STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **x**, Month 202x, pp 0–13.
Published online in International Academic Press (www.IAPress.org)

Optimization and Heuristic Approaches for k-Domination Connectivity

Meghna K M*, Minirani S

MPSTME, SVKM's NMIMS Deemed to be University, Mumbai

Abstract Let G be a connected graph. Let $\kappa^{\gamma(k)}(G)$ denote the k-domination connectivity. This is defined as the minimum number of vertices whose removal disconnects G so that every resulting component has domination number exactly k. This parameter combines the structural robustness of vertex connectivity with domination-based coverage constraints. In this paper, we study structural properties and computational aspects of $\kappa^{\gamma(k)}(G)$. General bounds for the parameter are established. Its behavior is analyzed for several structured graph classes, including trees, grid graphs, corona graphs, and Cartesian product graphs. We also discuss relationships between $\kappa^{\gamma(k)}(G)$ and classical graph parameters such as vertex connectivity and domination number. From the computational perspective, we show that the decision version of the k-domination connectivity problem is DP-hard for fixed $k \geq 2$. We formulate the computation of $\kappa^{\gamma(k)}(G)$ as an integer optimization problem. We propose both an exact enumeration algorithm and a heuristic optimization strategy to determine the parameter in general graphs. Our results provide a framework for analyzing domination-based connectivity and help understand how networks can maintain connectivity and monitoring capability under vertex failures.

Keywords k-domination connectivity, domination number, vertex connectivity, combinatorial optimization, network resilience

DOI: 10.19139/soic-2310-5070-3578

1. Introduction

Modern communication and infrastructure networks require resilient mechanisms to ensure continuous functionality under failures. In many real-world scenarios, it is desirable to partition a network in such a way that each resulting component retains sufficient monitoring or control capability. This motivates the study of graph-theoretic parameters that simultaneously capture robustness and coverage properties.

Domination and connectivity are two fundamental and widely studied concepts in graph theory. Domination focuses on coverage, where a subset of vertices ensures that all other vertices are monitored or controlled. Variants such as double domination [8], vertex–edge domination [2], and related extensions [9] have been introduced to enhance reliability in applications like sensor networks and surveillance systems. Connectivity, on the other hand, measures the robustness of a network against failures. Conditional connectivity refines this notion by requiring that the remaining components after vertex or edge removal satisfy additional structural constraints [7].

The interplay between domination and connectivity has led to the development of conditional domination-based connectivity parameters. In this direction, Aldemir et al. [11] introduced the concept of k-domination edge connectivity, where edge removals result in components each having domination number k. Subsequently, Süleyman Ediz and Ziyattin Taş [13] proposed the vertex analogue, namely k-domination connectivity. Further investigations by Cruz and Sangeetha [12, 17] explored structural properties and exact values of $\kappa^{\gamma(k)}(G)$ for specific graph classes, along with applications in IoT systems.

*Correspondence to: Meghna K M (Email: MEGHNAK.M007@nmims.in). Department of Basic Sciences and Humanities, MPSTME, SVKM's NMIMS Deemed to be University, Mumbai, India

ISSN 2310-5070 (online) ISSN 2311-004X (print)

Despite these developments, the study of k-domination connectivity remains limited in scope. Existing results are largely confined to specific graph families and small values of k, with no general structural theory available for arbitrary graphs.Furthermore, there is limited knowledge about the computational complexity of the computation of $\kappa^{\gamma(k)}(G)$, and the literature does not present any algorithmic or optimization approach to calculate the parameter. This limits the use of the concept in real-world problems.

To address these challenges, this paper presents a detailed study of the k-domination connectivity parameter. We establish general structural properties and derive bounds that hold for arbitrary graphs, and we further examine its relationship with classical parameters such as domination number and k-domination edge connectivity. In addition, bounds for $\kappa^{\gamma(k)}(G)$ are obtained for several graph classes, including trees, grid graphs, corona graphs, and Cartesian product graphs. The computational aspects of the problem are also considered, where we discuss its complexity and propose optimization-based approaches for evaluating the parameter.

The organization of the paper is as follows. The necessary background and definitions are provided in Section 2. Structural properties and bounds for $\kappa^{\gamma(k)}(G)$ are developed in Section 3, followed by an examination of its connections with classical graph parameters in Section 4. The computational complexity of the problem is discussed in Section 5. Section 6 presents optimization formulations together with algorithmic approaches, while applications and illustrative examples are given in Section 7. The paper concludes with Section 8.

2. Preliminaries

We consider only finite, simple, and undirected graphs. All terminology and notation follow standard references in graph theory, including West [10], Diestel [1], and other classical texts in the literature [4, 5]. Let $G = (V, E)$ be a simple connected graph, where V is a finite nonempty set of vertices and E is a set of unordered pairs of distinct vertices, called edges. A grid graph is the Cartesian product of two path graphs, denoted by $P_m \square P_n$, where vertices are ordered pairs (i, j), and adjacency holds when $|i - i'| + |j - j'| = 1$. The Cartesian product of two graphs G_1 and G_2, written as $G_1 \square G_2$, has vertex set $V(G_1) \times V(G_2)$, where two vertices are adjacent if they differ in exactly one coordinate and the corresponding vertices are adjacent in that graph. The corona of two graphs G and H, denoted by $G \circ H$, is obtained by taking one copy of G and attaching to each of its vertices a copy of H by joining that vertex to all vertices of the corresponding copy of H. A subset $D \subseteq V$ is called a *dominating set* of G if every vertex in $V \setminus D$ is adjacent to at least one vertex in D. The connectivity of G, denoted by $\kappa(G)$, is defined as the minimum number of vertices whose removal either disconnects the graph or reduces it to a single vertex. A set $S \subseteq V(G)$ is called a k-domination vertex cut if $G - S$ is disconnected and every component of $G - S$ has domination number equal to k, where $k \geq 1$ [17]. The minimum cardinality of such a set S is called the k-domination connectivity of G, denoted by $\kappa^{\gamma(k)}(G)$ [17]. If no k-domination vertex cut exists, we define $\kappa^{\gamma(k)}(G) = \infty$.

3. Results for Special Graph Classes

In this section, we first present some general bounds and properties related to the k-domination connectivity of a graph, and then we investigate its behavior for various types of graphs including trees, paths, grids, corona graphs, and the Cartesian product of graphs.

Observation 3.1. Let G be a connected graph of order n. If G admits a k-domination vertex cut, then

$$\kappa^{\gamma(k)}(G) \leq n - k.$$

Proof
Let S be a k-domination vertex cut of G, and let H be any component of $G - S$. Since $\gamma(H) = k$, we must have $|V(H)| \geq k$.

$$n - |S| = \sum_{i=1}^{r} |V(H_i)| \geq rk \geq k,$$

which implies $|S| \leq n - k$. Taking the minimum over all such cut sets gives the result. □

This result provides a universal upper bound for $\kappa^{\gamma(k)}(G)$ in terms of graph order and domination requirement. It shows that larger domination constraints restrict the size of admissible vertex cuts.

Corollary 3.1
Let T be a tree of order $n \geq 2$ that admits a k-domination vertex cut. Then

$$1 \leq \kappa^{\gamma(k)}(T) \leq n - rk,$$

where r is the number of connected components of $T - S$ corresponding to a minimum k-domination vertex cut S.

Proof
Since T is connected, at least one vertex must be removed to disconnect it, so $\kappa^{\gamma(k)}(T) \geq 1$.

Let S be a minimum k-domination vertex cut and let $T - S$ have r components $T_1, \ldots, T_r$. Because $\gamma(T_i) = k$, we have $|V(T_i)| \geq k$ for each i. Thus,

$$n - |S| = \sum_{i=1}^{r} |V(T_i)| \geq rk,$$

which implies $|S| \leq n - rk$. □

For trees, the bound becomes tighter due to their acyclic structure. The number of resulting components plays a crucial role, showing how fragmentation impacts domination-based connectivity.

Remark 3.2
Let T be a tree of order $n \geq 2$. If there exists a vertex whose removal produces components each having domination number 1, then $\kappa^{\gamma(1)}(T) = 1$.

More generally, for $k \geq 1$, the parameter $\kappa^{\gamma(k)}(T)$ is well-defined if and only if T admits a vertex cut whose resulting components each has a domination number k.

As a concrete illustration, consider the path graph P_3 with $k = 1$. Removing the central vertex disconnects the graph into two isolated vertices, each having domination number 1. Thus,

$$\kappa^{\gamma(1)}(P_3) = 1.$$

Since $n = 3$ and the number of components $r = 2$, we have

$$n - rk = 3 - 2 = 1,$$

showing that both the lower and upper bounds are attained.

This example demonstrates that the derived bounds are sharp. It also provides intuition on how small graph structures exactly realize theoretical limits.

Example 3.3 (Grid Graphs)
Let $G = P_m \Box P_n$, the Cartesian product of two paths (a grid graph). Since $P_m \Box P_n$ is 2-connected for $m, n \geq 3$, any vertex cut has size at least 2; therefore, if a valid k-domination vertex cut exists, its size is at least 2.

$$\kappa^{\gamma(k)}(G) \leq mn - rk,$$

where r is the number of components formed after the removal of a vertex set S, each satisfying $\gamma = k$.

Grid graphs exhibit higher connectivity, which increases the minimum size of admissible cuts. This makes satisfying k-domination conditions more restrictive compared to trees.

Example 3.4 (Corona Graphs)
Let $G = H \circ K_1$, where H is connected[3]. Then

$$\kappa^{\gamma(k)}(G) = \infty \quad \text{for } k \geq 2,$$

since the removal of any vertex of H produces an isolated vertex, and hence at least one component with domination number 1. Valid k-domination vertex cuts exist only when $k = 1$.

Corona graphs illustrate cases where the parameter becomes undefined (infinite). The presence of pendant vertices forces domination number 1 in some components, preventing higher k values.

Example 3.5 (Cartesian Product Graphs)
Let $G = G_1 \square G_2$[14]. If G_1 and G_2 are connected graphs, then their Cartesian product G is also connected and, in general, possesses comparatively high connectivity. Since any k-domination vertex cut is a vertex cut, and

$$\kappa^{\gamma(k)}(G) \geq \min\{\kappa(G_1)|V(G_2)|,\ \kappa(G_2)|V(G_1)|,\ \delta(G_1) + \delta(G_2)\}.$$

If a valid cut exists producing r components each with domination number k, then

$$\kappa^{\gamma(k)}(G) \leq |V(G_1)||V(G_2)| - rk.$$

Thus, the results follow, wherever a valid k-domination vertex cut exists. If no such vertex cut exists, then it is undefined.

In conclusion, the findings presented above clearly show that the k-domination connectivity invariant is very much dependent on graph structure. For trees and paths, one can get better bounds or even determine their exact values. However, for graphs of greater complexity like grids or Cartesian products, more restrictions apply.

4. Relationships with Classical Graph Parameters

In addition to the structural constraints and the consequences of the previous sections on special graph classes, we analyze the relationship between $\kappa^{\gamma(k)}(G)$ and the existing classical graph parameters. This approach allows one to establish the place of domination connectivity in the general context of graph theory. This section examines the relationship between the k-domination connectivity $\kappa^{\gamma(k)}(G)$ and classical graph parameters such as $\gamma(G)$ and $\lambda^{\gamma(k)}(G)$.

4.1. Illustrative Examples

The following examples follow directly from known formulas and are presented for illustration.

Example 4.1
For paths and cycles,

$$\kappa^{\gamma(2)}(G) \leq \lambda^{\gamma(2)}(G).$$

In particular:
Path P_n:

$$\lambda^{\gamma(2)}(P_n) = \left\lfloor \frac{n-1}{6} \right\rfloor, \quad \kappa^{\gamma(2)}(P_n) = \left\lfloor \frac{n}{7} \right\rfloor.$$

Cycle C_n:

$$\lambda^{\gamma(2)}(C_n) = \left\lceil \frac{n}{6} \right\rceil, \quad \kappa^{\gamma(2)}(C_n) = \left\lceil \frac{n}{7} \right\rceil.$$

These examples indicate that edge-based k-domination connectivity can dominate the vertex-based counterpart in certain graph classes. They also show how slight structural differences between paths and cycles influence the parameter values.

4.2. Relation with Domination Number

Observation 4.1. Let G be a connected graph admitting a k-domination vertex cut, and let S be a minimum such cut. If $G - S$ has t components, then

$$\gamma(G) \leq \kappa^{\gamma(k)}(G) + tk.$$

Proof
Let S be a minimum k-domination vertex cut of G, so that $|S| = \kappa^{\gamma(k)}(G)$. Let $H_1, H_2, \ldots, H_t$ be the components of $G - S$.

By definition, each component H_i admits a dominating set D_i such that $|D_i| = k$. Let

$$D = \bigcup_{i=1}^{t} D_i.$$

Then D dominates all vertices in $G - S$.

Now consider the set $S \cup D$. Every vertex in S is dominated since it belongs to the set, and every vertex in $G - S$ is dominated by D. Hence, $S \cup D$ is a dominating set of G.

Therefore,

$$\gamma(G) \leq |S \cup D| \leq |S| + |D|.$$

Since each $|D_i| = k$, we have

$$|D| = \left| \bigcup_{i=1}^{t} D_i \right| \leq \sum_{i=1}^{t} |D_i| = tk.$$

Thus,

$$\gamma(G) \leq \kappa^{\gamma(k)}(G) + tk.$$

□

This inequality establishes a direct upper bound on the domination number in terms of k-domination connectivity and the number of resulting components. It reflects how fragmentation under domination constraints contributes to the overall domination requirement of the graph.

In conclusion, the interconnections between these parameters presented above show that the k-domination connectivity parameter is highly interconnected with the traditional graph parameters like domination number and edge dominating connectivity. Such interrelations give useful bounds on their behavior, thus providing insight into the impact of their interplay on the value of $\kappa^{\gamma(k)}(G)$. The consideration of the computational side of this problem follows in the next part of this paper.

5. Extremal Bounds and Computational Aspects

In this section, we analyze the extremal behavior of the k-domination connectivity parameter and establish its computational complexity. In particular, we show that determining $\kappa^{\gamma(k)}(G)$ is computationally intractable for fixed $k \geq 2$, which motivates the use of algorithmic and heuristic approaches.

5.1. Extremal behavior

The value of $\kappa^{\gamma(k)}(G)$ depends strongly on the structural characteristics of the graph. In certain cases, it can be as small as 1, while for highly connected graphs it may grow close to its theoretical upper bound.

Consider the case where the value will be equal to one when deleting a particular vertex will result in a graph being split into different components that all have domination numbers of k. For example, the star graph denoted by $K_{1,n-1}$ has its domination number reduced to 1 when removing the central vertex, which makes $\kappa^{\gamma(1)}(K_{1,n-1}) = 1$. The above examples help us conclude that a centralized structure of a graph usually implies a smaller value of k-domination connectivity.

On the other hand, dense and well-connected graphs need a greater number of deleted vertices before obtaining desired subgraphs that fulfill the domination condition. These observations indicate that graphs with centralised structures tend to admit smaller k-domination vertex cuts, whereas highly connected graphs impose stricter constraints, leading to larger values of $\kappa^{\gamma(k)}(G)$.

5.2. Computational Complexity of $\kappa^{\gamma(k)}(G)$

The computation of $\kappa^{\gamma(k)}(G)$ requires determining a minimum vertex set $S \subseteq V(G)$ whose removal disconnects the graph such that every connected component H of $G - S$ satisfies $\gamma(H) = k$. While the classical connectivity of a graph can be computed in polynomial time[15], the additional domination constraint significantly increases the problem's complexity.

This is an indication that the problem involves both structural and dominance constraints, making it intrinsically harder than the classical problem of connectivity. For instance, the decision problem of checking whether a graph meets the criteria of $\gamma(G) = k$ is DP-complete [6]. As $\kappa^{\gamma(k)}(G)$ needs to check the same condition for every component of $G - S$, its decision version involves a subproblem which is DP-complete. The following theorem formalizes this intuition by establishing the decision problem's DP-hardness.

Theorem 5.1
For a fixed integer $k \geq 2$, the decision version of the k-domination connectivity problem is DP-hard.

Proof
We consider the following decision problem: given a graph G and an integer ℓ, determine whether there exists a set $S \subseteq V(G)$ with $|S| \leq \ell$ such that $G - S$ is disconnected and each connected component of $G - S$ has domination number exactly k.

We give a polynomial-time reduction from the domination equality problem, which asks whether $\gamma(G) \leq k$ and $\gamma(G) \geq k$. This problem is known to be DP-complete.

Let G be an arbitrary graph. Construct a graph H as follows. Take two disjoint copies of G, denoted by G_1 and G_2, and add a new vertex x adjacent to every vertex of both G_1 and G_2. Thus, x is a universal vertex in H. Clearly, this construction can be carried out in polynomial time.

We set $\ell = 1$. Observe that H is connected, since x is adjacent to all other vertices.

We now prove that $\gamma(G) = k$ if and only if there exists a set $S \subseteq V(H)$ with $|S| \leq 1$ such that $H - S$ is disconnected, and each component has domination number exactly k.

($\Rightarrow$) Suppose $\gamma(G) = k$. Let $S = \{x\}$. Then

$$H - S = G_1 \cup G_2,$$

which consists of two connected components. Since G_1 and G_2 are isomorphic to G, we have $\gamma(G_1) = \gamma(G_2) = k$. Hence, S is a valid solution.

($\Leftarrow$) Conversely, suppose there exists a set $S \subseteq V(H)$ with $|S| \leq 1$ such that $H - S$ is disconnected, and each connected component has domination number exactly k.

If $S = \emptyset$, then $H - S = H$, which is connected, a contradiction.

If $S = \{v\}$ for some $v \in V(G_1) \cup V(G_2)$, then the vertex x remains in the graph, and since x is adjacent to all other vertices, $H - S$ remains connected, again a contradiction.

Therefore, the only possible choice is $S = \{x\}$. In this case,

$$H - S = G_1 \cup G_2.$$

Since each component must have domination number exactly k, we obtain $\gamma(G_1) = \gamma(G_2) = k$. As both are copies of G, it follows that $\gamma(G) = k$. Thus, the only vertex whose removal can disconnect H is the universal vertex x.

Thus, $\gamma(G) = k$ if and only if there exists a set S with $|S| \leq 1$ such that $H - S$ is disconnected, and each component has domination number exactly k.

This establishes a polynomial-time reduction from a DP-complete problem. Hence, the decision version of the k-domination connectivity problem is DP-hard. □

The DP-hardness result indicates a higher computational complexity than standard NP-hardness. While NP-hard problems often admit approximation or heuristic solutions with performance guarantees, DP-hard problems involve the interaction of both NP and co-NP conditions, making such guarantees more difficult to obtain. This further justifies the use of heuristic approaches for approximating $\kappa^{\gamma(k)}(G)$ in large-scale graphs.

This emphasizes the necessity of developing efficient approximation or heuristic-based methods.

Remark 5.2
The restriction $k \geq 2$ is necessary. For $k = 1$, the problem can be decided in polynomial time. A graph has domination number $\gamma(G) = 1$ if and only if it contains a universal vertex, that is, a vertex adjacent to all other vertices. Hence, after removing a vertex set S, it suffices to check whether each component of $G - S$ contains a vertex of degree equal to the size of that component minus one. Since connectivity and degree verification can be performed in polynomial time, the problem for $k = 1$ is polynomial-time solvable.

This shows that the complexity sharply differs between trivial and non-trivial domination constraints.

This section demonstrates that while the k-domination connectivity parameter admits meaningful structural bounds, its computation is difficult due to DP-hardness. These results justify the development of optimization-based and heuristic methods discussed in the next section.

6. Optimization Approach for the Parameter Computation

In view of the computational hardness established in the previous section, we develop optimization-based approaches for computing the k-domination connectivity parameter. These include an exact branch-and-bound algorithm for small graphs and a genetic algorithm for efficiently approximating solutions in larger instances.

The computation of $\kappa^{\gamma(k)}(G)$ requires identifying a minimum vertex set whose removal disconnects the graph such that each resulting component satisfies a domination constraint. This problem combines structural and domination conditions, making it computationally challenging. Given that domination problems are already hard for general graphs, the additional connectivity requirement further increases the complexity. Consequently, exact computation becomes impractical for large graphs, motivating the use of optimization and approximation techniques.

6.1. Conceptual Optimization Formulation

The formulation below is conceptual, as connectivity and domination constraints are nonlinear and require auxiliary modeling.
Let $G = (V, E)$ be a connected graph on n vertices. We define binary decision variables:

$$x_i = \begin{cases} 1, & \text{if vertex } v_i \text{ is removed,} \\ 0, & \text{otherwise.} \end{cases}$$

The objective is to minimize the number of removed vertices:

$$\text{Minimize:} \quad \sum_{i=1}^{n} x_i$$

subject to the following conditions:

- The graph $G - S$, where $S = \{v_i : x_i = 1\}$, is disconnected.
- For each connected component C_j of $G - S$, the domination number satisfies $\gamma(C_j) = k$.
- $x_i \in \{0, 1\}$ for all $i = 1, 2, \ldots, n$.

This formulation provides a general optimization perspective on the problem. The connectivity and domination constraints are inherently non-linear and require auxiliary variables and constraint modeling techniques. As a result, the formulation serves as a conceptual framework rather than a directly solvable linear program.

6.2. *Exact Computation of $\kappa^{\gamma(k)}(G)$ via Branch and Bound*

The evaluation of $\kappa^{\gamma(k)}(G)$ is a combinatorial problem requiring a vertex subset whose removal yields components satisfying a domination condition. Exhaustive enumeration becomes infeasible for large graphs. To address this, we propose a branch-and-bound algorithm that efficiently explores candidate subsets while pruning unpromising regions of the search space.

Given a subset $S \subseteq V$, we first check whether the graph $G - S$ is disconnected. If not, the subset is discarded. Otherwise, the domination number of each connected component of $G - S$ is computed, and the subset is considered feasible only if every component has domination number equal to k. Among all feasible subsets, the one with minimum cardinality determines $\kappa^{\gamma(k)}(G)$.

The efficiency of the algorithm is improved through pruning strategies: any subset S with $|S|$ not smaller than the current best solution is discarded; subsets for which $G - S$ remains connected are ignored; and if any component of $G - S$ fails to satisfy the domination condition (i.e., $\gamma(C) \neq k$), the evaluation is terminated early.

The detailed steps of the algorithm are presented in Algorithm 1.

Algorithm 1 Exact Computation of $\kappa^{\gamma(k)}(G)$ using Branch and Bound

```
Input: Connected graph G = (V, E), integer k ≥ 1
Output: κ^{γ(k)}(G) (if it exists)
min_cut ← |V|
for r = 1 to |V| do
    if r ≥ min_cut then
        break
    end if
    for each subset S ⊆ V with |S| = r do
        H ← G − S
        if H has more than one connected component then
            valid ← True
            for each component C of H do
                if γ(C) ≠ k then
                    valid ← False
                    break
                end if
            end for
            if valid then
                min_cut ← r
                break
            end if
        end if
    end for
end for
if min_cut < |V| then
    return min_cut
else
    return "Does not exist"
end if
```

The worst-case time complexity of the algorithm is exponential in $|V|$, as it involves enumeration of vertex subsets. However, the use of pruning strategies discussed above significantly reduces the search space, making the algorithm more difficult than naive exhaustive search for moderate sized graphs.

6.3. Genetic Algorithm for Approximating $\kappa^{\gamma(k)}(G)$

For large graphs, exact computation of $\kappa^{\gamma(k)}(G)$ becomes computationally intractable. Therefore, we employ a Genetic Algorithm (GA) to obtain near-optimal solutions. Metaheuristic techniques such as genetic algorithms and particle swarm optimization have been shown to be effective for solving complex combinatorial problems [16].

The proposed GA is designed to approximate $\kappa^{\gamma(k)}(G)$ efficiently by exploring the search space of vertex subsets while balancing the size of the cut and domination constraints. The overall procedure is summarized in Algorithm 2.

Algorithm 2 Genetic Algorithm for Approximating $\kappa^{\gamma(k)}(G)$

Input: Graph $G = (V, E)$, integer $k \geq 1$
Output: Approximate value of $\kappa^{\gamma(k)}(G)$
Initialize population P of binary vectors of length $|V|$
Set parameters: population size N, mutation rate p_m, penalty λ
while termination criterion not met **do**
    for each solution $S \in P$ **do**
        Construct $H \leftarrow G - S$
        if H is disconnected **then**
            Compute $\gamma(C_j)$ for each component C_j of H
            Evaluate fitness:

$$f(S) = |S| + \lambda \sum_j |\gamma(C_j) - k|$$

        else
            Assign a large penalty to $f(S)$
        end if
    end for
    Select parent solutions based on fitness
    Apply crossover to generate offspring
    Apply mutation: flip each bit with probability p_m
    Update population P with new solutions
end while
return best solution S^* with minimum fitness value

Solution Representation: Each candidate solution is encoded as a binary vector $S = (s_1, \ldots, s_n)$, where $s_i = 1$ indicates that vertex v_i is removed.

Initial Population: The initial population is generated using a combination of random initialization and degree-based heuristics to ensure diversity and improve solution quality.

Fitness Function: For a subset S, the fitness is defined as

$$f(S) = |S| + \lambda \sum_j |\gamma(C_j) - k|,$$

where the first term minimizes the size of the cut set and the second term penalizes violations of the domination condition in each component C_j of $G - S$.

Genetic Operators: Selection favors solutions with lower fitness values. Crossover combines two parent solutions to generate offspring, while mutation flips bits with a small probability to maintain diversity in the population.

Parameter Settings: Typical values include population size $N = 50$, mutation rate $p_m = 0.1$, and penalty parameter $\lambda = 10$.

Termination Criteria: The algorithm terminates after a fixed number of iterations or when the population converges, i.e., no significant improvement in fitness values is observed.

This heuristic approach provides an approximate solution to $\kappa^{\gamma(k)}(G)$. Its effectiveness can be assessed by comparing the obtained solutions with exact values for small graphs. The proposed GA balances solution feasibility and computational efficiency making it suitable for large-scale instances where exact computation is infeasible.

6.4. Computational Experiments

To examine the effectiveness of the heuristic optimization approach developed, experimental analysis was performed using different graphs such as paths, cycles, and grids. Moreover, the outcomes received from the proposed heuristic approach were also compared to the exact algorithmic approach.

The main goal of the computations was the evaluation of the performance of the proposed heuristic algorithm based on (a) comparing the quality of the heuristic solution with that of the optimal solution, and (b) the computational cost of the heuristic algorithm compared to the exact algorithm. In cases of small graphs, the heuristic produced an optimal solution.

These findings suggest that the heuristic algorithm performs efficiently and yields reliable results, especially for large graphs where exact calculations are computationally infeasible. As shown in Table 1, the heuristic algorithm consistently matches the exact $\kappa^{\gamma(k)}(G)$ values (k=1) for all tested small graphs while requiring significantly less computation time. These results suggest that the heuristic can serve as a practical approach for approximating $\kappa^{\gamma(k)}(G)$, particularly for larger graphs.

Table 1. Comparison of Exact and Heuristic methods for computing $\kappa^{\gamma(k)}(G)$

Graph	Size	Exact Value	Heuristic Value	Time (Exact)	Time (Heuristic)
P_{10}	10	2	2	0.005564 sec	0.001074 sec
C_{10}	10	3	3	0.008015 sec	0.000971 sec
Grid 3×3	9	3	3	0.004117 sec	0.000985 sec

The results presented in Table 1 show that the heuristic approach consistently matches the exact solution for small graphs while significantly reducing computation time. This demonstrates both the accuracy and efficiency of the proposed methods. For larger graphs, where exact computation is infeasible, the heuristic provides a scalable alternative.

In conclusion, in this section, two categories of approaches to address the problem have been introduced, where the former category of approaches could deliver accurate solutions to the problem, while the latter category of approaches delivers near-optimal solutions. The Branch and Bound Approach could solve the problem in an exact manner, but due to its high time complexity, this approach is not scalable for large-scale problems. As a remedy to the above problem, we introduce our proposed heuristic algorithm based on Genetic Algorithms.

7. Applications

The concept of k-domination connectivity provides a framework that is used to make a thorough study of the interconnection between the notions of structural stability and functional efficiency in graph structures. The k-domination connectivity concept finds application in the disciplines of network design, communication, and combinatorial optimization. Below are some areas in which the k-domination connectivity concept has meaningful applications.

7.1. Application to Emergency Response Network Resilience

The parameter $\kappa^{\gamma(k)}(G)$ has the potential to provide a theoretical framework for measuring resilience in emergency response networks. If we take the urban transportation network as a graph $G = (V, E)$ where the vertices in the graph represent the intersections of roads and the edges in the graph represent the connectivity between the intersections, then the concept of a dominating set in the context of the graph would be the emergency centers located in strategic places to serve the neighboring intersections.

The parameter $\kappa^{\gamma(k)}(G)$ represents the minimum number of vertices to be removed from the network such that the network is disconnected in a manner such that every component has a domination number equal to k.

Thus, $\kappa^{\gamma(k)}(G)$ represents a resilience threshold indicating the number of critical vertex failures required to modify the supervisory coverage of a network to a specified level. Formally, it can be viewed as a combinatorial optimization problem that finds a minimum vertex cut S such that each component H of $G - S$ satisfies $\gamma(H) = k$.

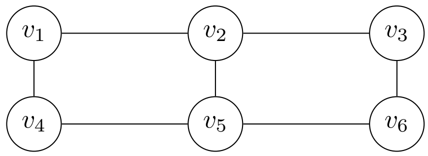

Figure 1. A 2×3 grid graph $G = P_2 \square P_3$

7.1.1. Illustrative Example on a Grid Network: Consider the grid graph shown in Figure 1. Let $k = 1$.

Step 1: Removing any single vertex does not disconnect the graph, so no singleton set can serve as a k-domination vertex cut.

Step 2: Consider the vertex set $S = \{v_2, v_5\}$, consisting of the middle vertices in each row, as illustrated in Figure 2. Then,

$$G - S = \{v_1, v_4\} \cup \{v_3, v_6\},$$

which results in two disconnected components.

Step 3: Each component is a path on two vertices, and hence has domination number $\gamma = 1$.

Thus,

$$\kappa^{\gamma(1)}(G) = 2.$$

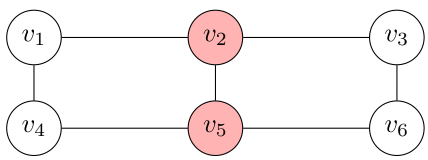

Figure 2. Removal of vertices v_2 and v_5 disconnects the network

Mini Case Study and Comparative Insight:

This example may be interpreted as a simplified model of a service network in which intersections correspond to locations equipped with essential facilities such as hospitals. The value $\kappa^{\gamma(1)}(G) = 2$ indicates that the network can tolerate the failure of any single node without compromising connectivity or minimum service coverage. However, the simultaneous removal of two strategically positioned nodes partitions the network into smaller independent regions, each still maintaining the required domination level.

This example demonstrates that even in moderately larger grid structures, the interplay between connectivity and domination constraints plays a crucial role in determining network resilience. In particular, larger values of $\kappa^{\gamma(k)}(G)$ correspond to higher robustness, highlighting the importance of this parameter in the comparative analysis and design of resilient network topologies.

Algorithm 3 provides a baseline exhaustive procedure tailored to the EMS network interpretation and is suitable for small-scale network instances such as the case study considered.

Algorithm 3 Baseline Computation of $\kappa^{\gamma(k)}(G)$ for EMS Network Resilience

Require: Graph $G = (V, E)$ representing an EMS network (vertices = hospital-equipped junctions, edges = roads); integer $k \geq 1$

Ensure: Value of $\kappa^{\gamma(k)}(G)$ and a minimum critical set S

function DOMINATIONNUMBER(H)
  for $r = 1$ to $|V(H)|$ **do**
    for each subset $D \subseteq V(H)$ with $|D| = r$ **do**
      if every vertex in $V(H)$ is in D or adjacent to a vertex in D **then**
        return r
      end if
    end for
  end for
end function

Interpretation: Identify the minimum set of failed hospital locations whose removal fragments the network into regions each maintaining exactly k hospitals.
for $r = 1$ to $|V| - 1$ **do**
  for each subset $S \subseteq V$ with $|S| = r$ **do**
    Remove hospital nodes in S to obtain $H = G - S$
    if H is disconnected **then**
      Let $\{C_1, C_2, \ldots, C_t\}$ be the components of H
      valid $\leftarrow$ TRUE
      for each component C_i **do**
        Compute $\gamma(C_i)$ using DOMINATIONNUMBER(C_i)
        if $\gamma(C_i) \neq k$ **then**
          valid $\leftarrow$ FALSE
          break
        end if
      end for
      if valid $=$ TRUE **then**
        return $\kappa^{\gamma(k)}(G) = r$, S
      end if
    end if
  end for
end for
return $\kappa^{\gamma(k)}(G) =$ undefined

7.2. Distributed Computing and Task Allocation

The parameter $\kappa^{\gamma(k)}(G)$ serves as a theoretical measure for fault tolerance in a distributed computing system. For a clustered system, in which some nodes act as supervisory nodes to coordinate the execution of tasks within clusters, the condition that every functioning cluster contains at least k supervisory nodes might be described in terms of the existence of a k-dominating set. In the context of k-domination connectivity, the value represents the number of processors that might fail before the subgraphs fail to maintain the necessary supervisory structure.

Furthermore, this parameter may provide a quantitative perspective for assessing robustness under adversarial conditions, where node failures are not necessarily random. Consequently, our theoretical results suggest that $\kappa^{\gamma(k)}(G)$ could be used as a design-oriented metric to enhance redundancy and reliability in cloud computing and parallel processing systems.

7.3. Infrastructures and Transportation Networks

Another use of the parameter $\kappa^{\gamma(k)}(G)$ is in the analysis of the resilience of infrastructure and transportation networks. The parameter can be used to identify key nodes in the network, which, if removed, would cause the network to be divided into smaller functional units. It would facilitate an analysis of network weaknesses and strengthen network robustness. It would enable the placement of the control units such that the network is fully functional and self-reliant despite any damage. In addition to this, the parameter facilitates comparison of networks based on their robustness.

As demonstrated in this section, the k-domination connectivity parameter has been explained using applications that involve distributed computing, infrastructure systems, and emergency response networks. This makes the case stronger for the utility of the concept, since it is evident from the discussion above that it acts as an effective robustness measure that considers both disconnectivity and domination aspects of functionality. The grid network example and the associated algorithm have been presented to make the point clear about the interpretation of the concept in real-life scenarios.

8. Conclusion and Future Directions

This paper investigates the k-domination connectivity parameter $\kappa^{\gamma(k)}(G)$ and its behavior across different graph structures. General bounds are established, and its connections with classical parameters such as connectivity and domination number are explored. From a computational standpoint, the decision version of the problem is shown to be difficult due to its DP-hard nature, which motivates the use of both exact and heuristic methods. While exact approaches may not be practical for large graphs, the heuristic method demonstrates good performance on typical instances. In addition to its theoretical significance, the parameter provides a meaningful framework for analyzing network resilience under coverage constraints. The results also indicate several avenues for future research, including the refinement of bounds and the development of more efficient computational techniques for wider classes of graphs.

Author Contributions

All authors contributed equally in developing the concept of the problem, analysis and computations, interpretation of the results, and preparation of the manuscript.

Acknowledgements

The authors would like to thank the reviewers for their constructive suggestions and remarks, which helped improve the manuscript.

REFERENCES

[1] R. Diestel, *Graph Theory*, Springer, 2024.

[2] R. Boutrig, M. Chellali, T. W. Haynes, and S. T. Hedetniemi, *Vertex-edge domination in graphs*, Aequationes Mathematicae, vol. 90, no. 2, pp. 355–366, 2016.

[3] M. Yamuna, and K. Karthika, *A survey on the effect of graph operations on the domination number of a graph*, International Journal of Engineering and Technology, vol. 8, no. 6, pp. 2749–2771, 2016.

[4] J. A. Bondy, and U. S. R. Murty, *Graph Theory with Applications*, Macmillan, 1976.

[5] T. W. Haynes, S. T. Hedetniemi, and P. Slater, *Fundamentals of Domination in Graphs*, CRC Press, 2013.

[6] F.-T. Hu, and J.-M. Xu, *On the complexity of the bondage and reinforcement problems*, Journal of Complexity, vol. 28, no. 2, pp. 192–201, 2012.

[7] F. Harary, Conditional connectivity, *Networks*, vol. 13, pp. 347–357, 1983.

[8] S. Mehry, and R. Safakish, *Three domination number and connectivity in graphs*, arXiv preprint, 2015.

[9] P. Maniya, and D. Pradhan, *Towards the conjecture on domination versus edge domination in graphs*, Bulletin of the Malaysian Mathematical Sciences Society, vol. 47, no. 1, pp. 1–7, 2024.

[10] D. B. West, *Introduction to Graph Theory*, 2nd ed., Prentice Hall, 2001.

[11] M. Aldemir, S. Ediz, İ. Çiftçi, K. Yamaç, and Z. Taş, *Domination edge connectivity of graphs*, Zenodo, pp. 1–10, 2024.

[12] M. S. D. Cruz, and V. Sangeetha, *k-domination connectivity in graphs*, International Journal of Computer Mathematics, pp. 1–21, 2025.

[13] S. Ediz, and Z. Taş, *A note on domination connectivity of paths and cycles*, Journal of Graph Theory Studies, vol. 6, no. 2, pp. 49–51, 2024.

[14] J.-M. Xu, and C. Yang, *Connectivity and super-connectivity of Cartesian product graphs*, Ars Combinatoria, vol. 95, pp. 235–245, 2010.

[15] S. Mattia, *Reformulations and complexity of the clique interdiction problem by graph mapping*, Discrete Applied Mathematics, vol. 354, pp. 48–57, 2024.

[16] A. G. Gad, *Particle swarm optimization algorithm and its applications: A systematic review*, Archives of Computational Methods in Engineering, vol. 29, no. 5, pp. 2531–2561, 2022.

[17] M. S. D. Cruz, and V. Sangeetha, *k-domination vertex connectivity in Internet of Things networks*, SN Computer Science, vol. 7, no. 2, p. 197, 2026.