Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Total Positive Influence Domination in Graphs: Bounds, Exact Optimization, and Heuristics

Ernesto Parra Inza
Universidad Autónoma de Guerrero
Ismael Rios Villamar
Universidad Autónoma de Guerrero
Juan Carlos Hernández Gómez
Universidad Autónoma de Guerrero
José María Sigarreta Almira
14366@uagro.mx

Universidad Autónoma de Guerrero

Research Article

Keywords: graph theory; total positive influence domination; heuristic algorithms; integer linear programming; graph domination

Posted Date: November 3rd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11284976/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Total Positive Influence Domination in Graphs: Bounds, Exact Optimization, and Heuristics

Ernesto Parra Inza[1], Ismael Rios Villamar[1], Juan Carlos Hernández Gómez[1], José María Sigarreta Almira[1*]

[1]Facultad de Matemáticas, Universidad Autónoma de Guerrero (UAGro), Acapulco de Juárez, Guerrero, México.

*Corresponding author(s). E-mail(s): 14366@uagro.mx;
Contributing authors: 18446932@uagro.mx; 18305783@uagro.mx; jcarloshg@gmail.com;

Abstract

We study the total positive influence domination problem (TPIDS), in which every vertex must have at least $\lceil d(v)/2 \rceil$ neighbors in the selected set. We establish lower and upper bounds in terms of classical graph parameters and introduce a new lower bound based on the degree sequence, which can be computed in linear time once the degree sequence is available. We characterize minimal total positive influence dominating sets through private influence vertices and develop a purification procedure that transforms feasible solutions into minimal ones while preserving feasibility. We also formulate the optimization problem as a binary integer linear program and propose a constructive greedy heuristic based on support-vertex initialization and a maximum-total-active-degree selection rule. Both the heuristic and the purification procedure admit adjacency-list implementations running in $O(n+m)$ time. We prove an instance-dependent harmonic approximation guarantee $H(\kappa(G))$ for the heuristic, where $\kappa(G) \leq \Delta(G)$; thus, $H(\kappa(G)) \leq H(\Delta(G))$, with a possible improvement of at most $\ln(\Delta(G)/\kappa(G))$ in the approximation factor. Computational experiments on 1196 benchmark instances show that the heuristic produces feasible solutions for all instances and that the three purification strategies produce feasible minimal TPIDS, consistently with the corresponding theoretical guarantees. Purification strictly reduces the solution cardinality in 89.38% of the instances. On the 76 instances with certified optimality, purification reduces the average absolute error from 3.20 to 2.37 vertices and the average relative error from 4.80% to 3.55%.

Keywords: graph theory; total positive influence domination; heuristic algorithms; integer linear programming; graph domination.

1 Introduction

Domination theory is a central area of graph theory with numerous applications in communication networks, facility location, monitoring systems, biological networks, and social networks. In its classical form, a dominating set in a graph $G = (V, E)$ is a subset $D \subseteq V$ such that every vertex in $V \setminus D$ has at least one neighbor in D. A related concept is total domination, where every vertex of the graph must be adjacent to a vertex in the dominating set. These and many other domination parameters have been extensively studied over the last decades; see, for example, [1, 2].

The increasing importance of social and communication networks has motivated the development of domination models that incorporate influence and collective behavior. In this context, Wang et al. [3] introduced the concept of a *positive influence dominating set*. In this model, a vertex is considered influenced only if at least half of its neighbors belong to a selected set. This majority-based criterion captures situations in which an individual adopts a behavior, opinion, or decision only when supported by a sufficiently large fraction of its local social environment.

Subsequent research focused mainly on the computational aspects of the problem. Wang et al. [4] studied the minimum positive influence domination problem, proving APX-hardness and proposing a greedy approximation algorithm with ratio $H(\Delta)$. Dinh et al. [5] presented a more comprehensive theoretical study by distinguishing between *positive influence domination* (PIDS) and *total positive influence domination* (TPIDS). In the latter model, the influence condition is required to hold for every vertex of the graph, including those belonging to the selected set. They proved strong hardness and inapproximability results, studied approximation algorithms, and obtained exact algorithms for particular graph classes. From an algorithmic-complexity perspective, Wang et al. proposed a greedy $H(\Delta)$-approximation algorithm whose running time has been reported as $O(n^3)$ [4, 6]. Dinh et al. subsequently established an $H(\Delta)$-approximation for TPIDS with running time $O((n+m)\log\log n)$ through a reduction to constrained multiset multicover [5]. In contrast, the constructive heuristic developed here admits an explicit adjacency-list implementation in $O(n+m)$ time, while achieving the instance-dependent guarantee $H(\kappa(G)) \leq H(\Delta(G))$. Thus, the proposed method improves the general running-time bound at the algorithmic level while preserving the harmonic approximation framework.

It is worth noting that the terminology used in the literature is not completely uniform. In particular, the condition defining a TPIDS has also appeared in the literature on local majorities and monopolies. Peleg [7] surveyed this line of work, and Kuziak, Peterin, and Yero [8] studied the corresponding open 0-monopoly formulation. In the latter terminology, the open 0-monopoly condition coincides with the feasibility condition considered here. The terminology used in earlier positive influence domination works is also not completely uniform: the model called positive influence domination

in some earlier works corresponds to what Dinh et al. [5] later distinguish as total positive influence domination.

Minimal positive influence dominating sets have also been considered from an algorithmic perspective. Wang et al. [9] proposed a self-stabilizing algorithm for finding a minimal positive influence dominating set in graphs without isolated vertices. In this paper, we study minimal TPIDS from a structural perspective by introducing private influence vertices and deriving a characterization that leads to a general purification procedure. Further developments have also considered total positive influence domination in weighted networks, extending the influence framework to settings in which neighboring vertices may contribute unequally; see Greetham et al. [10].

Due to the computational difficulty of these problems, considerable effort has been devoted to the design of heuristic and metaheuristic algorithms. Abu-Khzam and Lamaa [11] proposed efficient greedy procedures for positive influence domination in social networks, while Bouamama and Blum [6] developed improved constructive heuristics capable of producing high-quality solutions on large-scale instances. More recently, Sun et al. [12] proposed an efficient local search algorithm for the minimum positive influence dominating set problem, and Penedo et al. [13] developed a Basic Variable Neighborhood Search combining randomized greedy reconstruction with a local search strategy for positive influence domination. Related optimization techniques have also been considered in the broader context of domination-type problems arising in large networks. Nguyen et al. [14] studied domination on very large-scale networks, while Alvarez-Miranda and Sinnl [15] considered exact and heuristic approaches for weighted total domination. Parra Inza et al. [16] developed algorithms for the global domination problem. Despite these advances, the literature remains predominantly algorithmic.

In contrast with classical domination and total domination, for which numerous structural results, bounds, and extremal characterizations are known (see, for example, [1, 2]), the literature on total positive influence domination has placed considerable emphasis on computational complexity and approximation algorithms. For example, Dinh et al. [5] established strong inapproximability results for PIDS and TPIDS and derived logarithmic-factor approximation algorithms, while also studying several special graph classes. Nevertheless, compared with the broader domination literature, structural bounds and extremal characterizations for TPIDS remain less developed.

Motivated by this gap, this paper focuses primarily on the structural study of total positive influence domination. We develop new bounds and extremal characterizations, introduce a degree-sequence lower bound, and characterize minimal TPIDS through private influence vertices. We then complement these structural results with a linear-time purification procedure, a constructive greedy heuristic with an instance-dependent guarantee, and an exact binary integer linear programming formulation for computational evaluation.

The main contributions of this paper can be summarized as follows.

- We establish lower and upper bounds for the total positive influence domination number in terms of classical graph parameters, including the order, size, minimum degree, maximum degree, packing number, and degree sequence.

- We derive extremal characterizations for several of the bounds obtained in this paper, providing structural conditions for equality.
- We introduce a new degree-sequence lower bound for the total positive influence domination number and present an $O(n)$-time algorithm for its computation once the degree sequence is available.
- We characterize minimal total positive influence dominating sets in terms of private influence vertices and use this characterization to develop a purification procedure that transforms any feasible TPIDS into a minimal one without increasing its cardinality.
- We propose a constructive greedy heuristic for TPIDS and prove an instance-dependent approximation guarantee in terms of the residual coverage parameter $\kappa(G)$. This guarantee is no larger than $H(\Delta)$ and improves on it by an additive term of at most $\ln(\Delta/\kappa(G))$.
- We use a binary integer linear programming formulation of TPIDS as an exact optimization benchmark and combine it with the proposed heuristic and purification procedures to conduct a computational study on 1196 benchmark instances.

The remainder of the paper is organized as follows. Section 2 introduces the notation and preliminary results used throughout the paper. Section 3 establishes several lower and upper bounds for the total positive influence domination number and presents the degree-sequence lower bound and its computational procedure. Section 4 characterizes minimal TPIDS and presents the purification procedure. The integer linear programming formulation, the heuristic approach, and the approximation guarantee are presented in Section 5, while the computational results are discussed in Section 6. Finally, Section 7 concludes the paper and discusses directions for future research.

2 Notation and Preliminaries

In this section, we introduce the notation and basic concepts used throughout the paper. We consider only simple, finite, and undirected graphs. Let $G = (V(G), E(G))$ be a graph with vertex set $V(G)$ and edge set $E(G)$. The order of G is $n = |V(G)|$. For a vertex $v \in V(G)$, the *open neighborhood* of v is defined as $N_G(v) = \{u \in V(G) : uv \in E(G)\}$, and the *closed neighborhood* is $N_G[v] = N_G(v) \cup \{v\}$. The degree of a vertex v is $d(v) = |N_G(v)|$. When the graph is clear from the context, we omit the subscript G. The minimum and maximum degrees of G are denoted by $\delta(G)$ and $\Delta(G)$, respectively.

For a subset $S \subseteq V(G)$, the subgraph induced by S is denoted by $G[S]$. For a subset $D \subseteq V(G)$, we denote by $\overline{D} = V(G) \setminus D$ its complement. For a set $D \subseteq V(G)$ and a vertex $v \in V(G)$, we define $\delta_D(v) = |N(v) \cap D|$, that is, the number of neighbors of v that belong to D.

A set $P \subseteq V(G)$ is called a *packing* of G if $N[u] \cap N[v] = \emptyset$ for every pair of distinct vertices $u, v \in P$. The maximum cardinality of a packing in G is called the *packing number* of G and is denoted by $\rho(G)$. A vertex of degree one is called a *leaf*. A vertex adjacent to a leaf is called a *support vertex*. We denote by $\mathcal{L}(G)$ and $\mathcal{S}(G)$ the sets of leaves and support vertices of G, respectively.

The following concepts constitute the main objects of study in this paper.

Positive influence domination. Let G be a graph and let $D \subseteq V(G)$. A vertex $v \in V(G)$ is said to be *influenced* by D if $\delta_D(v) \geq \left\lceil \frac{d(v)}{2} \right\rceil$. The set D is called a *positive influence dominating set* (PIDS) of G if every vertex $v \in V(G) \setminus D$ is influenced by D. The minimum cardinality of a PIDS in G is called the *positive influence domination number* of G and is denoted by $\gamma^+(G)$.

Total positive influence domination. A set $D \subseteq V(G)$ is called a *total positive influence dominating set* (TPIDS) of G if every vertex $v \in V(G)$ is influenced by D, that is, $\delta_D(v) \geq \lceil d(v)/2 \rceil$ for all $v \in V(G)$.

The minimum cardinality of a TPIDS in G is called the *total positive influence domination number* of G and is denoted by $\gamma_t^+(G)$.

Remark 1 Every TPIDS of a graph G is also a PIDS of G. Consequently, $\gamma^+(G) \leq \gamma_t^+(G)$.

3 General bounds on total positive influence dominating number

A set D is a *total k-dominating set* of G if $\delta_D(v) \geq k$ for every $v \in V(G)$. Such a set exists if and only if $\delta(G) \geq k$, and the minimum cardinality of one is denoted by $\gamma_t^k(G)$.

Lemma 1 *For any nontrivial connected graph G with minimum degree δ and maximum degree Δ, $\gamma_t^+(G) \geq \gamma_t^{\lceil \delta/2 \rceil}(G)$.*

Proposition 2 *If G is a nontrivial connected δ-regular graph, then $\gamma_t^+(G) = \gamma_t^{\lceil \delta/2 \rceil}(G)$.*

Fernau et al. [17] proved the following result.

Lemma 3 *Let k be a positive integer. For any nontrivial connected graph G of order n and maximum degree Δ, $\gamma_k^t(G) \geq \left\lceil \frac{kn}{\Delta} \right\rceil$.*

As a consequence of Proposition 2 and Lemma 3, we obtain the following corollary.

Corollary 4 *If G is a nontrivial connected δ-regular graph of order n, then*

$$\gamma_t^+(G) \geq \begin{cases} \left\lceil \frac{(\delta+1)n}{2\delta} \right\rceil & \text{if } \delta \text{ is odd,} \\ \left\lceil \frac{n}{2} \right\rceil & \text{if } \delta \text{ is even.} \end{cases}$$

Kuziak, Peterin, and Yero [8] introduced open k-monopolies, which are closely related to total positive influence domination. In particular, an open 0-monopoly

requires every vertex to have at least half of its neighbors in the selected set, which, by integrality, is equivalent to requiring at least $\lceil d(v)/2 \rceil$ such neighbors. Thus, the open 0-monopoly condition coincides with the feasibility condition considered in TPIDS. Kuziak et al. also established degree-based bounds for open k-monopolies. In particular, their upper bound for $k = 0$ gives $M_0(G) \leq n - \lfloor \delta/2 \rfloor$. We include the corresponding upper bound here as part of the TPIDS framework, together with the lower bound and equality characterization stated below.

Theorem 5 *If G is a nontrivial connected graph of order $n \geq 3$ and minimum degree δ, then*

$$\left\lceil \frac{\delta}{2} \right\rceil + 1 \leq \gamma_t^+(G) \leq n - \left\lfloor \frac{\delta}{2} \right\rfloor.$$

Proof Let v be a vertex with $d(v) = \delta$. We first prove the upper bound. If $\delta = 1$, then the set $V(G) \setminus \{v\}$ is a TPIDS of G, and hence $\gamma_t^+(G) \leq n-1 \leq n - \left\lfloor \frac{\delta}{2} \right\rfloor$. Now, assume that $\delta \geq 2$. Let $Y \subseteq N_G(v)$ be a set such that $|Y| = \left\lfloor \frac{\delta}{2} \right\rfloor$ and let $S = V(G) \setminus Y$. If $u \in Y$, then $\delta_Y(u) \leq |Y| - 1 = \left\lfloor \frac{\delta}{2} \right\rfloor - 1 \leq \left\lfloor \frac{d(u)}{2} \right\rfloor - 1$. Therefore, $\delta_S(u) = d(u) - \delta_Y(u) \geq d(u) - \left\lfloor \frac{d(u)}{2} \right\rfloor + 1 > \left\lceil \frac{d(u)}{2} \right\rceil$. If $u \in S$, then $\delta_Y(u) \leq |Y| = \left\lfloor \frac{\delta}{2} \right\rfloor \leq \left\lfloor \frac{d(u)}{2} \right\rfloor$, and consequently $\delta_S(u) = d(u) - \delta_Y(u) \geq \left\lceil \frac{d(u)}{2} \right\rceil$. Thus, S is a TPIDS of G, and therefore $\gamma_t^+(G) \leq |S| = n - \left\lfloor \frac{\delta}{2} \right\rfloor$.

We now prove the lower bound. Let D be a $\gamma_t^+(G)$-set. Since D is a TPIDS and $d(v) = \delta$, there exists a vertex $v' \in N(v) \cap D$. Let $A = N(v) \cap D$ and $B = N(v') \cap D$. By the TPIDS property, $|A| \geq \left\lceil \frac{\delta}{2} \right\rceil$ and $|B| \geq \left\lceil \frac{d(v')}{2} \right\rceil \geq \left\lceil \frac{\delta}{2} \right\rceil$. Moreover, $v' \in A$ but $v' \notin B$, since a vertex is not adjacent to itself. Hence $|A \cup B| \geq \left\lceil \frac{\delta}{2} \right\rceil + 1$. Since $A \cup B \subseteq D$, it follows that $\gamma_t^+(G) = |D| \geq |A \cup B| \geq \left\lceil \frac{\delta}{2} \right\rceil + 1$. □

Theorem 6 *Let $t = \lfloor \delta/2 \rfloor + 1$. Then*

$$\gamma_t^+(G) = n - \left\lfloor \frac{\delta}{2} \right\rfloor$$

if and only if every set $X \subseteq V(G)$ with $|X| = t$ has a common neighbour $z \notin X$ with $d(z) \leq 2 \lfloor \delta/2 \rfloor + 1$.

Proof Since supersets of a TPIDS are TPIDS, $\gamma_t^+(G) < n - \lfloor \delta/2 \rfloor$ if and only if there is a set X with $|X| = t$ such that $V(G) \setminus X$ is a TPIDS. In turn, $V(G) \setminus X$ is a TPIDS if and only if $\delta_X(z) \leq \lfloor d(z)/2 \rfloor$ for every $z \in V(G)$.

Suppose that $\delta_X(z) > \lfloor d(z)/2 \rfloor$ for some z. Then $\delta_X(z) \geq \lfloor \delta/2 \rfloor + 1 = t = |X|$. Hence $X \subseteq N(z)$, so $z \notin X$, and $t > \lfloor d(z)/2 \rfloor$, that is, $d(z) \leq 2t - 1 = 2 \lfloor \delta/2 \rfloor + 1$.

Conversely, if $X \subseteq N(z)$ and $d(z) \leq 2t - 1$, then $\delta_X(z) = t > \lfloor d(z)/2 \rfloor$. Thus, the existence of a set X for which $V(G) \setminus X$ is not a TPIDS is equivalent to the existence of a set X of size t having a common neighbour $z \notin X$ with $d(z) \leq 2 \lfloor \delta/2 \rfloor + 1$. Taking the negation yields the stated characterization. □

Proposition 7 *The upper bound of Theorem 5 is attained by the complete graphs K_n ($n \geq 3$) and by the graphs $K_{2q} - M$, where $q \geq 2$ and M is a matching with $1 \leq |M| \leq q-1$. The lower bound $\lceil \delta/2 \rceil + 1$ is attained by K_n.*

Proof In K_n, a vertex of D needs $|D| - 1 \geq \lceil (n-1)/2 \rceil$ and a vertex outside D needs $|D| \geq \lceil (n-1)/2 \rceil$. Hence $\gamma_t^+(K_n) = \lceil (n-1)/2 \rceil + 1 = n - \lfloor (n-1)/2 \rfloor$.

Let $G = K_{2q} - M$. Then $\delta = 2q-2$, so the upper bound equals $q+1$. Suppose D is a TPIDS with $|D| = q$. Since $|M| \leq q-1$, some vertex w is not covered by M, so $d(w) = 2q-1$. The vertex w needs q neighbours in D, so $D \subseteq N(w)$ and $w \notin D$. Every $u \in D$ needs $\lceil d(u)/2 \rceil \leq \delta_D(u) \leq q-1$. This forces $d(u) = 2q-2$, so u is covered by M, and also $\delta_D(u) = q-1$, so u is adjacent to all of $D \setminus \{u\}$. Thus D contains exactly one endpoint of q distinct edges of M, which contradicts $|M| \leq q-1$. Hence $\gamma_t^+(G) = q+1$. □

Remark 2 The condition $|M| \leq q-1$ is necessary: for the cocktail-party graph $K_{2q} - qK_2$ one has $\gamma_t^+ = q < q+1$.

Theorem 8 *If G is a nontrivial connected graph of order $n \geq 3$ and minimum degree δ, then*

$$\rho(G) \left\lceil \frac{\delta}{2} \right\rceil \leq \gamma_t^+(G) \leq n - \rho(G).$$

Proof First we proceed to prove the upper bound. Let P be a $\rho(G)$-set such that $|P \cap \mathcal{S}(G)|$ is minimum. This condition implies that $P \cap \mathcal{S}(G) = \emptyset$. Indeed, if $s \in P \cap \mathcal{S}(G)$, let u be a leaf adjacent to s. Since P is a packing, $u \notin P$, and replacing s by u preserves the packing property because $N[u] \subseteq N[s]$. This produces a packing of the same cardinality with fewer support vertices, a contradiction. Consequently, $P \cap \mathcal{S}(G) = \emptyset$, and hence $S(G) \subseteq \overline{P}$. In addition, if $d(v) \geq 2$, then $\delta_{\overline{P}}(v) \geq d(v) - 1 \geq \lceil d(v)/2 \rceil$ because P is a packing set. If $d(v) = 1$, then v is a leaf and its unique neighbor is a support vertex belonging to $\overline{P}$. Hence, $\delta_{\overline{P}}(v) = 1 = \lceil d(v)/2 \rceil$. Therefore, $\overline{P}$ is a TPIDS of G. Thus, $\gamma_t^+(G) \leq |\overline{P}| = n - \rho(G)$.
Now we proceed to prove the lower bound. Let D be a $\gamma_t^+(G)$-set. By definition of $\rho(G)$-set and $\gamma_t^+(G)$-set,

$$\gamma_t^+(G) = |D| \geq \sum_{v \in P} |N(v) \cap D| \geq \sum_{v \in P} \left\lceil \frac{d(v)}{2} \right\rceil \geq |P| \left\lceil \frac{\delta}{2} \right\rceil = \rho(G) \left\lceil \frac{\delta}{2} \right\rceil,$$

which completes the proof. □

The bounds given in Theorem 8 are tight. For example, for $G = K_q \odot K_1$ ($q \geq 2$), which has order $2q$, we have $\rho(G) = q$, $\delta(G) = 1$, and $\gamma_t^+(G) = q = |V(G)| - \rho(G) = \rho(G) \left\lceil \frac{\delta(G)}{2} \right\rceil$.

Theorem 9 *If G is a nontrivial connected graph with order n and minimum degree δ, then*

$$\gamma_t^+(G) \geq \sqrt{n \left\lceil \frac{\delta}{2} \right\rceil + \left(\frac{\lceil \delta/2 \rceil - 1}{2} \right)^2} - \frac{\lceil \delta/2 \rceil - 1}{2}. \tag{1}$$

Moreover, equality holds if and only if there exists a $\gamma_t^+(G)$-set D, with $r = |D|$, such that

1. $G[D] \cong K_r$;
2. $\delta_{\overline{D}}(u) = r - 1$ *for every* $u \in D$;
3. $\delta_D(v) = \lceil \delta/2 \rceil$ *for every* $v \in \overline{D}$.

In particular, equality implies $n \left\lceil \frac{\delta}{2} \right\rceil = r \left(r + \left\lceil \frac{\delta}{2} \right\rceil - 1 \right)$.

Proof Let D be a $\gamma_t^+(G)$-set and put $r = |D|$ and $k = \lceil \delta/2 \rceil$. Since D is a TPIDS, every vertex $v \in \overline{D}$ satisfies $\delta_D(v) \geq \lceil d(v)/2 \rceil \geq k$. Hence

$$(n-r)k \leq \sum_{v \in \overline{D}} \delta_D(v) = E(D, \overline{D}) = \sum_{u \in D} \delta_{\overline{D}}(u) \leq \sum_{u \in D} \delta_D(u) \leq r(r-1). \tag{2}$$

The penultimate inequality follows from the TPIDS condition for vertices in D, while the last one follows from $\delta_D(u) \leq r - 1$ for every $u \in D$.

Consequently, $r^2 + (k-1)r - nk \geq 0$. Since $r > 0$, it follows that $r \geq \frac{-(k-1)+\sqrt{(k-1)^2+4nk}}{2} = \sqrt{nk + \left(\frac{k-1}{2}\right)^2} - \frac{k-1}{2}$. Substituting $k = \lceil \delta/2 \rceil$ gives the stated lower bound.

Now suppose that equality holds. Then equality must hold throughout the chain above. In particular, $\sum_{v \in \overline{D}} \delta_D(v) = (n-r)k$. Since every term satisfies $\delta_D(v) \geq k$, we obtain $\delta_D(v) = k$ for every $v \in \overline{D}$. Moreover, $\sum_{u \in D} \delta_{\overline{D}}(u) = \sum_{u \in D} \delta_D(u)$. Since $\delta_D(u) \geq \delta_{\overline{D}}(u)$ for every $u \in D$, this implies $\delta_{\overline{D}}(u) = \delta_D(u)$ for every $u \in D$. Finally, $\sum_{u \in D} \delta_D(u) = r(r-1)$, and since $\delta_D(u) \leq r - 1$ for every $u \in D$, we have $\delta_D(u) = r - 1$ for every $u \in D$. Thus $G[D] \cong K_r$, and consequently $\delta_{\overline{D}}(u) = r - 1$ for every $u \in D$. This proves the necessity of the three stated conditions.

Conversely, suppose that there exists a $\gamma_t^+(G)$-set D with $r = |D|$ satisfying conditions (1)–(3). Since $G[D] \cong K_r$, we have $E(D) = \frac{r(r-1)}{2}$. By condition (2), $E(D, \overline{D}) = \sum_{u \in D} \delta_{\overline{D}}(u) = r(r-1)$, while condition (3) gives $E(D, \overline{D}) = \sum_{v \in \overline{D}} \delta_D(v) = (n-r)k$. Hence $(n-r)k = r(r-1)$, and therefore $nk = r(r+k-1)$. Consequently, $r^2 + (k-1)r - nk = 0$, so $r = \sqrt{nk + \left(\frac{k-1}{2}\right)^2} - \frac{k-1}{2}$, and equality holds in the stated lower bound. □

We use Hoeffding's inequality [18]: if $X = \sum_{i=1}^N X_i$, where the X_i are independent random variables with $a_i \leq X_i \leq b_i$, and $s \geq 0$, then

$$\Pr[X \geq \mathbb{E}[X] + s] \leq \exp\left(-\frac{2s^2}{\sum_{i=1}^N (b_i - a_i)^2} \right),$$

and the same bound holds for $\Pr[X \leq \mathbb{E}[X] - s]$. In particular, if $X \sim \mathrm{Bin}(d, p)$, then $\Pr[X \leq pd - s] \leq e^{-2s^2/d}$. The probabilistic construction used below is related to the methods of [19, 20].

Lemma 10 *Let* G *be a graph of order* n *without isolated vertices and let* $0 < \varepsilon \leq 1/2$. *Then*

$$\gamma_t^+(G) \leq \left(\frac{1}{2} + \varepsilon \right) n + \sum_{v \in V(G)} \left\lceil \frac{d(v)}{2} \right\rceil e^{-2\varepsilon^2 d(v)}.$$

Proof Let $p = \frac{1}{2} + \varepsilon$ and let D_0 be a random set containing each vertex independently with probability p. For each v, put $X_v = |N(v) \cap D_0|$, $t(v) = \left\lceil \frac{d(v)}{2} \right\rceil$, $Y_v = \max\{0, t(v) - X_v\}$. Adding, for each v, any Y_v vertices of $N(v) \setminus D_0$ produces a TPIDS D with $|D| \le |D_0| + \sum_{v \in V(G)} Y_v$. Since $t(v) - 1 \le \frac{d(v)-1}{2} < pd(v) - \varepsilon d(v)$, Hoeffding's inequality gives $\mathbb{E}[Y_v] \le t(v) \Pr[X_v \le t(v) - 1] \le t(v) e^{-2\varepsilon^2 d(v)}$. Taking expectations and using $\mathbb{E}[|D_0|] = pn$ yields

$$\mathbb{E}[|D|] \le \left(\frac{1}{2} + \varepsilon\right) n + \sum_{v \in V(G)} \left\lceil \frac{d(v)}{2} \right\rceil e^{-2\varepsilon^2 d(v)}.$$

Hence, some choice of D_0 produces a TPIDS satisfying the claimed bound. □

Theorem 11 *Let G be a graph of order n and minimum degree $\delta \ge 15$. Then*

$$\gamma_t^+(G) \le n \left(\frac{1}{2} + \sqrt{\frac{\ln \delta}{\delta}} + \frac{1}{\delta} \right).$$

Proof Apply Lemma 10 with $\varepsilon = \sqrt{\ln \delta / \delta} \le 1/2$. The function $g(x) = \frac{x+1}{2} e^{-2\varepsilon^2 x}$ is non-increasing for $x + 1 \ge \delta/(2 \ln \delta)$, which holds for every $x \ge \delta$. Hence, every term of the sum in Lemma 10 is at most $g(\delta) = \frac{\delta+1}{2} \delta^{-2} \le \frac{1}{\delta}$. The result follows. □

Remark 3 The proof of Theorem 11 is valid for every $\delta \ge 9$, but the factor $\frac{1}{2} + \sqrt{\frac{\ln \delta}{\delta}} + \frac{1}{\delta}$ is smaller than 1 only for $\delta \ge 15$. The bound improves on the bound $n - \lfloor \delta/2 \rfloor$ of Theorem 5 if and only if $n \left(\frac{1}{2} - \sqrt{\frac{\ln \delta}{\delta}} - \frac{1}{\delta} \right) > \left\lfloor \frac{\delta}{2} \right\rfloor$.

Proposition 12 *For every graph G without isolated vertices,*

$$\gamma_t^+(G) \ge \frac{\delta}{\delta + \Delta} n.$$

Proof Let D be a $\gamma_t^+(G)$-set. Since D is a TPIDS, every vertex in $\overline{D}$ satisfies $\delta_D(v) \ge \left\lceil \frac{d(v)}{2} \right\rceil \ge \frac{\delta}{2}$. Therefore, $E(D, \overline{D}) = \sum_{v \in \overline{D}} \delta_D(v) \ge \frac{\delta}{2}(n - |D|)$. On the other hand, for every $u \in D$, the TPIDS condition gives $\delta_D(u) \ge \delta_{\overline{D}}(u)$. Hence $\delta_{\overline{D}}(u) \le \frac{d(u)}{2} \le \frac{\Delta}{2}$. Consequently, $E(D, \overline{D}) = \sum_{u \in D} \delta_{\overline{D}}(u) \le \frac{\Delta}{2}|D|$. Combining the two inequalities yields $\delta(n - |D|) \le \Delta |D|$, and therefore $|D| \ge \frac{\delta}{\delta+\Delta} n$. □

Corollary 13 (10') *If $\delta \ge 15$, then*

$$\frac{\delta}{\delta + \Delta} n \le \gamma_t^+(G) \le n \left(\frac{1}{2} + \sqrt{\frac{\ln \delta}{\delta}} + \frac{1}{\delta} \right).$$

In particular, $\gamma_t^+(G) = \left(\frac{1}{2} + o(1)\right) n$ when $\delta \to \infty$ and $\Delta/\delta \to 1$.

Proof The lower bound is Proposition 12, and the upper bound follows from Theorem 11. The asymptotic statement is immediate. □

We next derive a probabilistic lower bound for the total positive influence domination number of the binomial random graph $G(n,p)$. For $A \subseteq V(G)$, let $e(A)$ denote the number of edges of $G[A]$ and let $e(A,\overline{A})$ denote the number of edges between A and $\overline{A}$.

Lemma 14 *If D is a TPIDS of a graph G, then $e(D,\overline{D}) \geq 2e(\overline{D})$.*

Proof For every $v \in \overline{D}$, since D is a TPIDS, $\delta_D(v) \geq \left\lceil \frac{d(v)}{2} \right\rceil \geq \delta_{\overline{D}}(v)$. Summing over $v \in \overline{D}$ gives

$$e(D,\overline{D}) = \sum_{v\in\overline{D}} \delta_D(v) \geq \sum_{v\in\overline{D}} \delta_{\overline{D}}(v) = 2e(\overline{D}),$$

as required. □

Theorem 15 *Let $0 < p \leq 1$, $\omega > 0$, and let $G = G(n,p)$. Then, with probability at least $1 - e^{-\omega}$,*

$$\gamma_t^+(G) > \frac{n-1}{2} - \frac{1}{2p}\sqrt{2n\ln 2 + 2\omega}.$$

Proof Put $t_0 = \frac{1}{p}\sqrt{2n\ln 2 + 2\omega}$. Fix $D \subseteq V(G)$ with $|D| = k \leq (n-1-t_0)/2$, and let $t = n - 2k - 1 \geq t_0$. Consider $X_D = e(D,\overline{D}) - 2e(\overline{D})$. This is a sum of $k(n-k)$ independent indicator variables, one for each pair joining D and $\overline{D}$, and of $\binom{n-k}{2}$ independent variables taking the values 0 or -2, one for each pair inside $\overline{D}$. Thus, $\mathbb{E}[X_D] = p\,(k(n-k) - (n-k)(n-k-1)) = -p(n-k)t$. Moreover, the sum of the squared ranges is $k(n-k) + 4\binom{n-k}{2} \leq 2n(n-k)$. By Hoeffding's inequality with $s = p(n-k)t$, and using $n - k \geq n/2$, $\Pr[X_D \geq 0] \leq \exp\left(-\frac{p^2(n-k)t^2}{n}\right) \leq e^{-p^2t_0^2/2} = 2^{-n}e^{-\omega}$. By Lemma 14, if D is a TPIDS then $X_D \geq 0$. Since there are at most 2^n subsets D, the probability that G has a TPIDS of cardinality at most $(n-1-t_0)/2$ is at most $e^{-\omega}$. Therefore, with probability at least $1 - e^{-\omega}$, $\gamma_t^+(G) > \frac{n-1}{2} - \frac{1}{2p}\sqrt{2n\ln 2 + 2\omega}$. □

As usual, an event holds *asymptotically almost surely* (a.a.s.) if its probability tends to 1 as $n \to \infty$.

Corollary 16 *If $p = p(n)$ satisfies $p\sqrt{n} \to \infty$, then a.a.s.*

$$\gamma_t^+(G(n,p)) = \left(\frac{1}{2} + o(1)\right) n.$$

Proof Taking $\omega = \ln n$ in Theorem 15, a.a.s. $\gamma_t^+(G(n,p)) \geq \frac{n-1}{2} - O\left(\frac{\sqrt{n}}{p}\right) = \left(\frac{1}{2} - o(1)\right) n$. For the upper bound, the degree of each vertex is distributed as $\mathrm{Bin}(n-1,p)$, with mean

$\mu = (n-1)p$. By the Chernoff bound, $\Pr[d(v) \le \mu/2] \le e^{-\mu/8}$. Since $\mu = \sqrt{n}\,(p\sqrt{n})(1-1/n)$ grows faster than $16 \ln n$, the union bound gives $\Pr[\delta \le \mu/2] \le n e^{-\mu/8} \to 0$. Thus, a.a.s. $\delta \ge \mu/2 \to \infty$, and Theorem 11 yields $\gamma_t^+(G(n,p)) \le \left(\frac{1}{2} + o(1)\right) n$. Combining both bounds proves the result. □

We next derive a lower bound for $\gamma_t^+(G)$ based on the degree sequence of the graph. Let $d(v_1) \le d(v_2) \le \cdots \le d(v_n)$ be the nondecreasing degree sequence of a graph G of order n, and write $d_i = d(v_i)$. We also let n_0 denote the number of vertices of odd degree. Since $\sum_{i=1}^{n} d_i = 2m$, we have

$$\sum_{i=1}^{n} \left\lfloor \frac{d_i}{2} \right\rfloor = m - \frac{n_0}{2}, \qquad \sum_{i=1}^{n} \left\lceil \frac{d_i}{2} \right\rceil = m + \frac{n_0}{2}.$$

We begin with the following lemma, which establishes a necessary condition on the degree of a vertex belonging to a TPIDS.

Lemma 17 *Let D be a TPIDS of a graph G, and let $v \in D$. Then $d(v) \le 2|D| - 2$.*

Proof Since D is a TPIDS and $v \in D$, we have $\delta_D(v) \ge \left\lceil \frac{d(v)}{2} \right\rceil$. On the other hand, $\delta_D(v) \le |D| - 1$, since v is not adjacent to itself. Therefore, $\left\lceil \frac{d(v)}{2} \right\rceil \le |D| - 1$, which implies $d(v) \le 2|D| - 2$. □

For every integer $k \ge 1$, let $\mu(k) = |\{i : d_i \le 2k-2\}|$. By Lemma 17, every TPIDS of cardinality k is contained in $\{v_1, \ldots, v_{\mu(k)}\}$. We now combine this observation with two necessary conditions obtained from the degree sequence.

We say that an integer k is *admissible* if $\mu(k) \ge k$, $k \ge \lceil d_n/2 \rceil$, and

$$\text{(C1)} \qquad \sum_{i=\mu(k)-k+1}^{\mu(k)} \left\lfloor \frac{d_i}{2} \right\rfloor \ge \sum_{i=1}^{n-k} \left\lceil \frac{d_i}{2} \right\rceil,$$

$$\text{(C2)} \qquad \sum_{i=\mu(k)-k+1}^{\mu(k)} d_i \ge m + \frac{n_0}{2}.$$

We define the degree-sequence lower bound by $\ell_{TP}(G) = \min\{k \in \{1, \ldots, n\} : k \text{ is admissible}\}$. We now establish the main result of this section.

Theorem 18 *Let G be a graph without isolated vertices. Then $\gamma_t^+(G) \ge \ell_{TP}(G)$.*

Proof Let D be a minimum TPIDS of G, and let $k = |D| = \gamma_t^+(G)$. By Lemma 17, every vertex of D satisfies $d(v) \le 2k-2$. Hence $D \subseteq \{v_1, \ldots, v_{\mu(k)}\}$, and therefore $\mu(k) \ge k$.

Moreover, every vertex of maximum degree has at least $\lceil d_n/2 \rceil$ neighbors in D. Thus $k \geq \left\lceil \frac{d_n}{2} \right\rceil$.

To establish (C1), since D is a TPIDS, every vertex in $\overline{D}$ has at least $\lceil d(v)/2 \rceil$ neighbors in D. Therefore,

$$\sum_{v \in \overline{D}} \left\lceil \frac{d(v)}{2} \right\rceil \leq \sum_{v \in \overline{D}} \delta_D(v) = \sum_{v \in D} \delta_{\overline{D}}(v) \leq \sum_{v \in D} \left\lfloor \frac{d(v)}{2} \right\rfloor.$$

Since $D \subseteq \{v_1, \ldots, v_{\mu(k)}\}$, the left-hand side is at least $\sum_{i=1}^{n-k} \left\lceil \frac{d_i}{2} \right\rceil$, while the right-hand side is at most $\sum_{i=\mu(k)-k+1}^{\mu(k)} \left\lfloor \frac{d_i}{2} \right\rfloor$. Hence (C1) holds.

For (C2), counting the pairs (v, x) such that $x \in D$ and v is adjacent to x, we obtain

$$\sum_{x \in D} d(x) = \sum_{v \in V} \delta_D(v) \geq \sum_{v \in V} \left\lceil \frac{d(v)}{2} \right\rceil = m + \frac{n_0}{2}.$$

Since D is contained in $\{v_1, \ldots, v_{\mu(k)}\}$, its degree sum is at most the sum of the k largest degrees among these vertices. Therefore, (C2) follows.

Thus k is admissible, and consequently $\gamma_t^+(G) = k \geq \ell_{TP}(G)$. □

The degree-sequence lower bound also dominates several of the previously established lower bounds used in this paper.

Proposition 19 *Let G be a graph of order n without isolated vertices, with minimum degree δ and maximum degree Δ, and let $k = \lceil \delta/2 \rceil$. Then*

$$\ell_{TP}(G) \geq \max \left\{ k+1, \left\lceil \frac{nk}{\lfloor \Delta/2 \rfloor + k} \right\rceil, \left\lceil -\frac{k-1}{2} + \sqrt{nk + \left(\frac{k-1}{2}\right)^2} \right\rceil \right\}.$$

In particular, $\ell_{TP}(G)$ is at least each of the lower bounds established in Theorem 5, Theorem 9, Proposition 12, and the regular-graph bound in Corollary 4; moreover, it is at least the lower bound obtained by applying Lemma 3 through the preceding total-domination lemma.

Proof Let t be an admissible integer. If a vertex belongs to a TPIDS of cardinality t, then it has at most $t-1$ neighbors in the selected set. Hence the TPIDS condition gives $\lceil d(v)/2 \rceil \leq t-1$, and therefore $d(v) \leq 2t-2$. Since $\mu(t) \geq t$, the first t degrees in the sorted degree sequence also satisfy $d_i \leq 2t-2$. In particular, $\delta \leq 2t-2$, and hence $t \geq k+1$.

By condition (C1), each of the t terms on its left-hand side satisfies $\left\lfloor \frac{d_i}{2} \right\rfloor \leq \left\lfloor \frac{\Delta}{2} \right\rfloor$. while each of the $n-t$ terms on its right-hand side satisfies $\left\lceil \frac{d_i}{2} \right\rceil \geq \left\lceil \frac{\delta}{2} \right\rceil = k$. Therefore, $t \left\lfloor \frac{\Delta}{2} \right\rfloor \geq (n-t)k$. which gives $t \geq \frac{nk}{\lfloor \Delta/2 \rfloor + k}$.

Similarly, condition (C1) together with $\lfloor d_i/2 \rfloor \leq t-1$ for the vertices that can belong to a TPIDS of cardinality t yields $t(t-1) \geq (n-t)k$. Equivalently, $t^2 + (k-1)t - nk \geq 0$. so $t \geq -\frac{k-1}{2} + \sqrt{nk + \left(\frac{k-1}{2}\right)^2}$. Since $\ell_{TP}(G)$ is an admissible integer and is the minimum admissible cardinality, the displayed lower bound follows.

Finally, $\lfloor \Delta/2 \rfloor + k \leq \Delta$ and $\frac{k}{\lfloor \Delta/2 \rfloor + k} \geq \frac{\delta}{\delta + \Delta}$, so the first fractional bound dominates the lower bound $\delta n/(\delta + \Delta)$ in Corollary 13. Applying the same argument with the appropriate

value of k in the regular case gives the bound stated in Corollary 4. Finally, the inequality $\gamma_k^t(G) \geq \lceil kn/\Delta \rceil$, together with the preceding lemma relating TPIDS to total k-domination, gives the remaining bound. □

The two conditions above provide complementary necessary restrictions on the cardinality of a TPIDS. Condition (C1) follows from balancing the influence requirements of vertices inside and outside the dominating set, while (C2) provides an additional restriction on the degree sum of the selected vertices.

The value of $\ell_{TP}(G)$ can be computed efficiently from the degree sequence. Since $\mu(k)$ is nondecreasing in k, and all the required sums can be evaluated using prefix sums, the following procedure computes the bound without explicitly scanning the degree sequence for every value of k.

Algorithm 1 Degree Sequence Lower Bound

Require: A graph $G = (V, E)$.
Ensure: The lower bound $\ell_{TP}(G)$.

1: Compute the degree sequence $d_1 \leq d_2 \leq \cdots \leq d_n$ using counting sort.
2: Compute the prefix sums

$$P[j] = \sum_{i=1}^{j} d_i, \qquad F[j] = \sum_{i=1}^{j} \left\lfloor \frac{d_i}{2} \right\rfloor, \qquad C[j] = \sum_{i=1}^{j} \left\lceil \frac{d_i}{2} \right\rceil,$$

with $P[0] = F[0] = C[0] = 0$.
3: Compute n_0, the number of vertices of odd degree.
4: $\mu \leftarrow 0$.
5: **for** $k = 1$ to n **do**
6: **while** $\mu < n$ and $d_{\mu+1} \leq 2k - 2$ **do**
7: $\mu \leftarrow \mu + 1$
8: **end while**
9: **if** $\mu \geq k$ and $k \geq \lceil d_n/2 \rceil$ **then**
10: **if** $F[\mu] - F[\mu - k] \geq C[n - k]$ **then**
11: **if** $2(P[\mu] - P[\mu - k]) \geq P[n] + n_0$ **then**
12: **return** k.
13: **end if**
14: **end if**
15: **end if**
16: **end for**
17: **return** n.

Theorem 20 *Algorithm 1 computes $\ell_{TP}(G)$ in $O(n)$ time from the degree sequence. Consequently, its running time is $O(n + m)$ when the graph is given by adjacency lists and $O(n^2)$ when the graph is given by an adjacency matrix.*

Proof Counting sort takes $O(n)$ time because the degrees belong to $\{0, \ldots, n-1\}$. Computing the three prefix-sum arrays and the number n_0 of odd-degree vertices also takes $O(n)$ time.

The value of μ is maintained incrementally. Since it only increases from 0 to at most n, the while-loop performs at most n iterations in total. All remaining operations in each iteration of the for-loop take constant time. Hence, after the degree sequence has been computed, the algorithm runs in $O(n)$ time.

Computing the degrees requires $O(n+m)$ time with adjacency lists and $O(n^2)$ time with an adjacency matrix. Therefore, the total running time is $O(n+m)$ with adjacency lists and $O(n^2)$ with an adjacency matrix. □

4 Minimal Total Positive Influence Dominating Sets

Let G be a graph and let $D \subseteq V(G)$ be a TPIDS. We say that D is a minimal TPIDS if no proper subset of D is a TPIDS of G.

Observe that every minimum TPIDS is minimal, although the converse does not necessarily hold.

The following concept plays a role analogous to that of a private neighbor in classical domination theory; see Hernández Mira et al. [21] and Parra Inza et al. [22].

Let D be a TPIDS of a graph G and let $v \in D$. A vertex $x \in V(G)$ is called a *private influence vertex* of v (with respect to D) if $v \in N(x)$ and $\delta_D(x) = \left\lceil \frac{d(x)}{2} \right\rceil$. The set of all private influence vertices of v is denoted by $PIV_D(v)$.

The following lemma gives a simple sufficient condition for a TPIDS to be non-minimal.

Lemma 21 *Let D be a TPIDS of a graph G. If there exists a vertex $v \in D$ such that $PIV_D(v) = \emptyset$, then $D \setminus \{v\}$ is a TPIDS of G.*

Proof Let $v \in D$ be such that $PIV_D(v) = \emptyset$. By definition of private influence vertex, for every vertex $x \in N(v)$ we have $\delta_D(x) > \left\lceil \frac{d(x)}{2} \right\rceil$. Therefore, $\delta_{D\setminus\{v\}}(x) = \delta_D(x) - 1 \geq \left\lceil \frac{d(x)}{2} \right\rceil$ for every $x \in N(v)$. On the other hand, if $y \notin N(v)$, then $\delta_{D\setminus\{v\}}(y) = \delta_D(y)$.

Since D is a TPIDS, every vertex of G satisfies the total positive influence condition with respect to D. The previous observations show that every vertex also satisfies the condition with respect to $D \setminus \{v\}$. Hence, $D \setminus \{v\}$ is a TPIDS of G. □

The next theorem characterizes minimal TPIDS in terms of private influence vertices.

Theorem 22 *Let D be a TPIDS of a graph G. Then D is minimal if and only if $PIV_D(v) \neq \emptyset$ for every vertex $v \in D$.*

Proof Suppose first that D is a minimal TPIDS and let $v \in D$. Since $D \setminus \{v\}$ is not a TPIDS, there exists a vertex $x \in V(G)$ such that $\delta_{D\setminus\{v\}}(x) < \left\lceil \frac{d(x)}{2} \right\rceil$. Since D is a TPIDS, $\delta_D(x) \geq \left\lceil \frac{d(x)}{2} \right\rceil$. The only difference between D and $D \setminus \{v\}$ is the removal of the vertex v. Hence,

$v \in N(x)$ and $\delta_D(x) = \left\lceil \frac{d(x)}{2} \right\rceil$. Therefore, $x \in PIV_D(v)$, and so $PIV_D(v) \neq \emptyset$. Conversely, suppose that $PIV_D(v) \neq \emptyset$ for every vertex $v \in D$. Let $v \in D$ and let $x \in PIV_D(v)$. Then $\delta_D(x) = \left\lceil \frac{d(x)}{2} \right\rceil$ and $v \in N(x)$. Consequently, $\delta_{D\setminus\{v\}}(x) = \delta_D(x) - 1 < \left\lceil \frac{d(x)}{2} \right\rceil$. Hence, $D \setminus \{v\}$ is not a TPIDS of G. Since this holds for every $v \in D$, the set D is minimal. □

Theorem 22 shows that every vertex of a minimal TPIDS is essential for maintaining the influence condition of at least one vertex in the graph.

Lemma 23 *Let D be a TPIDS of a graph G. Then there exists a minimal TPIDS $D^* \subseteq D$.*

Proof Starting with D, repeatedly remove a vertex v satisfying $PIV_D(v) = \emptyset$. By Lemma 21, each removal preserves the TPIDS property. Since one vertex is removed at each step and D is finite, the process terminates after finitely many iterations. Let D^* denote the final set obtained. At termination, $PIV_{D^*}(v) \neq \emptyset$ for every vertex $v \in D^*$. Therefore, by Theorem 22, the set D^* is a minimal TPIDS of G. Since the procedure only removes vertices from D, we have $D^* \subseteq D$. □

Lemma 23 shows that every TPIDS contains a minimal TPIDS. The following lemma establishes the monotonicity property that allows the purification process to be implemented in a single pass.

Lemma 24 *Let D be a TPIDS of a graph G, let $w \in D$ with $PIV_D(w) = \emptyset$, and let $D' = D \setminus \{w\}$. Then $PIV_D(v) \subseteq PIV_{D'}(v)$ for every $v \in D'$.*

Proof Let $x \in PIV_D(v)$. Then $\delta_D(x) = \left\lceil \frac{d(x)}{2} \right\rceil$ and $v \in N(x)$. We claim that $w \notin N(x)$. Indeed, if $w \in N(x)$, then $\delta_D(x) = \left\lceil \frac{d(x)}{2} \right\rceil$ and $w \in N(x)$ would imply $x \in PIV_D(w)$, contradicting $PIV_D(w) = \emptyset$.

Therefore, $w \notin N(x)$ and hence $\delta_{D'}(x) = \delta_D(x) = \left\lceil \frac{d(x)}{2} \right\rceil$. Since $v \in N(x)$, it follows that $x \in PIV_{D'}(v)$. Thus, $PIV_D(v) \subseteq PIV_{D'}(v)$. □

The previous lemma shows that once a vertex has a private influence vertex, that private influence vertex cannot be lost as a consequence of subsequent removals of redundant vertices. This property leads to the following one-pass purification procedure.

Theorem 25 *Algorithm 2 returns a minimal TPIDS contained in the input set D. Moreover, it runs in $O(n^2)$ time when G is represented by an adjacency matrix and in $O(n+m)$ time when G is represented by adjacency lists.*

Proof Let D_0 be the input TPIDS. The value $c(x)$ maintained by the algorithm is always equal to $\delta_D(x)$ for the current set D. Initially, this holds by definition. Whenever a vertex v is removed, the algorithm decreases $c(x)$ by one for every $x \in N(v)$, and therefore the

Algorithm 2 TPIDS Purification Procedure

Require: A TPIDS D of G
Ensure: A minimal TPIDS contained in D

```
D_0 ← D
Compute c(x) = δ_D(x) for all x ∈ V(G)
for each v ∈ D_0 in a fixed order do
    if v ∈ D and c(x) > ⌈d(x)/2⌉ for every x ∈ N(v) then
        D ← D \ {v}
        for each x ∈ N(v) do
            c(x) ← c(x) − 1
        end for
    end if
end for
return D
```

invariant is preserved. Consider a vertex $v \in D$ when it is examined. If $c(x) > \left\lceil \frac{d(x)}{2} \right\rceil$ for every $x \in N(v)$, then $PIV_D(v) = \emptyset$. By Lemma 21, removing v preserves the TPIDS property. Thus, every intermediate set produced by the algorithm is a TPIDS. If v is not removed when it is examined, then $PIV_D(v) \neq \emptyset$. By Lemma 24, every private influence vertex of v at that moment remains a private influence vertex of v after all subsequent removals. Consequently, every vertex that remains in the final set has at least one private influence vertex. By Theorem 22, the final set is a minimal TPIDS. Since the algorithm only removes vertices from D_0, the output is contained in the input set.

It remains to establish the running time. With an adjacency matrix, the initial values $c(x)$ can be computed in $O(n^2)$ time. Each vertex of D_0 is examined at most once, and checking the condition for a vertex v requires $O(n)$ time. Updating the values $c(x)$ after a removal also requires $O(n)$ time. Since at most n vertices are examined, the total running time is $O(n^2)$.

Now suppose that G is represented by adjacency lists. The initial values $c(x)$ can be computed in $O(n+m)$ time. For every $v \in D_0$, checking the condition requires $O(d(v))$ time, and if v is removed, updating the counters also requires $O(d(v))$ time. Since every vertex is examined at most once, the total cost of these operations is bounded by $2\sum_{v \in V(G)} d(v) = 4m$. Therefore, the total running time is $O(n+m)$. □

The purification procedure is not unique. Whenever several redundant vertices are available, different selection criteria may be adopted, leading to different minimal TPIDS. In particular, the order in which the vertices of D_0 are examined in Algorithm 2 may affect the resulting minimal TPIDS. In the computational experiments, we consider three deterministic ordering strategies: P_1 processes vertices in nondecreasing degree order, P_2 processes vertices in nonincreasing degree order, and P_3 processes vertices according to a fixed input vertex order $0, 1, \ldots, n-1$ used by the implementation. For P_1 and P_2, equal-degree vertices retain this original input order. We denote by $U_P = \min\{P_1, P_2, P_3\}$ the best purified solution.

5 ILP Formulation and Heuristic Algorithm

5.1 ILP Formulation

In this section, we present an integer linear programming (ILP) formulation for the total positive influence domination problem. Let $G = (V, E)$ be a graph with $V = \{1, 2, \ldots, n\}$. For each vertex $i \in V$, we define a binary variable:

$$x_i = \begin{cases} 1 & \text{if vertex } i \text{ belongs to the TPIDS,} \\ 0 & \text{otherwise.} \end{cases}$$

The objective is to minimize the number of selected vertices.

Model:

$$\min \quad \sum_{i \in V} x_i \tag{3}$$

$$\text{s.t.} \quad \sum_{j \in N(i)} x_j \geq \left\lceil \frac{d(i)}{2} \right\rceil, \qquad \forall i \in V \tag{4}$$

$$x_i \in \{0, 1\}, \qquad \forall i \in V \tag{5}$$

Explanation of the constraints. Constraint (4) enforces the total positive influence condition: every vertex $i \in V$ must have at least $\lceil d(i)/2 \rceil$ neighbors in the selected set, regardless of whether i itself belongs to the set or not. Constraint (5) ensures that each decision variable is binary.

Since $\lceil d(i)/2 \rceil$ is fixed for a given graph, the formulation is a standard binary integer linear program, and its feasible solutions are exactly the TPIDSs of G; hence its optimal value is $\gamma_t^+(G)$.

For graphs satisfying the hypotheses of Lemma 26, every TPIDS contains all support vertices. Hence, the general ILP can be reduced by fixing $x_s = 1$ for every $s \in S(G)$. This preserves the optimal value $\gamma_t^+(G)$ while reducing the number of free binary variables from n to $n - |S(G)|$. Moreover, the constraints corresponding to leaves become redundant and may therefore be omitted.

5.2 Heuristic Algorithm

The proposed heuristic constructs a feasible solution for the total positive influence domination problem by iteratively adding vertices to an initially selected set. We denote by IP_h the set of selected vertices after iteration h. The implementation resolves ties deterministically: when several candidate vertices have the same maximum TAD_h value, the first vertex in the corresponding bucket is selected. The buckets are initialized by scanning the vertices in increasing input-index order, so the tie-breaking rule is fully determined by the input vertex ordering. For a given set IP_h, define

$$B_h = \left\{ x \in V : |N(x) \cap IP_h| < \left\lceil \frac{d(x)}{2} \right\rceil \right\},$$

and

$$A_h = \left\{ x \in V : |N(x) \cap IP_h| \geq \left\lceil \frac{d(x)}{2} \right\rceil \right\}.$$

Thus, B_h contains the vertices that do not satisfy the total positive influence condition with respect to IP_h, whereas A_h contains those that satisfy it.

For graphs without isolated vertices, every TPIDS is also a dominating set. The support vertices provide a natural initialization for the heuristic, so we set $IP_0 = \mathcal{S}(G)$. The initial sets A_0 and B_0 are determined by evaluating the influence condition for every vertex of G. More precisely,

$$B_0 = \left\{ x \in V : |N(x) \cap IP_0| < \left\lceil \frac{d(x)}{2} \right\rceil \right\}, \qquad A_0 = V \setminus B_0.$$

The following lemma is useful for the initialization.

Lemma 26 *Let G be a graph without isolated vertices such that $G \ncong K_2$. Every TPIDS D of G contains all support vertices; that is, $S(G) \subseteq D$. In particular, if $B_0 = \emptyset$ and $IP_0 = S(G)$, then IP_0 is an optimal TPIDS of G.*

Proof Let $s \in S(G)$ and let u be a leaf adjacent to s. Since $d(u) = 1$, every TPIDS D must satisfy $|N(u) \cap D| \geq \left\lceil \frac{d(u)}{2} \right\rceil = 1$. Because u has only one neighbor, namely s, it follows that $s \in D$. Hence $S(G) \subseteq D$.

Now suppose that $B_0 = \emptyset$. By definition of B_0, every vertex of G satisfies the total positive influence condition with respect to $IP_0 = S(G)$. Therefore, IP_0 is a TPIDS. On the other hand, every TPIDS contains $S(G)$ by the first part of the lemma. Consequently, $|D| \geq |S(G)| = |IP_0|$ for every TPIDS D, and hence IP_0 is optimal. □

For $h \geq 1$ and $v \in V \setminus IP_{h-1}$, define the total active degree by $TAD_h(v) = |N(v) \cap B_{h-1}|$. Thus, $TAD_h(v)$ measures the number of currently unsatisfied vertices that are adjacent to v. At each iteration, the heuristic selects a vertex $v_h \in V \setminus IP_{h-1}$ with maximum $TAD_h(v)$ and sets $IP_h = IP_{h-1} \cup \{v_h\}$.

The sets B_h can be updated efficiently by maintaining, for every vertex $x \in V$, the counter $c(x) = |N(x) \cap IP_h|$. When a vertex v_h is added to IP_h, only the counters corresponding to vertices in $N(v_h)$ can change. Specifically, $c(x) \leftarrow c(x) + 1$ for every $x \in N(v_h)$. A vertex x leaves B_h as soon as $c(x) \geq \left\lceil \frac{d(x)}{2} \right\rceil$. Since the counters $c(x)$ are nondecreasing, every vertex leaves B_h at most once during the execution of the heuristic.

The values of TAD_h can also be maintained incrementally. Whenever a vertex x leaves B_h, only its neighbors can be affected. For every unselected neighbor $v \in N(x) \setminus IP_h$, we update $TAD(v) \leftarrow TAD(v) - 1$. Consequently, the heuristic does not need to recompute the values of TAD for all candidate vertices at every iteration.

To obtain a maximum-TAD vertex efficiently, we maintain the unselected vertices in buckets indexed by their current TAD values. Since $TAD(v)$ can only decrease during the execution, each decrement moves a vertex to a lower bucket. A pointer to the largest nonempty bucket is maintained to select a maximum-TAD vertex.

Remark 4 At each iteration exactly one new vertex is added to IP_h. Hence, the heuristic terminates after at most $n-|\mathcal{S}(G)|$ iterations. Moreover, every vertex leaves B_h at most once.

Proposition 27 *The constructive heuristic terminates with a TPIDS of G.*

Proof At every iteration, IP_h is obtained by adding one vertex to IP_{h-1}. The set B_h contains exactly the vertices that do not satisfy the total positive influence condition with respect to IP_h.

The algorithm terminates when $B_h = \emptyset$. At that point, every vertex of G satisfies $|N(x) \cap IP_h| \geq \left\lceil \frac{d(x)}{2} \right\rceil$. Therefore, IP_h is a TPIDS of G. □

Proposition 28 *The constructive heuristic can be implemented in $O(n^2)$ time when G is represented by an adjacency matrix and in $O(n+m)$ time when G is represented by adjacency lists.*

Proof First consider an adjacency matrix representation. The initial values $c(x) = |N(x) \cap IP_0|$ and the initial set B_0 can be computed in $O(n^2)$ time. The initial values of $TAD(v)$ can also be computed in $O(n^2)$ time.

Whenever a vertex v is added to IP, only its neighbors need to be checked, and a vertex leaves B at most once. With an adjacency matrix, each such update requires $O(n)$ time. Since at most n vertices are added to IP, the total update cost is $O(n^2)$.

The maximum TAD vertex can be maintained using buckets indexed by the possible values $0, \ldots, n-1$. Since the TAD values only decrease, the pointer to the largest nonempty bucket moves at most $O(n)$ positions during the entire execution. Thus, the total running time is $O(n^2)$.

Now consider an adjacency-list representation. The initial values $c(x)$ can be computed in $O(n+m)$ time by scanning the adjacency lists. The initial TAD values can likewise be computed in $O(n+m)$ time.

When a vertex v is added to IP, only vertices in $N(v)$ are processed. Since every vertex is added to IP at most once, the total cost of these updates is bounded by $\sum_{v \in V} d(v) = 2m$. Moreover, whenever a vertex x leaves B, only its neighbors need to have their TAD values decreased. Since every vertex leaves B at most once, the total cost of these updates is also bounded by $\sum_{x \in V} d(x) = 2m$. The bucket operations require constant time per insertion, deletion, or decrement, apart from the movement of the pointer to the largest nonempty bucket, which is $O(n)$ in total.

Therefore, the overall running time is $O(n+m)$. □

5.3 Approximation guarantee of the constructive heuristic

The constructive heuristic can be interpreted as a greedy algorithm for a residual constrained multiset multicover problem. This interpretation yields an instance-dependent approximation guarantee that can be smaller than the classical $H(\Delta)$ bound.

Algorithm 3 Constructive Heuristic for TPIDS

Require: A graph $G = (V, E)$.
Ensure: A TPIDS IP of G.
1: $IP \leftarrow \mathcal{S}(G)$.
2: Compute $c(x) = |N(x) \cap IP|$ for every $x \in V$.
3: $B \leftarrow \{x \in V : c(x) < \lceil d(x)/2 \rceil\}$.
4: **if** $B = \emptyset$ **then**
5: **return** IP
6: **end if**
7: Compute $TAD(v) = |N(v) \cap B|$ for every $v \in V \setminus IP$.
8: Initialize buckets according to the values $TAD(v)$.
9: **while** $B \neq \emptyset$ **do**
10: Select a vertex v with maximum $TAD(v)$.
11: Remove v from its bucket and add v to IP.
12: **for all** $x \in N(v)$ **do**
13: $c(x) \leftarrow c(x) + 1$.
14: **if** $x \in B$ and $c(x) \geq \lceil d(x)/2 \rceil$ **then**
15: Remove x from B.
16: **for all** $y \in N(x) \setminus IP$ **do**
17: $TAD(y) \leftarrow TAD(y) - 1$.
18: Remove y from its current bucket and insert it into the bucket corresponding to its new $TAD(y)$ value.
19: **end for**
20: **end if**
21: **end for**
22: **end while**
23: **return** IP.

Let $S = S(G)$ and let $B_0 = \left\{u \in V : |N(u) \cap S| < \left\lceil \frac{d(u)}{2} \right\rceil \right\}$. For each $u \in B_0$, define its residual demand by $r_u = \left\lceil \frac{d(u)}{2} \right\rceil - |N(u) \cap S|$. For every candidate vertex $v \in V \setminus S$, define the residual multiset $M_v = \{u : u \in N(v) \cap B_0\}$, where each vertex of $N(v) \cap B_0$ occurs with multiplicity one. Finally, define the residual coverage parameter $\kappa(G) = \max_{v \in V \setminus S} |N(v) \cap B_0|$.

Lemma 29 *Let G be a connected graph of order at least 3, and let $S = S(G)$. A set $X \subseteq V \setminus S$ satisfies that $S \cup X$ is a TPIDS of G if and only if, for every $u \in B_0$, $|N(u) \cap X| \geq r_u$. Consequently, the minimum cardinality of such a set X is $\gamma_t^+(G) - |S|$.*

Proof For every vertex $u \in B_0$, the definition of r_u gives $|N(u) \cap S| + r_u = \left\lceil \frac{d(u)}{2} \right\rceil$. Therefore, $S \cup X$ satisfies the total positive influence condition at u if and only if $|N(u) \cap S| + |N(u) \cap X| \geq \left\lceil \frac{d(u)}{2} \right\rceil$, which is equivalent to $|N(u) \cap X| \geq r_u$. Vertices outside B_0 already satisfy the total positive influence condition with respect to S and therefore remain satisfied after vertices are added to S.

Hence, $S \cup X$ is a TPIDS if and only if X satisfies all residual demands r_u. Moreover, by Lemma 26, every TPIDS of G contains all support vertices. Thus, an optimal TPIDS has the form $S \cup X^*$, where $X^* \subseteq V \setminus S$ is a minimum cardinality feasible residual solution. Consequently, $|X^*| = \gamma_t^+(G) - |S|$. □

Theorem 30 *Let G be a connected graph of order at least 3, and let U_H be the solution returned by the constructive heuristic in Section 5.2. If $B_0 = \emptyset$, then $U_H = S(G)$ and U_H is*

optimal. Otherwise, with $H(k) = \sum_{i=1}^{k} \frac{1}{i}$*, the constructive heuristic satisfies* $|U_H| - |S(G)| \leq H(\kappa(G))\big(\gamma_t^+(G) - |S(G)|\big)$*. In particular,* $|U_H| \leq H(\kappa(G))\,\gamma_t^+(G)$*. Moreover, if* U_P *denotes the best solution obtained after the purification phase, then* $|U_P| \leq |U_H| \leq H(\kappa(G))\,\gamma_t^+(G)$*.*

Proof If $B_0 = \emptyset$, Lemma 26 shows that $S = S(G)$ is an optimal TPIDS. Assume $B_0 \neq \emptyset$. Let $v_1, \ldots, v_T$ be the vertices selected by the heuristic, in this order, so that $U_H = S \cup \{v_1, \ldots, v_T\}$. Let $g_h = |N(v_h) \cap B_{h-1}| = TAD_h(v_h)$. Note that $g_h \geq 1$: every $x \in B_{h-1}$ satisfies $|N(x) \cap IP_{h-1}| < \lceil d(x)/2 \rceil \leq d(x)$, so x has an unselected neighbour.

Charging. At step h, every $u \in N(v_h) \cap B_{h-1}$ receives one unit at price $1/g_h$. The total price is exactly 1. A vertex $u \in B_0$ leaves B precisely after receiving r_u units; vertices outside B_0 receive no units.

Assignment to an optimal solution. By Lemma 29, there is $X^* \subseteq V \setminus S$ with $|X^*| = \gamma_t^+(G) - |S|$ and $|N(u) \cap X^*| \geq r_u$ for every $u \in B_0$. Fix $u \in B_0$, let $A_u = N(u) \cap X^*$, and let $P_u \subseteq A_u$ be the set of selected vertices of A_u that gave a unit to u. Assign each unit of u given by a vertex of P_u to that vertex. Assign the remaining $r_u - |P_u|$ units of u injectively to vertices of $A_u \setminus P_u$; this is possible because $|A_u \setminus P_u| \geq r_u - |P_u|$. Then each $w \in X^*$ receives at most one unit from each u.

Moreover, if a unit received by u at step h is assigned to w, then w was not selected before step h. Indeed, either $w = v_h$, or $w \in A_u \setminus P_u$. In the latter case, if w had been selected at a step $h' < h$, then $u \in B_{h-1} \subseteq B_{h'-1}$ and $w \in N(u)$, so w would have given a unit to u, that is, $w \in P_u$, a contradiction.

Bound for each $w \in X^*$. Let $u_1, \ldots, u_s$ be the vertices whose units are assigned to w, ordered by the steps $h_1 \leq \cdots \leq h_s$ at which those units were given. Then $s \leq |N(w) \cap B_0|$. Since the sets B_h are nested, $u_i, \ldots, u_s \in B_{h_i-1}$. Since w is an unselected candidate at step h_i, the greedy rule gives $g_{h_i} = TAD_{h_i}(v_{h_i}) \geq TAD_{h_i}(w) = |N(w) \cap B_{h_i-1}| \geq s - i + 1$. Hence the total price assigned to w is at most $\sum_{i=1}^{s} \frac{1}{s-i+1} = H(s) \leq H(\kappa^*)$, $\kappa^* = \max_{w \in X^*} |N(w) \cap B_0| \leq \kappa(G)$.

Conclusion. Every unit is assigned to exactly one vertex of X^*, so $|U_H| - |S| = T \leq H(\kappa^*)|X^*| \leq H(\kappa(G))\big(\gamma_t^+(G) - |S|\big)$. Since $H(\kappa(G)) \geq 1$, it follows that $|U_H| \leq H(\kappa(G))\gamma_t^+(G)$. Algorithm 2 only removes vertices during purification, so $|U_P| \leq |U_H|$. □ □

Corollary 31 *Under the assumptions of Theorem 30,* $\kappa(G) \leq \Delta(G)$*, and consequently*

$$|U_P| \leq |U_H| \leq H(\kappa(G))\,\gamma_t^+(G) \leq H(\Delta(G))\,\gamma_t^+(G) \leq (\ln \Delta(G) + 1)\,\gamma_t^+(G).$$

Remark 5 For TPIDS, Dinh et al. [5] established an $H(\Delta)$-approximation, with running time $O((n+m)\log\log n)$. For $\rho = 1/2$, their bound can be expressed as $(\ln \Delta + 1)\gamma_t^+(G)$. Our bound replaces Δ by the residual parameter $\kappa(G) \leq \Delta(G)$, while the proposed adjacency-list implementation runs in $O(n+m)$ time. The factor $H(\kappa(G))$ is an instance-dependent refinement of the

classical $H(\Delta(G))$ bound: since $\kappa(G) \leq \Delta(G)$, it is never larger, and it is strictly smaller whenever $\kappa(G) < \Delta(G)$. The corresponding improvement is additive and satisfies

$$H(\Delta(G)) - H(\kappa(G)) \leq \ln\left(\frac{\Delta(G)}{\kappa(G)}\right).$$

Thus, the refinement is instance-dependent rather than an asymptotic improvement in the logarithmic order. Moreover, the strict inequality $\kappa(G) < \Delta(G)$ admits the following structural characterization: it holds if and only if every vertex $v \in V \setminus S(G)$ with $d(v) = \Delta(G)$ has at least one neighbour in A_0.

Example. For $t \geq 3$, let G_t be obtained from the corona $K_t \odot K_1$, with support vertices $s_1, \ldots, s_t$ and leaves $\ell_1, \ldots, \ell_t$, by adding a vertex c adjacent to $s_1, \ldots, s_t$ and a pendant path cab. Then $S(G_t) = \{s_1, \ldots, s_t, a\}$ and $B_0 = \{a\}$. Indeed, each s_i has $t-1 \geq \lceil (t+1)/2 \rceil$ neighbours in $S(G_t)$, the vertex c has $t+1 \geq \lceil (t+1)/2 \rceil$ neighbours in $S(G_t)$, and every leaf is adjacent to a support vertex. Hence $\kappa(G_t) = 1$ and $\Delta(G_t) = t+1$. Therefore, $H(\kappa(G_t)) = H(1) = 1$, and Theorem 30 implies that the constructive heuristic is optimal on G_t. In contrast, the classical bound is $H(\Delta(G_t)) = H(t+1)$, which tends to infinity as $t \to \infty$. Thus, the ratio $H(\Delta(G_t))/H(\kappa(G_t)) = H(t+1)$ is unbounded.

6 Experimental Results

This section evaluates the proposed ILP formulation, constructive heuristic, purification procedures, and theoretical lower bounds. All algorithms were implemented in C++ under the Windows 10 operating system (64-bit architecture) using Microsoft Visual Studio 2019 with the v142 platform toolset and the C++17 language standard. The experiments were conducted on a computer equipped with an Intel Core i7-1065G7 processor and 8 GB of RAM. The ILP formulation was solved using IBM ILOG CPLEX Optimization Studio 22.1.1 through the C++ libraries `ilcplex/cplex.h` and `ilocplex.h`. CPLEX was configured with a time limit of 1200 seconds (20 minutes) per instance and with automatic parallelism (`Threads=0`); all other parameters were left at their default values. The deterministic tie-breaking rules of the constructive heuristic and the three purification orderings are described in Sections 5 and 4, respectively. The reported CPLEX times are measured externally using a wall-clock timer around the `solve()` call; therefore, they may slightly exceed the 1200-second CPLEX time limit because of termination and solver-call overhead.

The computational study is designed to complement the theoretical analysis by evaluating the practical behavior of the proposed constructive heuristic and purification procedures, their computational cost, and their relationship with the exact ILP formulation and the derived lower bounds. In particular, the experiments assess solution feasibility, solution cardinality, running time, and distance from certified optimal solutions or available lower bounds. The study is not designed as a comparative benchmark of heuristic methods, since the main contributions of this work are structural and theoretical and include a provable instance-dependent approximation guarantee and explicit complexity bounds. Comparisons with existing heuristics are meaningful only after aligning the precise PIDS/TPIDS problem definitions and implementation conditions.

6.1 Benchmark repository and experimental setting

The complete benchmark repository contains 1196 graph instances grouped into seven families, denoted BD0–BD6. The number of instances in each family is summarized in Table 1. The repository includes small and medium-sized graphs as well as large sparse and dense instances, with orders ranging from 50 to 14,100 vertices.

Table 1 Summary of the benchmark families used in the computational experiments.

Family	Instances	Order range	Observed density	Main characteristics
BD0	100	50–545	0.250–0.333	Medium-density random graphs
BD1	100	50–545	0.134–0.322	Random graphs with varying densities
BD2	106	1000–6450	0.242–0.332	Medium-density random graphs
BD3	200	50–1045	0.220–0.491	Dense random graphs
BD4	30	9850–14100	0.498–0.499	Large dense graphs
BD5	160	1050–9700	0.0002–0.0021	Extremely sparse graphs ($m \approx n$)
BD6	500	100–1098	0.543–0.885	Dense random graphs with increasing order
Total	**1196**	50–14100	–	

Here, the observed density is defined as $m/\binom{n}{2}$.

For each instance, the heuristic constructs a feasible TPIDS, denoted by U_H. We then apply the three purification strategies described in Section 4, obtaining solutions P_1, P_2, and P_3. As an implementation check, the feasibility of the heuristic solution and of each purified solution was explicitly verified for every instance by testing the TPIDS constraints.

For the exact approach, UB denotes the best feasible solution found by CPLEX, while LB denotes the best lower bound obtained during the branch-and-bound process. The relative optimality gap is computed as gap $= \frac{UB-LB}{UB}$. A zero gap certifies optimality. Percentages reported in the tables are rounded to two decimal places.

The benchmark instances used in this study are openly available in Mendeley Data through three complementary datasets [23–25]. The main repository is available as V10, while the BD4 and BD5 datasets contain additional instances that are distributed separately because of repository size constraints. The benchmark used here consists of the BD0–BD3 and BD6 families from V10 (1006 instances), together with the 30 BD4 instances and 160 BD5 instances from their respective V1 datasets, for a total of 1196 instances. Across the benchmark repository, the order and size of each instance are generated pseudo-randomly within the prescribed ranges. To guarantee connectivity, the first $n-1$ edges are generated sequentially: for each $j = 2, \ldots, n$, vertex j is

connected to a vertex $i < j$ chosen at random. These edges form a spanning tree. Additional edges are then selected at random from the remaining non-adjacent vertex pairs until the prescribed number m of edges is reached [23]. Thus, the reported density $m/\binom{n}{2}$ is an observed graph density and should not be interpreted as the parameter p of an Erdős–Renyi $G(n,p)$ model. In particular, Theorem 15 is stated for $G(n,p)$ and is not used as a lower bound for the benchmark instances. Since CPLEX was not run on every benchmark instance, results involving UB, LB, or certified optimality are reported only for the instances for which the corresponding CPLEX information is available. In particular, CPLEX was run on 900 of the 1196 instances, while the remaining 296 instances were evaluated using the heuristic, purification procedures, and theoretical lower bounds.

Table 2 reports the results for an illustrative sample of 30 instances selected from the complete benchmark repository. In addition to the computational quantities, the table reports the minimum and maximum degrees, δ and Δ, and the degree-sequence lower bound $\ell_{TP}(G)$. The latter is computed from the degree sequence according to Theorem 18. Here, T_H and T_{ILP} denote the running times of the heuristic and ILP solver, respectively, measured in seconds.

Table 2 Computational results and theoretical lower-bound information for a representative sample of benchmark instances.

Instance	n	m	dens.	δ	Δ	ℓ_{TP}	U_H	T_H (s)	U_P	UB	T_{ILP} (s)	gap (%)	LB
BD0-16	125	2583	0.3333	1	60	57	67	0.0005	67	63	1204.72	2.21	61.61
BD6-46	190	13081	0.7285	1	165	88	100	0.0015	99	96	1204.16	1.36	94.69
BD3-38	235	11740	0.4270	1	129	103	127	0.0022	124	119	1201.59	1.96	116.67
BD6-117	332	41661	0.7582	1	276	158	173	0.0057	171	167	1200.06	1.06	165.23
BD0-66	375	23375	0.3333	1	154	178	201	0.0047	198	190	1200.06	2.98	184.33
BD3-66	375	31795	0.4534	1	206	171	199	0.0062	196	190	1200.03	2.43	185.38
BD1-81	450	32351	0.3202	1	178	210	241	0.0063	238	229	1200.05	3.01	222.10
BD6-179	456	76631	0.7387	1	375	215	236	0.0089	235	230	1200.05	1.11	227.44
BD3-93	510	60415	0.4655	1	277	238	271	0.0126	267	259	1200.05	2.31	253.01
BD6-226	550	131497	0.8710	1	509	268	285	0.0141	282	276	1200.09	0.64	274.24
BD6-282	662	165676	0.7572	1	543	317	346	0.0228	339	333	1200.08	0.93	329.89
BD6-304	706	189128	0.7600	1	582	339	366	0.0217	363	356	1200.08	1.15	351.90
BD3-134	715	121300	0.4752	1	394	339	381	0.0216	374	363	1200.09	2.11	355.36
BD6-326	750	214085	0.7622	1	619	361	387	0.0248	384	378	1200.08	1.09	373.88
BD6-348	794	240592	0.7642	1	656	383	410	0.0257	409	400	1200.11	1.06	395.74
BD6-388	874	292808	0.7675	1	722	422	451	0.0308	448	440	1200.11	0.99	435.66
BD6-391	880	303144	0.7838	1	733	430	455	0.0326	452	443	1200.09	1.08	438.21
BD3-178	935	214148	0.4904	1	518	450	492	0.0394	484	472	1200.11	1.97	462.71
BD2-1	1000	163526	0.3274	1	379	478	525	0.0272	520	–	–	–	–
BD6-454	1006	390109	0.7717	1	827	488	522	0.0551	515	506	1200.14	0.89	501.50
BD6-456	1010	442443	0.8683	1	926	492	521	0.0500	515	507	1200.16	0.55	504.20
BD3-200	1045	263447	0.4830	1	560	501	548	0.0489	542	529	1200.23	1.85	519.23
BD5-1	1050	1147	0.0021	1	12	414	559	0.0041	557	–	–	–	–
BD6-481	1060	488155	0.8697	1	967	517	542	0.0516	540	532	1200.17	0.54	529.15
BD2-10	1450	341554	0.3251	1	530	695	760	0.0555	753	–	–	–	–
BD5-56	3800	3905	0.0005	1	13	1566	2061	0.0479	2057	–	–	–	–
BD2-61	4000	1981564	0.2478	1	1168	1911	2051	0.3798	2036	–	–	–	–
BD5-139	7950	7983	0.0003	1	14	3301	4291	0.2018	4290	–	–	–	–
BD5-162	9100	9107	0.0002	1	14	3770	4927	0.2379	4925	–	–	–	–
BD4-8	10200	25915758	0.4982	1	6053	4826	5001	3.7669	4977	–	–	–	–

6.2 Exact and heuristic performance

The computational results show a clear difference between the exact and heuristic approaches. CPLEX was executed on 900 instances. It obtained a feasible solution for 877 instances, while 23 instances terminated without a feasible solution within the imposed computational setting. Among the 877 instances with a feasible CPLEX solution, optimality was certified for 76 instances. Thus, optimality was certified for 8.44% of the 900 instances processed by CPLEX, corresponding to 6.35% of the complete benchmark repository.

All 1196 benchmark instances were successfully processed by the constructive heuristic. Since feasibility of the heuristic output is guaranteed by Theorem 30, this computation serves as an implementation verification. The maximum heuristic running time was 8.62 seconds, even though the benchmark includes instances with up to 14,100 vertices. The purification phase was applied to every heuristic solution, and all three purification strategies produced feasible minimal TPIDS, consistently with Theorem 25; this also serves as an implementation verification of the purification procedure.

The purification phase strictly reduced the solution cardinality on 1069 of the 1196 instances (89.38%). The average reduction in solution cardinality was 3.67 vertices, with a maximum reduction of 36 vertices. The computational study is descriptive rather than comparative: no claim of experimental superiority over existing heuristic or metaheuristic methods is made. To provide a more informative comparison of the purification strategies, we classify each instance according to the strategies that attain the minimum purified cardinality. Among the 1196 benchmark instances, P_1, P_2, and P_3 are uniquely best in 91 (7.61%), 81 (6.77%), and 107 (8.95%) instances, respectively. Ties between P_1 and P_2 occur in 50 instances (4.18%), between P_1 and P_3 in 59 (4.93%), and between P_2 and P_3 in 69 (5.77%). All three strategies attain the same minimum cardinality in 739 instances (61.79%). Thus, at least two strategies attain the minimum in 917 instances (76.67%). Table 3 summarizes these overlaps.

For the 877 instances for which CPLEX obtained a feasible solution, the average differences between the heuristic solutions and the CPLEX lower bound were 12.63 vertices for U_H and 10.11 vertices for U_P. Thus, purification reduced the average distance to the CPLEX lower bound by approximately 20%. Since LB is a lower bound and does not necessarily equal the optimum, these differences should not be interpreted as optimality errors.

For each instance with a feasible CPLEX solution, the relative distance shown in Fig. 1 is defined as $\frac{|U(G)|-LB(G)}{LB(G)}$, where $U(G)$ denotes either the constructive heuristic solution $U_H(G)$ or the purified solution $U_P(G)$, and $LB(G)$ is the CPLEX lower bound.

The horizontal axis uses $n+m$, consistently with the $O(n+m)$ running-time bounds established for the constructive heuristic and purification procedure when adjacency lists are used.

6.3 Quality on instances with certified optimality

The approximation guarantee is a worst-case theoretical statement valid for every connected graph of order at least three, whereas the empirical evaluation below is

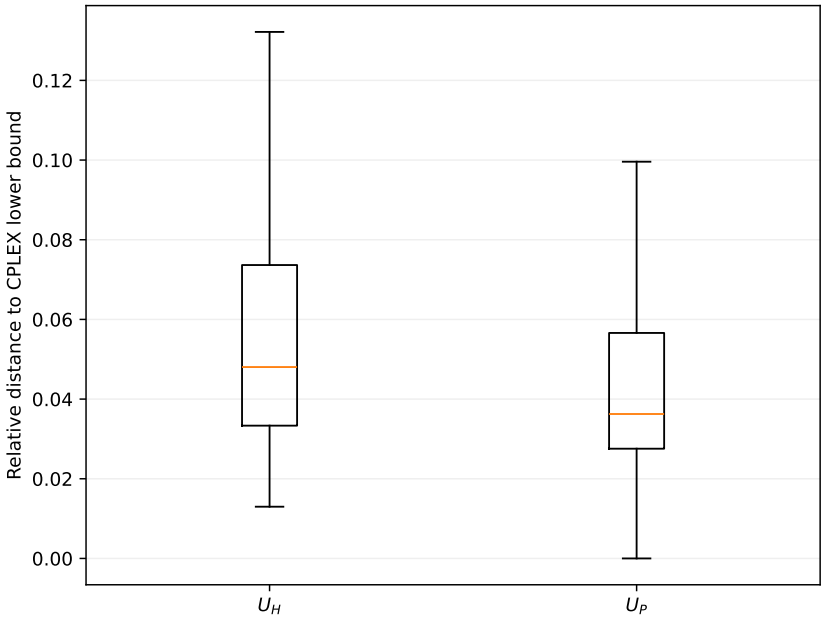


Fig. 1 Relative distance of the heuristic and purified solutions from the CPLEX lower bound over the 877 instances for which CPLEX found a feasible solution

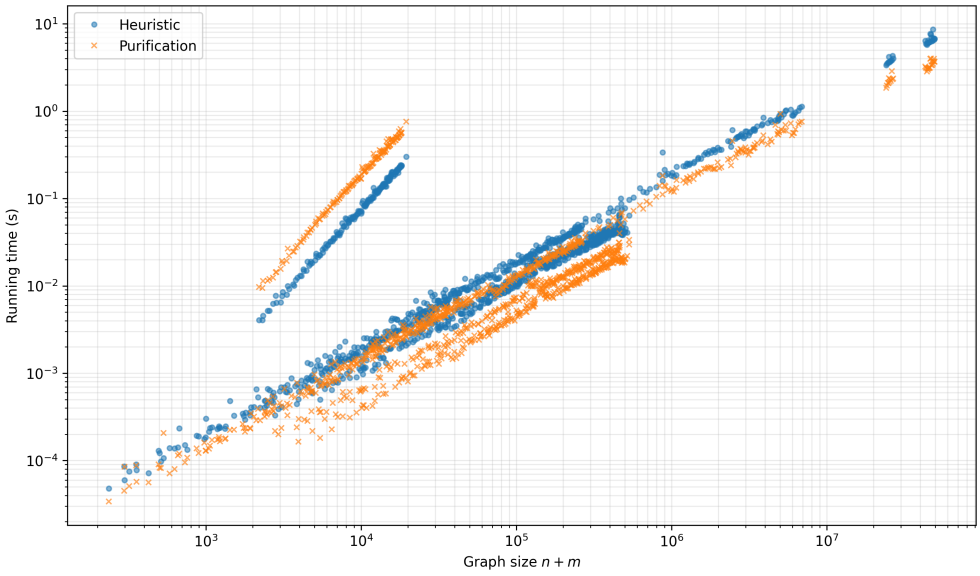


Fig. 2 Running time of the constructive heuristic and purification phase as a function of the graph size $n + m$, shown on logarithmic scales

Table 3 Overlap of the purification strategies on the minimum purified cardinality.

Strategies attaining the minimum	Instances	Percentage (%)
P_1 only	91	7.61
P_2 only	81	6.77
P_3 only	107	8.95
P_1 and P_2	50	4.18
P_1 and P_3	59	4.93
P_2 and P_3	69	5.77
P_1, P_2, and P_3	739	61.79
Total	1196	100.00

restricted to the 76 instances for which CPLEX certified optimality. A direct evaluation of the heuristic quality is possible on these instances. Let $OPT(G)$ denote the certified optimal value. On these instances, the constructive heuristic never reached the optimum, whereas the purified heuristic reached the optimum in one instance. Hence, U_P attained the optimum in 1/76 instances, or 1.32%.

The average absolute error of the constructive heuristic was $\frac{1}{76}\sum_G (U_H(G) - OPT(G)) = 3.20$ vertices, while purification reduced this value to $\frac{1}{76}\sum_G (U_P(G) - OPT(G)) = 2.37$ vertices. In relative terms, the corresponding average errors were 4.80% and 3.55%, respectively. The maximum absolute error of the purified heuristic on these instances was four vertices, and the largest observed ratio was $\max_G \frac{U_P(G)}{OPT(G)} = \frac{39}{36} = 1.0833$. These results show that purification improves the average quality of the constructive solutions on the certified-optimal instances, although the certified-optimal subset is relatively small compared with the complete benchmark repository.

6.4 Summary of the computational results

Overall, the experiments show that the proposed heuristic provides feasible TPIDS for all 1196 benchmark instances with low computational cost. The purification phase strictly reduces the constructive solution cardinality in 89.38% of the instances and reduces the average distance to the CPLEX lower bound from 12.63 to 10.11 vertices.

The detailed quality comparison on the 76 certified-optimal instances is reported in Section 6.3. In addition, the degree-sequence bound provides a separate theoretical reference for the instances on which CPLEX returned a feasible solution.

Proposition 19 shows analytically that $\ell_{TP}(G)$ dominates several previously established lower bounds, including those based on the minimum and maximum degrees, the regular-graph case, and total k-domination. We therefore use $\ell_{TP}(G)$ as the principal degree-sequence lower bound in the computational analysis. Among the 877 instances for which CPLEX obtained a feasible solution, $\ell_{TP}(G)$ was, on average, 10.43 vertices below the CPLEX lower bound, corresponding to an average relative difference of 5.59%.

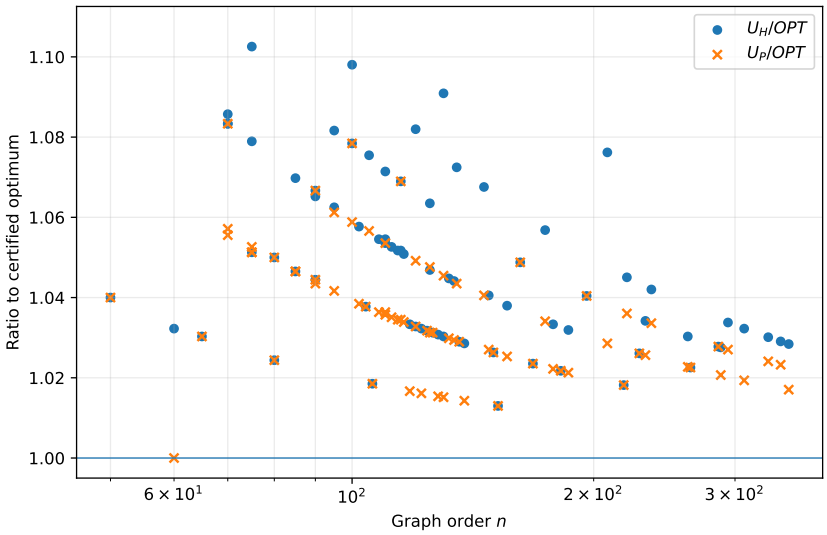


Fig. 3 Ratios of the constructive and purified heuristic solutions to the certified optimum as a function of the graph order, for the 76 instances for which CPLEX certified optimality

7 Conclusions

In this paper, we studied structural and computational aspects of total positive influence domination. We established lower and upper bounds in terms of classical graph parameters, together with extremal characterizations for several of them, and introduced a degree-sequence lower bound computable in linear time once the degree sequence is available. We also established analytically that this bound dominates several of the previously derived lower bounds. For the 877 instances with a feasible CPLEX solution, the degree-sequence bound $\ell_{TP}(G)$ was, on average, 10.43 vertices below the CPLEX lower bound, with an average relative difference of 5.59

We characterized minimal total positive influence dominating sets through private influence vertices and derived a purification procedure that preserves feasibility while removing redundant vertices. The purification strictly reduced the solution cardinality in 1069 of the 1196 instances (89.38%), with an average reduction of 3.67 vertices and a maximum reduction of 36 vertices.

The constructive heuristic combines support-vertex initialization with a maximum-total-active-degree greedy rule. Its analysis yields the instance-dependent guarantee

$$|U_P| \leq H(\kappa(G))\gamma_t^+(G) \leq H(\Delta(G))\gamma_t^+(G) \leq (\ln \Delta(G) + 1)\gamma_t^+(G),$$

where $\kappa(G)$ reflects the residual structure after initialization. The factor $H(\kappa(G))$ is never larger than $H(\Delta(G))$ and improves on it by an additive term of at most $\ln(\Delta(G)/\kappa(G))$.

Computationally, all 1196 benchmark graphs were processed, with a maximum constructive running time of 8.62 seconds. On the 76 instances for which CPLEX certified optimality, purification reduced the average absolute error from 3.20 to 2.37

vertices and the average relative error from 4.80% to 3.55%. These results complement the exact binary integer linear programming formulation and illustrate the practical behavior of the proposed heuristic and purification procedures on the benchmark instances. They are not intended to establish experimental superiority over methods developed for other formulations of positive influence domination.

Future work includes sharper structural bounds and extremal characterizations, the study of graph families for which $\kappa(G)$ is substantially smaller than $\Delta(G)$, stronger integer programming formulations, and tighter approximation guarantees. Further computational work should compare the proposed constructive procedure with existing greedy, local-search, and metaheuristic approaches; in particular, exchange neighborhoods could be investigated to further improve the cardinality of purified solutions. Such comparisons require particular care because the terminology used for positive influence domination is not uniform across the literature. In particular, Wang et al. [4] use the term positive influence dominating set for a formulation in which the influence condition is imposed on every vertex, whereas Dinh et al. [5] explicitly distinguish between PIDS, for which the condition is imposed only on vertices outside the selected set, and TPIDS, for which it is imposed on all vertices. Therefore, comparisons with previous algorithms should first align the problem definitions and then compare the corresponding solution methods. Further experiments on non-random graph families, including trees, grids, power-law graphs, and real-world networks, would also be useful to assess the behavior of the proposed methods beyond the current benchmark repository.

Statements and Declarations.

Funding. The authors received no financial support for the research, authorship, or publication of this article.

Competing interests. The authors declare no competing interests.

Contributions. Conceptualization, E.P.I. and J.M.S.A.; methodology, E.P.I., I.R.V., and J.M.S.A.; formal analysis, E.P.I., J.C.H.G., and J.M.S.A.; software, E.P.I. and I.R.V.; data curation, E.P.I.; validation, I.R.V. and J.C.H.G.; writing—original draft preparation, E.P.I.; writing—review and editing, I.R.V., J.C.H.G., and J.M.S.A.; supervision, J.M.S.A. All authors have read and agreed to the published version of the manuscript.

Data availability. The benchmark instances, source code, and computational results supporting this study are openly available in Mendeley Data through three complementary datasets: *Random Graph*, V10, DOI 10.17632/rr5bkj6dw5.10; *Random Graph BD4*, V1, DOI 10.17632/hyw9nckdsj.1; and *Random Graph BD5*, V1, DOI 10.17632/bhb7vpc2r8.1 [23–25]. The BD4 and BD5 datasets contain additional benchmark instances that are distributed separately because of repository size constraints. The benchmark used in this study consists of the BD0–BD3 and BD6 families from V10 (1006 instances), together with 30 BD4 instances and 160 BD5 instances from the corresponding V1 datasets. The V10 dataset also contains the source code used for the reported computational experiments and the corresponding experimental results, allowing the computational study to be reproduced from the archived data and code.

References

[1] T.W. Haynes, S.T. Hedetniemi, P.J. Slater, *Fundamentals of Domination in Graphs* (Marcel Dekker, New York, 1998)

[2] M.A. Henning, A. Yeo, *Total Domination in Graphs*. Springer Monographs in Mathematics (Springer, New York, 2013)

[3] F. Wang, E. Camacho, K. Xu, *Positive Influence Dominating Set in Online Social Networks*, in *Combinatorial Optimization and Applications - Third International Conference, COCOA 2009, Proceedings*, *Lecture Notes in Computer Science*, vol. 5573 (2009), pp. 313–321. https://doi.org/10.1007/978-3-642-02026-1_29

[4] F. Wang, H. Du, E. Camacho, K. Xu, W. Lee, Y. Shi, S. Shan, On positive influence dominating sets in social networks. Theoretical Computer Science **412**(3), 265–269 (2011). https://doi.org/10.1016/j.tcs.2009.10.001

[5] T.N. Dinh, Y. Shen, D.T. Nguyen, M.T. Thai, On the approximability of positive influence dominating set in social networks. Journal of Combinatorial Optimization **27**(3), 487–503 (2014). https://doi.org/10.1007/s10878-012-9530-7

[6] S. Bouamama, C. Blum, An improved greedy heuristic for the minimum positive influence dominating set problem in social networks. Algorithms **14**(79) (2021). https://doi.org/10.3390/a14030079

[7] D. Peleg, Local majorities, coalitions and monopolies in graphs: a review. Theoretical Computer Science **282**(2), 231–257 (2002). https://doi.org/10.1016/S0304-3975(01)00055-X

[8] D. Kuziak, I. Peterin, I.G. Yero, Open k-monopolies in graphs: complexity and related concepts. Discrete Mathematics and Theoretical Computer Science **18**(3) (2016). https://doi.org/10.46298/dmtcs.654

[9] G. Wang, H. Wang, X. Tao, J. Zhang, *A Self-Stabilizing Algorithm for Finding a Minimal Positive Influence Dominating Set in Social Networks*, in *Proceedings of the 24th Australasian Database Conference (ADC 2013)*, *Conferences in Research and Practice in Information Technology*, vol. 137 (2013), pp. 93–100

[10] D.V. Greetham, N. Charlton, A. Poghosyan Hopes, *Total Positive Influence Domination on Weighted Networks*, in *Complex Networks and Their Applications VIII*, *Studies in Computational Intelligence*, vol. 881, ed. by H. Cherifi, S. Gaito, J. Mendes, E. Moro, L.M. Rocha (Springer, 2020), pp. 325–336. https://doi.org/10.1007/978-3-030-36687-2_27

[11] F.N. Abu-Khzam, K. Lamaa, *Efficient Heuristic Algorithms for Positive-Influence Dominating Set in Social Networks*, in *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops* (2018), pp. 610–615. https:

//doi.org/10.1109/INFCOMW.2018.8406851

[12] R. Sun, J. Wu, C. Jin, Y. Wang, W. Zhou, M. Yin, An efficient local search algorithm for minimum positive influence dominating set problem. Computers & Operations Research **154**, 106197 (2023). https://doi.org/10.1016/j.cor.2023.106197

[13] I. Penedo, I. Lozano-Osorio, J. Sánchez-Oro, O. Cordón, *Solving the Minimum Positive Influence Dominating Set Problem in Social Networks Using Metaheuristics*, in *Variable Neighborhood Search: 11th International Conference, ICVNS 2025, Montreal, QC, Canada, May 12–14, 2025, Revised Selected Papers*, *Lecture Notes in Computer Science*, vol. 16256 (Springer, 2026), pp. 1–14. https://doi.org/10.1007/978-3-032-19582-1_1

[14] M.H. Nguyen, M.H. Ha, D.N. Nguyen, T.T. Tran, Solving the k-dominating set problem on very large-scale networks. Computational Social Networks **7**(4) (2020). https://doi.org/10.1186/s40649-020-00078-5

[15] E. Alvarez-Miranda, M. Sinnl, Exact and heuristic algorithms for the weighted total domination problem. Computers & Operations Research **127**, 105157 (2021). https://doi.org/10.1016/j.cor.2020.105157

[16] E. Parra Inza, N. Vakhania, J.M. Sigarreta Almira, F.A. Hernández Mira, Algorithms for the global domination problem. Computers & Operations Research **173**, 106876 (2025). https://doi.org/10.1016/j.cor.2024.106876

[17] H. Fernau, J.A. Rodríguez-Velázquez, J.M. Sigarreta, Global powerful r-alliances and total k-domination in graphs. Utilitas Mathematica **98**, 127–147 (2015)

[18] W. Hoeffding, Probability inequalities for sums of bounded random variables. Journal of the American Statistical Association **58**(301), 13–30 (1963). https://doi.org/10.1080/01621459.1963.10500830

[19] Z. Füredi, D. Mubayi, Signed domination in regular graphs and set-systems. Journal of Combinatorial Theory, Series B **76**(2), 223–239 (1999). https://doi.org/10.1006/jctb.1999.1905

[20] A. Harutyunyan, Global offensive alliances in graphs and random graphs. Discrete Applied Mathematics **164**, 522–526 (2014). https://doi.org/10.1016/j.dam.2013.08.018

[21] F. Hernández Mira, E. Parra Inza, J. Sigarreta Almira, N. Vakhania, A polynomial-time approximation to a minimum dominating set in a graph. Theoretical Computer Science **930**, 142–156 (2022). https://doi.org/10.1016/j.tcs.2022.07.020

[22] F. Hernández Mira, E. Parra Inza, J. Sigarreta Almira, N. Vakhania, Exact and heuristic algorithms for the domination problem. European Journal of Operational Research **313**(3), 926–936 (2024). https://doi.org/10.1016/j.ejor.2023.08.033

[23] E. Parra Inza. Random graph. Mendeley Data, V10 (2026). https://doi.org/10.17632/rr5bkj6dw5.10

[24] E. Parra Inza. Random graph bd4. Mendeley Data, V1 (2025). https://doi.org/10.17632/hyw9nckdsj.1

[25] E. Parra Inza. Random graph bd5. Mendeley Data, V1 (2025). https://doi.org/10.17632/bhb7vpc2r8.1