline{N}(b_2)) \setminus (\overline{N}(b_1) \cap \overline{N}(b_2))|$ is maximized;
    if $(w, u) \notin E(G) \wedge (w, v) \notin E(G)$ then
        $S_0 := \{u, w, b_1, b_2\}$;
        $v := b_1$;
        $q := b_2$;
    else if $(w, u) \notin E(G) \wedge (w, v) \in E(G)$ then
        $S_0 := \{q, w, b_1, b_2\}$;
        $u := b_1$;
        $v := b_2$;
    end if
end if
$D_0 := ((N(u) \cup N(v) \cup N(q) \cup N(w)) \setminus (N(u) \cap N(v) \cap N(q) \cap N(w))) \setminus S_0$;
$B_0 := N(u) \cap N(v) \cap N(q) \cap N(w)$;
$A_0 := V(G) \setminus (N[u] \cup N[v] \cup N[q] \cup N[w])$;
    { iterative step }
while $A_h \neq \emptyset \vee B_h \neq \emptyset$ do
    if $x \in D_h$ exists such that $GTAD_{A_h,B_h}(x) > 0$ then
        if $A_h \neq \emptyset$ then
            $h := h + 1$;
            $v_h :=$ a vertex $v \in D_{h-1}$ such that $|N_{A_{h-1}}[v]|$ be maximum;
            $S_h := S_{h-1} \cup \{v\}$;
            Compute $A_h$, $B_h$, and $D_h$ by definition;
        end if
        if $B_h \neq \emptyset$ then
            $h := h + 1$;
            $u_h :=$ a vertex $u \in D_{h-1}$ such that $|\overline{N}_{B_{h-1}}[u]|$ be maximum;
            $S_h := S_{h-1} \cup \{u\}$;
            Compute $A_h$, $B_h$, and $D_h$ by definition;
        end if
    else
        choose a set $X_h \subseteq A_h \cup B_h$, with $2 \leq |X_h| \leq 3$, such that $S_h \cup X_h$ is a partial GTDS;
        $S_{h+1} := S_h \cup X_h$;
        Compute $A_{h+1}$, $B_{h+1}$, and $D_{h+1}$ by definition;
        $h := h + 1$;
    end if
end while
```

Theorem 4. *Heuristic H2 works in at most* $n - |D_0| - 4$ *iterations and hence it runs in time* $O(n^3)$.

Proof. First, we bound the total number of augmenting steps performed by the heuristic. By Remark 2, the sets A_h, B_h, D_h, and S_h form a partition of $V(G)$. In particular, at iteration $h = 0$, we have $|A_0| + |B_0| + |D_0| + |S_0| = n$. Since $|S_0| = 4$, it follows that $|A_0| + |B_0| = n - |D_0| - 4$.

At each augmenting step, the heuristic does one of the following:

- Selects a vertex from D_h that dominates at least one vertex of A_h or globally dominates at least one vertex of B_h, or
- Applies the escape rule from Corollary 3.

In the first case, at least one vertex is removed from $A_h \cup B_h$. In the second case, since $S_{h+1} := S_h \cup X$ is again a partial GTDS with $X \subseteq A_h \cup B_h$, at least one unresolved vertex is also removed from $A_h \cup B_h$. Therefore, every augmenting step decreases $|A_h| + |B_h|$ by at least one. Hence, the total number of augmenting steps is at most $|A_0| + |B_0| = n - |D_0| - 4 < n$.

Now we analyze the running time. Similar to the proof of Theorem 2, the construction of the initial set $S_0 = \{u, v, w, q\}$ takes time $O(n^2)$.

For each subsequent augmenting step, we inspect all vertices in D_h. For every vertex $x \in D_h$, we compute the quantities $|N_{A_h}(x)|$ and $|\overline{N}_{B_h}(x)|$ by scanning the corresponding row of the adjacency matrix. Since each row contains n entries and there are at most n vertices in D_h, this requires a time $O(n^2)$ per augmenting step.

If the greedy step is not possible, then by Corollary 3, we apply the escape rule. The selection of the corresponding set $X \subseteq A_h \cup B_h$ with $2 \leq |X| \leq 3$ can also be performed within $O(n^2)$ time by checking the adjacency relations among candidate vertices in $A_h \cup B_h$.

Since the total number of augmenting steps is at most $n - |D_0| - 4 < n$, the total running time of Heuristic H2 is $O(n^2) + O(n) \cdot O(n^2) = O(n^3)$. Therefore, Heuristic H2 runs in time $O(n^3)$. □

Proposition 19. *Heuristic H2 finds an optimal feasible solution if it halts at iteration* $h = 0$ *with* $|S_h| = 4$.

Proof. Recall that $\gamma_t^g(G) \geq 4$. Assume that the heuristic halts at iteration $h = 0$; that is, $S_0 = \{u, v, q, w\}$ and $A_0 = B_0 = \emptyset$. By Lemma 8, it follows that S_0 is a GTDS. Hence, $\gamma_t^g(G) \leq |S_0| = 4$. Since $\gamma_t^g(G) \geq 4$ as well, we conclude that $\gamma_t^g(G) = 4$, and therefore S_0 is an optimal solution. □

Unlike Heuristic H1, for $|S_h| = 5$, S_h is not necessarily an optimal solution. Figure 10 shows a graph with $\gamma_t^g(G) = 4$ and $|S_h| = 5$. At iteration $h = 0$, $S_0 = \{u, v, q, w\}$. Then $D_0 = \{v_1, 1, 2, 3, 6, 7, 8\}$, $A_0 = \{4, 5\}$, and $B_0 = \emptyset$. At iteration $h = 1$, the vertex v_1 is chosen. Then $S_1 = \{u, v, q, w, v_1\}$, and S_1 is a GTDS of cardinality 5.

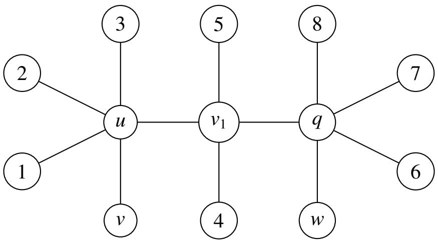


Figure 10. A graph with $\gamma_t^g(G) = 4$ and $|S_h| = 5$.

Next, consider a connected graph G of order $n > 2$, and let Q_i, $i = \overline{1,n}$, be a connected graph of order $k_i + 1 > 2$ containing the star graph S_{k_i} as a subgraph. Let, q_i also be the center of graph S_{k_i} (a vertex with the maximum degree k_i in that graph). $G' = G \circ_{q_i} Q_i$ is the graph composed of these $n + 1$ graphs, in which every vertex $v_i \in V(G)$ is associated with the vertex q_i of graph Q_i. Figure 11 illustrates the structure of such a compounded graph.

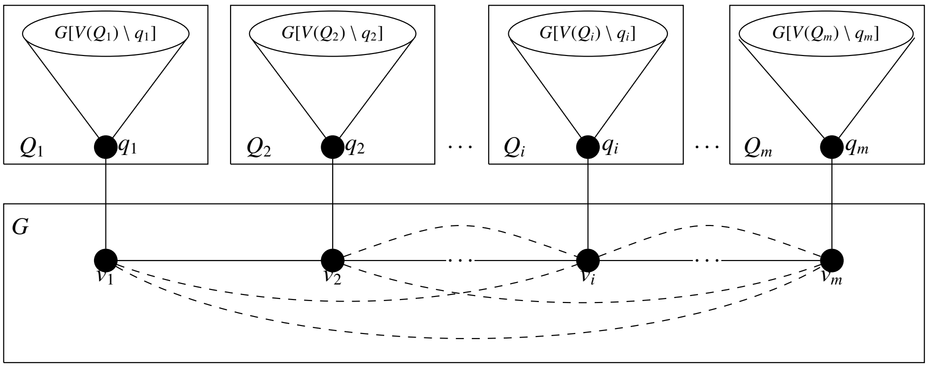


Figure 11. A graph $G' = G \circ_{q_i} Q_i$.

Proposition 20. $\gamma_t^g(G') = 2n$.

Proof. As it is easy to see, $n \geq 2$ implies that any two vertices of type q_i form a total dominating set of the graph $\overline{G'}$, whereas $\{q_i, v_i | i = \overline{1,n}\}$ forms a total dominating set of the graph G', since $\{q_i, v_i\}$ is a $\gamma_t(G[V(Q_i) \cup v_i])$-*set* and every vertex $v \in V(G)$ has one neighbor and one non-neighbor of type q_i; see Figure 11. Note that $\gamma_t(Q_i) = 2$, for $i = \overline{1,n}$. Hence, $\{q_i, v_i | i = \overline{1,n}\}$ is a $\gamma_t^g(G')$-*set* and $\gamma_t^g(G') = 2n$. □

Proposition 21. *Assume that for graph $G' = G \circ_{v_i} Q_i$, Q_i has order $k_i \geq 2$, for $i = \overline{1,n}$; let $k_1 \geq k_2 \geq \ldots \geq k_i \geq \ldots \geq k_{n-1} \geq k_n > \Delta(G)$, $\delta(Q_1) > N_{V(G)}(v_1)$; and let $\delta(Q_2) > N_{V(G)}(v_2)$, where $v_1, v_2 \in V(G)$. Then Heuristic H2 finds an optimal solution for graph G'.*

Proof. Clearly, $\{q_i, v_i\}$ is a total dominating set of the graph $G[V(Q_i) \cup v_i]$, $\gamma_t(Q_i) = 2$, and the $\gamma_t^g(G')$-*set* is $\cup_{i=1}^{n}\{q_i, v_i\}$ (see Proposition 20 and Figure 11). Since $k_1 \geq k_2 \geq \ldots \geq k_i \geq \ldots \geq k_{n-1} \geq k_n > \Delta(G)$, $\delta(Q_1) > N_{V(G)}(v_1)$, and $\delta(Q_2) > N_{V(G)}(v_2)$, Heuristic H2 will select the vertices $\{q_1, v_1, q_2, v_2\}$ in iteration $h = 0$, forming the set S_0. This set is a total dominating set of the graph $\overline{G'}$. Note that in every odd iteration $h \geq 1$, the only vertices in the set D_{h-1} that satisfy $N_{V \setminus N[S_{h-1}]}[v] \neq \emptyset$ and $N_{S_{h-1}}(v) \neq \emptyset$ are of type v_i. When h is even, there is a vertex of type q_i in the set D_{h-1} that has the largest number of undominated neighbors in G'. In every following iteration $h = 1, \ldots, 2n - 4$, Heuristic H2 adds one vertex of type v_i or q_i to the current solution S_h and returns a GTDS $S_{2n-4} = \{q_i, v_i | i = \overline{1,n}\}$, which is optimal by Proposition 20. □

3.3. Heuristic H3

Our third heuristic combines iteration 0 of Heuristic H2 with the iterative part of Heuristic H1. In the initial iteration (iteration 0), the vertices u, v, q, and w are defined as in Heuristic H2 (hence, note that Heuristic H3 also satisfies the optimality condition from Proposition 19). For each subsequent iteration $h > 0$, a vertex v_h is selected as in Heuristic H1 (see the formal description of Algorithm 3).

Algorithm 3 Heuristic H3.

```
Input: A graph $G = (V, E)$.
Output: A GTDS $S_h$.
$h := 0$;
$\{u, v\} :=$ $u$ is the vertex with the highest degree in $V(G)$ and $v$ is its neighbor
        with the lowest degree;
$w :=$ vertex in $V(G) \setminus \{u, v\}$ with the highest active degree such that $\{u, v\} \nsubseteq N(w)$;
$q :=$ vertex in $V(G) \setminus \{u, v, w\}$ with the lowest degree such that for all $x \in \{u, v, w, q\}$,
    $1 \leq |N_{\{u,v,w,q\}}(x)| < 3$;
$S_0 := \{u, v, q, w\}$;
$D_0 := ((N(u) \cup N(v) \cup N(q) \cup N(w)) \setminus (N(u) \cap N(v) \cap N(q) \cap N(w))) \setminus S_0$;
$B_0 := N(u) \cap N(v) \cap N(q) \cap N(w)$;
if $D_0 = \emptyset$ then
  $\{b_1, b_2\} :=$ two vertices in $B_0$ that are not adjacent, such that
        $|(\overline{N}(b_1) \cup \overline{N}(b_2)) \setminus (\overline{N}(b_1) \cap \overline{N}(b_2))|$ is maximized;
  if $(w, u) \notin E(G) \wedge (w, v) \notin E(G)$ then
    $S_0 := \{u, w, b_1, b_2\}$;
    $v := b_1$;
    $q := b_2$;
  else if $(w, u) \notin E(G) \wedge (w, v) \in E(G)$ then
    $S_0 := \{q, w, b_1, b_2\}$;
    $u := b_1$;
    $v := b_2$;
  end if
end if
$D_0 := ((N(u) \cup N(v) \cup N(q) \cup N(w)) \setminus (N(u) \cap N(v) \cap N(q) \cap N(w))) \setminus S_0$;
$B_0 := N(u) \cap N(v) \cap N(q) \cap N(w)$;
$A_0 := V(G) \setminus (N[u] \cup N[v] \cup N[q] \cup N[w])$;
    { iterative step }
while $A_h \neq \emptyset \vee B_h \neq \emptyset$ do
  $h := h + 1$;
  if there exists $v \in D_h$ such that $GTAD_{A_h,B_h}(v) > 0$ then
    choose $v_h \in D_h$ with maximum $GTAD_{A_h,B_h}(v_h)$;
    $S_{h+1} := S_h \cup \{v_h\}$;
  else
    choose a set $X_h \subseteq A_h \cup B_h$, with $2 \leq |X_h| \leq 3$, such that $S_h \cup X_h$ is a partial GTDS;
    $S_{h+1} := S_h \cup X_h$;
  end if
  Compute $A_{h+1}$, $B_{h+1}$, and $D_{h+1}$ by definition;
end while
```

Since Heuristic H3 combines the initialization procedure of Heuristic H2 with the iterative selection rule of Heuristic H1, its complexity analysis follows directly from the corresponding arguments established for Theorems 2 and 4. For completeness, we state the resulting bound below.

Theorem 5. *Heuristic H3 works in at most* $n - |D_0| - 4$ *iterations and hence it runs in time* $O(n^3)$.

Proof. Heuristic H3 uses the same initialization step as Heuristic H2, which runs in time $O(n^2)$, and the same iterative phase as Heuristic H1. By the argument used in the proof of Theorem 2, the iterative process performs at most $n - |D_0| - 4$ iterations, and each iteration requires $O(n^2)$ time. Therefore, the total running time is $O(n^3)$. □

3.4. Purification procedure

Purification procedures have turned out to be efficient tools for reducing the number of vertices in a non optimal dominating sets (see e.g., in [20]). In this section, we describe a purification procedure for our global total domination problem. We need the following definitions.

The *total private neighborhood* $tpn(v, S, G)$ of vertex $v \in S \subseteq V(G)$ is $tpn(v, S, G) = \{u \in V(G) \mid N(u) \cap S = \{v\}\}$, and a vertex in $tpn(v, S, G)$ is a *total private neighbor* of v. In contrast to $pn[v, S]$, the set $tpn(v, S, G)$ may also contain vertices of S. See Algorithm 4.

Algorithm 4 Purify.

```
Input: A graph $G = (V, E)$ and a GTDS $S_h$.
Output: A GTDS $S^*$.
$S^* = \{v_1, ..., v_p\}$; {vertices of $S_h$ in the order they were added by H1, H2, or H3.}
$h := p$;
    { iterative step }
while $h > 0$ do
    if $tpn(v_h, S^*, G) = \emptyset \wedge tpn(v_h, S^*, \overline{G}) = \emptyset$ then
        $S^* := S^* \setminus \{v_h\}$; {we purify vertex $v_h$}
    end if
    $h := h - 1$;
end while
```

The formed set S^* is a minimal GTDS for the graph G. Indeed, by the construction, the removal of any vertex from the set S^* would result in a solution that does not globally totally dominate the private neighbors of that vertex. It is easy to see that the procedure can be implemented in time $O(pn^2) = O(n^3)$ (this can be seen in a similar way to in Theorems 2 and 4).

4. Experimental results

In this section, we describe our computational experiments. Our ILP formulation and heuristics were implemented in C++ under the Windows 10 operating system (64-bit architecture) using Microsoft Visual Studio 2019 (version 16.11) on a personal computer equipped with an Intel Core i7-9750H (2.6 GHz) processor and 16 GB of DDR4 RAM. The ILP formulation was solved using the IBM ILOG CPLEX C++ libraries (`ilcplex/cplex.h` and `ilcplex/ilocplex.h`). Unless otherwise stated, the default CPLEX parameters were used in all experiments, except for the time limit imposed in the corresponding tests. We tested our algorithm on 1130 benchmark instances from [21–23].

The benchmark set consists of randomly generated simple graphs. For each instance, the order $n = |V(G)|$ and the size $m = |E(G)|$ were generated pseudo randomly within the corresponding ranges associated with each benchmark family. Then, starting from an empty graph on n vertices, edges were inserted uniformly at random between pairs of non adjacent vertices until the prescribed size m was reached. Hence, the generated instances follow a random graph construction of fixed order and size, similar to the classical Erdős–Rényi random graph model $G(n, m)$ [24]. The benchmark families BD0, BD1, . . . , BD7 were organized according to different ranges of graph order, size, and density. Here, the density of a graph G is defined as $dens(G) = \frac{2|E(G)|}{|V(G)|(|V(G)|-1)}$.

First, we summarize our results for the ILP formulation. As no benchmark instances for the our domination problem existed previously, we used optimal values obtained by CPLEX and the known upper and lower bounds to analyze the performance of our heuristics. In the following, we use ILP^L for the best lower bound obtained by the relaxation of our ILP formulation within the time limit of 20 minutes, and we use U for the objective value of the best feasible solution obtained by our heuristics. Among the tested benchmark instances, CPLEX was able to certify an optimal solution for 1050 instances within the imposed time limit. From this subset, we selected 30 instances at random as an illustrative sample and report the obtained results in Table 1. The complete computational results for the remaining instances are publicly available in the dataset reported in [21]. The entries in the table corresponding to $\gamma_t(G)$, $\gamma_t^g(G)$, and U can be compared with the lower and the upper bounds proposed in Section 2 and with the best solutions obtained by our heuristics. Table 1 includes the *gap* provided by CPLEX, which indicates how close the solution found by CPLEX is to an optimal one; it is the relative difference between the current solution value found by the solver and a best lower bound: $gap = \frac{|S|-L}{|S|}$, where S is the best feasible solution found by CPLEX and L is the best lower bound delivered by the solver. In the case of a positive gap, either the solver could not find an optimal solution or the available time or computational resources were insufficient to close the gap.

Table 1. Comparison of $\gamma_t(G)$ and $\gamma_t^g(G)$ values with heuristic solutions for 30 CPLEX-analyzed instances.

Graph	$\vert V(G)\vert$	$\vert E(G)\vert$	**ILP** $\gamma_t(G)$			**ILP** $\gamma_t^g(G)$			ILP^L	**U**	
			$\vert S\vert$	**Time(s)**	**gap**	$\vert S\vert$	**Time(s)**	**gap**		$\vert S\vert$	**Time(s)**
BD0_21	150	3725	6	0.433868	0	6	0.565838	0	6	7	0.001727
BD3_26	175	6858	7	0.171801	0	7	0.185212	0	7	7	0.0034096
BD1_31	200	4547	11	14.9226	0	11	15.0656	0	11	11	0.00301
BD0_41	250	10375	7	66.5806	0	7	97.6669	0	7	7	0.003799
BD3_46	275	16470	10	0.759432	0	10	0.446515	0	10	12	0.0054548
BD3_56	325	24896	7	1.19912	0	7	0.773005	0	7	9	0.0077495
BD1_61	350	18337	13	349.769	0	13	276.863	0	13	14	0.0089748
BD0_66	375	23375	7	567.937	0	7	690.26	0	7	8	0.0062221
BD3_76	425	41332	11	18.7068	0	11	17.6541	0	11	12	0.0113868
BD6_176	450	87361	5	0.034316	0	5	0.0968375	0	5	5	0.0040491
BD3_86	475	54183	8	30.3236	0	8	26.7823	0	8	9	0.0126022
BD3_101	550	73046	8	48.4127	0	8	41.2481	0	8	10	0.0176894
BD6_251	600	156934	5	0.0696031	0	5	0.151407	0	5	6	0.0074838
BD3_121	650	102571	8	126.061	0	8	217.006	0	8	10	0.0276011
BD6_288	674	171941	9	0.239355	0	9	0.253122	0	9	10	0.0163373
BD3_136	725	127996	8	158.403	0	8	171.538	0	8	10	0.0273989
BD6_326	750	214085	9	0.31592	0	9	0.343116	0	9	10	0.0344682
BD6_351	800	281227	5	0.0980406	0	5	0.236529	0	5	6	0.0209552
BD6_378	854	285299	6	0.316186	0	6	0.426589	0	6	7	0.035279
BD6_401	900	356878	6	0.564847	0	6	0.552009	0	6	6	0.0298821
BD6_431	960	398942	9	0.366656	0	9	0.550262	0	9	9	0.0512926
BD6_455	1008	398824	6	0.537497	0	6	0.682789	0	6	7	0.0604479
BD6_483	1064	500420	6	0.562107	0	6	0.54335	0	6	6	0.0397546
BD5_4	1200	1305	525	0.0395533	0	525	0.285345	0	525	531	0.700585
BD5_26	2300	2407	1009	0.0792524	0	1009	0.785574	0	1009	1014	5.6417
BD5_43	3150	3221	1398	0.117124	0	1398	1.55636	0	1398	1399	13.1076
BD5_65	4250	4299	1927	0.205809	0	1927	2.96783	0	1927	1929	37.4256
BD5_89	5450	5489	2472	0.334689	0	2472	4.86376	0	2472	2472	77.5741
BD5_107	6350	6491	2857	0.453513	0	2857	7.15439	0	2857	2863	124.236
BD5_120	7000	7038	3186	0.554199	0	3186	8.40799	0	3186	3186	178.857
BD5_130	7500	7535	3398	0.627284	0	3398	10.7075	0	3398	3398	219.461
BD5_140	8000	8126	3586	0.757664	0	3586	14.912	0	3586	3589	261.609
BD4_13	10450	27251021	12	1220.34	0.51802	11	1264.61	0.818182	1.999998	12	6.98681
BD4_18	10700	28523632	14	1248.79	0.372537	14	1450.51	0.372533	8.784538	15	8.49546

Figure 12 compares the best solutions produced by our heuristic algorithms and the optimal values obtained by CPLEX for the instances for which CPLEX proved optimality within the imposed time limit. Blue nodes represent optimal solutions, and red squares represent the best solutions found by our heuristics. At least one of our heuristics matched the optimal value in approximately 41% of these instances. For the remaining instances in this subset, the average absolute deviation from the optimum was 1.58 vertices.

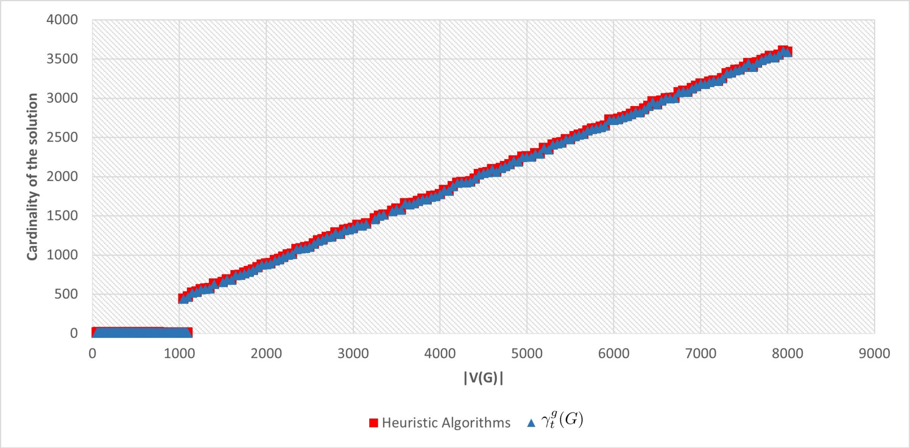
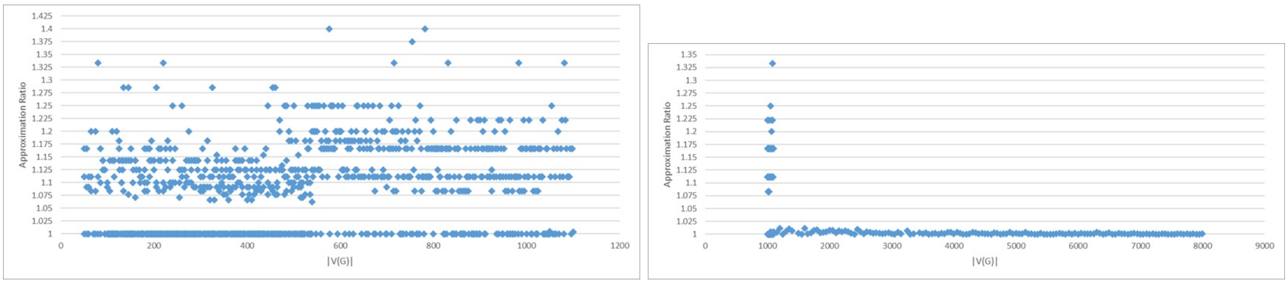


Figure 12. Approximation results vs. $\gamma_t^g(G)$.

Figure 13 shows the approximation ratio of our heuristics in those instances where the optimal solution was not found and the optimal objective value was obtained by CPLEX. For the moderately large instances with up to 1100 vertices, the minimum approximation ratio was 1.0042, the maximum was 1.4, and the average was 1.1436. We obtained even better results for the larger-sized instances: The minimum approximation ratio was 1.0002, the maximum was 1.0909, and the average was 1.0073.



(a) Instances with fewer than 1100 vertices

(b) Instances with more than 1000 vertices

Figure 13. Approximation ratio.

Figure 14 reflects a comparative analysis of the running times required for CPLEX and our heuristics to find the parameters $\gamma_t^g(G)$ and $\gamma_t(G)$. As we can observe, for the instances where the heuristic found optimal solutions (the $\gamma_t^g(G)$-sets), they were, on average, 12,374 times faster than CPLEX. Surprisingly, for 24.8% of the tested instances, CPLEX found $\gamma_t^g(G)$-sets faster than the corresponding $\gamma_t(G)$-sets. However, on average, $\gamma_t(G)$-sets were found 2.9 times faster than the corresponding $\gamma_t^g(G)$-sets.

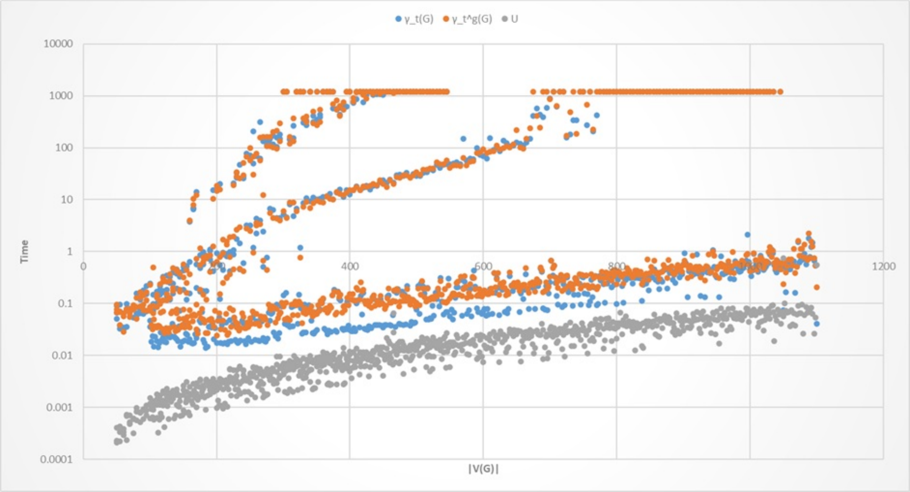


Figure 14. Time ILP formulation vs. heuristics (a logarithmic scale was applied).

In Table 2, we show the results obtained by our heuristics before and after the purification procedure, denoted by $|S|$ and $|S^*|$, respectively.

In Table 3, we summarize the results obtained by our heuristics before the application of the purification procedure. As we can observe, the three heuristics obtained the solutions with equal cardinality in 2.59% of the tested instances. For the remaining instances, Heuristics H1, H2, and H3 obtained the best solutions in 41.21%, 50.49%, and 5.71% of the tested instances, respectively.

As we can see from Table 2, when the purification procedure was applied to the solutions generated by Heuristic H2, the resulting purified solutions were better than those constructed by Heuristics H1 and H3. On average, the purification procedure reduced the size of the solutions from Heuristics H1, H2, and H3 by 9.40%, 18.22%, and 27.64%, respectively. Table 4 shows the number of instances where our heuristics found the best solutions after the purification.

Table 5 shows the number of the purified feasible solutions. In particular, Heuristics H1, H2, and H3 found minimal feasible solutions in 69.5%, 9.72%, and 11.78% of the tested instances, respectively.

Table 2. Results of Heuristics H1, H2, and H3 for the randomly generated graphs.

Graph	$\lvert V\rvert$	$\lvert E\rvert$	Heuristic H1		Purified			Heuristic H2		Purified			Heuristic H3		Purified			ILP^L
			Time(s)	$\lvert S\rvert$	Time(s)	$\lvert S*\rvert$	%	Time(s)	$\lvert S\rvert$	Time(s)	$\lvert S*\rvert$	%	Time(s)	$\lvert S\rvert$	Time(s)	$\lvert S*\rvert$	%	
BD0_91	500	31187	6088.13	11	0.0053	11	0.00	0.0154	11	0.0030	11	0.00	0.0128	11	0.0028	11	0.00	6.01
BD1_96	525	33228	6548.14	15	0.0056	14	6.67	0.0271	16	0.0039	15	6.25	0.0225	16	0.0073	15	6.25	9.03
BD0_100	545	49413	12458.8	9	0.0056	9	0.00	0.0240	11	0.0034	11	0.00	0.0150	10	0.0035	10	0.00	4.91
BD3_101	550	73046	—	—	—	—	—	0.0317	12	0.0046	10	16.67	0.0177	12	0.0051	11	8.33	8.00
BD3_111	600	84557	—	—	—	—	—	0.0516	15	0.0082	13	13.33	0.0236	15	0.0060	13	13.33	11.00
BD6_276	650	164146	—	—	—	—	—	0.0258	10	0.0058	7	30.00	0.0338	19	0.0143	7	63.16	6.00
BD3_131	700	116132	—	—	—	—	—	0.0611	15	0.0062	12	20.00	0.0316	15	0.0079	14	6.67	11.00
BD6_327	752	248140	—	—	—	—	—	0.0182	7	0.0072	6	14.29	0.0151	7	0.0053	6	14.29	6.00
BD3_151	800	152707	—	—	—	—	—	0.0838	15	0.0102	13	13.33	0.0413	15	0.0100	12	20.00	9.87
BD6_376	850	317920	—	—	—	—	—	0.0432	10	0.0071	6	40.00	0.0774	30	0.0151	6	80.00	6.00
BD3_171	900	198258	—	—	—	—	—	0.0876	12	0.0109	11	8.33	0.0455	12	0.0093	11	8.33	6.86
BD3_182	955	219280	—	—	—	—	—	0.1363	15	0.0160	13	13.33	0.0586	15	0.0095	14	6.67	9.18
BD3_191	1000	240857	—	—	—	—	—	0.1239	15	0.0134	14	6.67	0.0708	16	0.0131	14	12.50	9.08
BD5_10	1500	1598	—	—	—	—	—	1.3666	720	0.1242	653	9.31	3.2679	720	0.1354	653	9.31	652.00
BD5_20	2000	2128	—	—	—	—	—	3.7344	975	0.2305	887	9.03	7.9509	975	0.1947	887	9.03	880.00
BD5_41	3050	3064	—	—	—	—	—	11.6821	1492	0.3959	1381	7.44	50.5312	1492	0.3819	1381	7.44	1381.00
BD5_63	4150	4160	—	—	—	—	—	34.3193	2041	0.8292	1876	8.08	81.4496	2041	0.6999	1876	8.08	1876.00
BD5_82	5100	5139	—	—	—	—	—	63.0205	2501	1.1791	2294	8.28	152.9580	2501	1.0145	2294	8.28	2292.00
BD5_100	6000	6010	—	—	—	—	—	111.5620	2971	1.7376	2722	8.38	259.4440	2971	1.7651	2722	8.38	2722.00
BD5_110	6500	6586	—	—	—	—	—	135.6990	3168	1.8169	2925	7.67	327.7410	3168	1.5391	2925	7.67	2918.00
BD5_121	7050	7142	—	—	—	—	—	180.1830	3446	2.4496	3181	7.69	422.7810	3446	1.9864	3181	7.69	3180.00
BD5_131	7550	7557	—	—	—	—	—	221.3130	3738	2.5921	3446	7.81	527.7400	3738	2.2750	3446	7.81	3446.00
BD5_140	8000	8126	—	—	—	—	—	261.6090	3866	3.2614	3589	7.17	682.2990	3866	2.7646	3589	7.17	3586.00
BD4_3	9950	24703396	—	—	—	—	—	13.4991	15	0.9191	13	13.33	5.9653	15	0.9167	14	6.67	5.84
BD4_14	10500	27465482	—	—	—	—	—	15.1761	17	0.9967	15	11.76	8.1691	18	1.0769	16	11.11	8.78
BD4_24	11000	30148357	—	—	—	—	—	18.8644	18	1.2541	16	11.11	9.1008	18	1.1713	17	5.56	—
BD4_35	11550	33295796	—	—	—	—	—	18.0031	16	1.3649	14	12.50	8.5208	15	1.2841	14	6.67	—
BD4_44	12000	35943033	—	—	—	—	—	18.0301	15	1.2757	13	13.33	9.7955	16	1.3034	15	6.25	—
BD4_54	12500	38946982	—	—	—	—	—	22.9198	19	1.4388	17	10.53	13.1701	18	1.5308	17	5.56	—
BD4_65	13050	42455020	—	—	—	—	—	23.7640	18	1.4864	16	11.11	13.0467	19	1.5886	17	10.53	—
BD4_75	13550	45836296	—	—	—	—	—	24.6814	15	2.0193	13	13.33	12.4248	16	1.6505	15	6.25	—
BD4_86	14100	49635558	—	—	—	—	—	21.9406	14	1.8046	13	7.14	13.5461	16	1.9129	14	12.50	—
BD7_1	25000	249810756	—	—	—	—	—	38.8526	11	4.4324	8	27.27	115.8030	42	10.7639	9	78.57	—
BD7_2	25000	2499475	—	—	—	—	—	278.2380	328	2.9041	327	0.30	1006.1800	328	7.6087	327	0.30	—

Table 3. Comparative analysis of Heuristics H1–H3 before the purification.

	Instances	%
H1	462	41.21
H2	566	50.49
H3	64	5.71
All	29	2.59

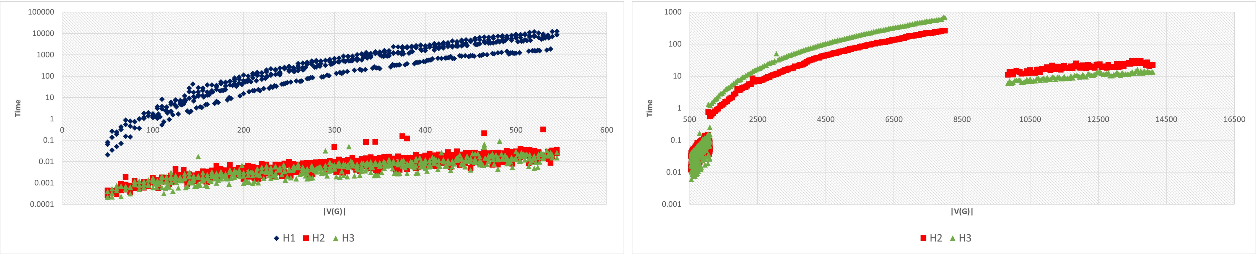
Table 4. Comparative analysis of Heuristics H1–H3 after the purification.

	Instances	%
H1	294	26.23
H2	617	55.04
H3	33	2.94
All	177	15.79

Table 5. Instances purified by the heuristics.

H1	**H2**	**H3**
157	1012	989

As for the execution times, for the largest benchmark instances with 25,000 vertices, Heuristics H2 and H3 halted in less than two minutes. Figure 15 shows the execution times for all the tested instances. On average, Heuristic H2 delivered a GTDS in less time than Heuristics H1 and H3. Heuristic H2 was 95,160 times faster than H1 and it was 1.18 times faster than Heuristic H3. Heuristics H2 and H3 are competitive in terms of time; Heuristic H3 was faster than Heuristic H2 in 57.62% of the cases.



(a) Instances with fewer than 550 vertices

(b) Instances with more than 500 vertices

Figure 15. Execution times for all the tested instances.

Code and data availability. The C++ source codes of the heuristics and the ILP implementation, together with the benchmark instances and the computational results generated in this study, are publicly available in the Mendeley Data repositories [21]. Additional large-scale benchmark instances are available at [22, 23].

5. Conclusions

We proposed the first direct solution methods for the classical GTDS problem. The proposed ILP model is a natural adaptation of the model given in [16] for the global dominating set problem. We also developed several theoretical results on the global total domination number $\gamma_t^g(G)$, including new bounds and structural properties for different classes of graphs, such as graphs with diameter 2 and triangle-free graphs. We used optimal values obtained by the CPLEX solver run for the ILP model to estimate the efficiency of our heuristics, using the earlier existing benchmark instances for the classical domination problem and also generated large-sized instances. CPLEX was able to solve the 1,050 benchmark instances with up to 10,700 vertices. The solutions for the largest benchmark instances with up to 25,000 vertices were obtained within less than 5 minutes. We also gave families of graphs where our heuristics guarantee an optimal solution. As to future work, we believe that our line of research can further be extended for more complex domination problems, applied to social networks, modeling biological processes, and supply chains.

Author contributions

Ernesto Parra Inza: Conceptualization, methodology, formal analysis, algorithm design, writing – original draft; José María Sigarreta Almira: Conceptualization, theoretical analysis, supervision, validation, writing – review and editing; Juan C. Hernández Gómez: Investigation, data curation, formal analysis, writing – review and editing; Nodari Vakhania: Theoretical analysis, validation, visualization, writing – review and editing. All authors have read and approved the final version of the manuscript for publication.

Use of Generative AI tools declaration

The authors confirm that no artificial intelligence (AI) tools were used in the preparation of this manuscript.

Conflict of interest

The authors declare no conflicts of interest.

References

1. W. Carballosa, J. Wisby, Total k-domination in cartesian product of complete graphs, *Discrete Appl. Math.*, **337** (2023), 25–41. https://doi.org/10.1016/j.dam.2023.04.008
2. A. Casado, J. Sánchez-Oro, A. Martínez-Gavara, Heuristics for the weighted total domination problem, *Top*, **33** (2025), 395–436. https://doi.org/10.1007/s11750-025-00695-1
3. I. Rios-Villamar, A. Cabrera-Martínez, G. Reyna Hernández, J. M. Sigarreta, Double total domination number of some graph operators, *Bull. Malays. Math. Sci. Soc.*, **49** (2026), 18. https://doi.org/10.1007/s40840-025-02016-y

4. M. H. Akhbari, C. Eslahchi, N. J. Rad, R. Hasni, Some remarks on global total domination in graphs, *Appl. Math. E-Notes*, **15** (2015), 22–28.
5. B. S. Panda, P. Goyal, Global total k-domination: approximation and hardness results, *Theoretical Comput. Sci.*, **850** (2021), 1–19. https://doi.org/10.1016/j.tcs.2020.10.027
6. B. S. Panda, P. Goyal, Hardness results of global total k-domination problem in graphs, *Discrete Appl. Math.*, **319** (2022), 223–238. https://doi.org/10.1016/j.dam.2021.02.018
7. S. Bermudo, A. C. Martínez, F. A. H. Mira, J. M. Sigarreta, On the global total k-domination number of graphs, *Discrete Appl. Math.*, **263** (2019), 42–50. https://doi.org/10.1016/j.dam.2018.05.025
8. P. Corcoran, A. Gagarin, Heuristics for k-domination models of facility location problems in street networks, *Comput. Oper. Res.*, **133** (2021), 105368. https://doi.org/10.1016/j.cor.2021.105368
9. P. J. Wan, K. M. Alzoubi, O. Frieder, Distributed construction of connected dominating set in wireless ad hoc networks, *Mobile Netw. Appl.*, **9** (2004), 141–149. https://doi.org/10.1023/B:MONE.0000013625.87793.13
10. J. Wu, B. Wu, I. Stojmenovic, Power-aware broadcasting and activity scheduling in ad hoc wireless networks using connected dominating sets, *Wirel. Commun. Mob. Comput.*, **3** (2003), 425–438. https://doi.org/10.1002/wcm.125
11. V. R. Kulli, B. Janakiram, The total global domination number of a graph, *Indian J. Pure Appl. Math.*, **27** (1996), 537–542.
12. T. W. Haynes, M. A. Henning, C. M. Mynhardt, Global total domination in graphs, *Bull. Inst. Comb. Appl.*, **24** (1998), 75–91.
13. J. D. Raj, V. S. Flower, A note on global total domination in graphs, *Bull. Pure Appl. Sci.-Math. Stat.*, **30** (2011), 63–70.
14. N. J. Rad, E. Sharifi, Bounds on global total domination in graphs, *Comput. Sci. J. Moldova*, **23** (2015), 3–10.
15. F. A. Hernández Mira, E. Parra Inza, J. M. Sigarreta Almira, N. Vakhania, Properties of the global total k-domination number, *Mathematics*, **9** (2021), 480. https://doi.org/10.3390/math9050480
16. E. Parra Inza, N. Vakhania, J. M. Sigarreta Almira, F. A. Hernández Mira, Algorithms for the global domination problem, *Comput. Oper. Res.*, **173** (2025), 106876. https://doi.org/10.1016/j.cor.2024.106876
17. E. Parra Inza, Random graph, *Mendeley Data*, Version 7, 2024. https://doi.org/10.17632/rr5bkj6dw5.7
18. W. J. Desormeaux, T. W. Haynes, M. A. Henning, A. Yeo, Total domination in graphs with diameter 2, *J. Graph Theory*, **75** (2014), 91–103. https://doi.org/10.1002/jgt.21725
19. W. Goddard, M. A. Henning, Domination in planar graphs with small diameter, *J. Graph Theory*, **40** (2002), 1–25. https://doi.org/10.1002/jgt.10027
20. E. Parra Inza, N. Vakhania, J. M. Sigarreta Almira, J. A. Hernández-Aguilar, Approximating a minimum dominating set by purification, *Algorithms*, **17** (2024), 258. https://doi.org/10.3390/a17060258

21. E. Parra Inza, Random graph, *Mendeley Data*, Version 9, 2025. https://doi.org/10.17632/rr5bkj6dw5.9

22. E. Parra Inza, Random graph BD4, *Mendeley Data*, Version 1, 2025. https://doi.org/10.17632/hyw9nckdsj.1

23. E. Parra Inza, Random graph BD5, *Mendeley Data*, Version 1, 2025. https://doi.org/10.17632/bhb7vpc2r8.1

24. P. Erdős, A. Rényi, On random graphs I, *Publ. Math.*, **6** (1959), 290–297.

AIMS Press