International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(20s) 2026 pp. 1837-1864
https://doi.org/10.70917/ijcisim-2026-5048
Article

Secure Total Domination for Network Protection: Exact Guard Failure Thresholds and Reliability

Rayaprolu Viswaprasad[1], Konda Srinivasarao[2]

[1]Research Scholar, E-mail: rvprasad2012@gmail.com
[2]Professor of Mathematics, E-mail: raokonda@kanchiuniv.ac.in
SCSVMV, Kanchipuram, Tamil Nadu, India

Abstract: In secure total domination, every vertex must have a neighbouring guard. A guard must also be able to move to any unoccupied vertex while preserving total domination. We study guard failures on a fixed graph; a failed guard's vertex remains in the graph and may still need support. For a deployment D, let μ_D be the least number of guards in any open neighbourhood, and let $\tau_G(D)$ be the fewest failures that destroy security. We prove that $\tau_G(D) \in \{\mu_D - 1, \mu_D\}$ whenever $\mu_D \geq 1$. The main theorem characterises the smaller value through critical guarded neighbourhoods and gives an $O(n^4)$ algorithm with a certificate that can be checked. We then generate every minimum-cardinality destructive set in $O(n^5)$ time, so the exact first unreliability coefficient can be computed in polynomial time. A separate construction has exponentially many inclusion-minimal destructive sets, which shows the limit of this enumeration. For complete multipartite graphs, we prove that resilient secure total dominating sets are exactly the closed $(f + 2)$-tuple dominating sets, and derive exact capacity formulas. Earlier tools are also used to obtain reliability identities and universal-centre compositions. The proofs are supported by exhaustive finite checks. A bipartite example illustrates that two deployments of the same size may have different failure thresholds and different families of minimum destructive sets. Our review of the literature is limited in scope, and we do not claim global historical priority. These results may help in comparing proposed guard placements for network protection, monitoring and backup service planning. Before practical use, they must be validated with real network data and suitable operating constraints.
Keywords: secure total domination; guard failures; critical neighbourhoods; minimum destructive sets; tuple domination; graph reliability

1. Introduction

In a graph-based protection system, we distinguish a guard from the place where it is stationed. For example, when a controller, monitoring device or response unit fails, its station may still be accessible and require support. We therefore check domination for the surviving deployment in the original graph. Under total domination, each occupied vertex must also be supported by another occupied vertex. Secure total domination further requires a guard to be able to move to an unoccupied neighbour without leaving any vertex unsupported.

Prasad [14] introduces satellite-based secure domination parameters and discusses random failures and multiple disruptions. This leads us to ask how many guards in a given deployment can fail before secure total domination is lost, and how the resulting deterministic guarantee can be translated into a probability of successful operation. Answering these questions requires us to specify the failure sets considered. The expected number of surviving guards gives an average count; whether domination succeeds also depends on where those guards are located.

We adopt a precise failure model: a guard is removed from the deployment when it fails, but the graph and all its vertices remain unchanged. After failures, an unoccupied vertex may require a response. A surviving guard at a neighbouring vertex moves there, and the resulting deployment must be total dominating. We do not require that deployment to stay secure through an unlimited sequence of further responses, as this is a different dynamic problem.

We also impose no connectivity condition on the subgraph induced by the guards. Our conclusions are therefore tied to this protection model.

1.1 Earlier characterisations

Secure total domination is an established graph parameter [12]. Castillano, Ugbinada, Canoy, and Go [5, Theorem 2.5] characterise defenders using the internal and external private neighbours of a selected vertex. In Lemma 2.4, their union is denoted by $P_S(v)$. We retain $v \in N(u)$, as required by the definition of a defender. Although their result is stated for connected graphs, its local argument applies component by component to the graphs considered here. Lemma 2.4 therefore restates an attributed defender criterion; we do not claim it as new.

The local criterion tests a proposed response in a particular surviving deployment. Our question is instead how many guard failures are needed to make some response impossible. Theorem 4.1 adds an argument showing that, at the first possible failure level, every relevant obstruction arises from a minimum guarded neighbourhood. We can therefore restrict the failure sets that must be checked. Theorem 4.2 evaluates the criterion for a given deployment. Selecting a smallest secure total dominating set is a separate optimisation problem studied in the bounds and complexity literature [8].

Multiple domination provides another point of comparison. In a closed k-tuple dominating set, $|N[x] \cap D| \geq k$ at every vertex [4,11]; in a k-tuple total dominating set, $|N(x) \cap D| \geq k$ [10]. The local conditions necessary for resilience in this paper are precisely those of closed tuple domination with $k = f + 2$. We prove that, for complete multipartite graphs, these conditions are sufficient as well when guards fail and the survivors must respond. Thus the additional result in Theorem 5.5 is an equivalence between the two families of deployments. The capacity formula then expresses the optimisation in terms of part sizes. Some numerical values are already known in the tuple-domination literature. Kazemi [11, Theorem 6] studies closed k-tuple domination on complete k-partite graphs. The example after Theorem 2.2 of Cabrera Martínez [4] gives $\gamma_{\times k}\left(K_{k,k'}\right) = 2k - 1$ when $k' \geq k$. We identify these overlaps where the formulas occur.

The universal-centre classification also uses known structure. Go and Canoy [9, Theorem 4.2] characterise secure total dominating sets in a corona. With $K_1 \circ G = K_1 \vee G$, their result gives the cases with zero or one centre. Their result for an adjacent pair of universal vertices [9, Theorem 5.1] gives the two-centre case. Proposition 7.1 is a formal consequence of these results and upward closure. The subsequent reliability and failure-threshold formulas apply this classification to our guard-failure model.

Dohmen and Tittmann [7] examine domination reliability under independent component failures, including unequal failure probabilities. Dod [6, Section 5.4] examines total domination reliability. In Section 6, we apply standard methods of probability—state sums, inclusion-exclusion, monotonicity and conditioning—to the secure total domination event and cite the relevant sources. The graph-specific part is the structure of destructive guard-failure sets.

1.2 Recent work and the question studied here

Aita and Araki [1] prove sharp bounds for secure total domination on maximal outerplanar graphs. Araki and Aita [2] study a minimum secure total dominating set in proper interval graphs. Both papers appeared in 2024 and concern the choice or size of a deployment. Our main algorithm takes a deployment as input and determines its exact tolerance to arbitrary guard failures.

Boufelgha, Ahmia, and Guettiche [3] study secure 2-domination in a 2025 paper. Its support and replacement conditions differ from total domination at every vertex, so that parameter cannot be substituted for the one studied here. Lamprou, Sigalas, Vaxevanakis, and Zissimopoulos [13] study fault-tolerant total and partial positive influence domination. Their preprint appeared in 2025 and the journal article in 2026; these are two versions of one work. Their

total-domination variant requires several selected neighbours outside the selected set and support inside it, but does not use the defensive swap required here. These works give relevant context, while a careful comparison still needs to match the definitions directly.

1.3 Contributions and scope of the novelty assessment

The main result is the critical-neighbourhood characterisation in Theorem 4.1 and its exact evaluation for a fixed deployment in Theorem 4.2. The two-value threshold theorem reduces the problem to a single failure budget. Theorem 4.4 and Corollary 4.5 extend this by generating every minimum-cardinality destructive set, including when the threshold takes its upper value. Corollary 6.6 uses the enumeration to calculate the exact first unreliability coefficient. As Example 6.7 shows, this polynomial-size family can be much smaller than the family of all inclusion-minimal destructive sets, which can be exponential. A second structural contribution is the equivalence for complete multipartite graphs in Theorem 5.5. We present its numerical corollaries alongside the known tuple-domination results with which they overlap. The sections on probability and universal centres are applications; they claim neither new probability principles nor a new universal-centre classification.

Result	**Closest established ingredient**	**Role here**
Lemma 2.4	Private-neighbour defender rule [5]	Attributed local criterion
Theorem 3.1 and Lemma 3.4	Multiplicity and the response definition	Reduction to one failure budget
Theorems 4.1 and 4.2	Local defender rule [5]	Critical candidate family and exact evaluation
Theorem 4.4 and Corollary 4.5	Unique surviving supports and Theorem 4.1	All minimum destructive sets in polynomial time
Theorem 5.5	Closed tuple domination [4,11]	Equivalence with resilient secure deployment
Corollaries 5.6 and 5.7	Tuple-domination part counts [4,11]	Capacity calculation and exact special cases
Section 6	Domination reliability [6,7]	Standard identities with a computable leading term
Section 7	Corona and join results [9]	Derived universal-centre compositions

Table 1. Main results and the established results they use.

Contribution comparison. A result does not become new merely because its notation or failure interpretation changes. We distinguish results attributed to earlier work, formal consequences of those results and the additional arguments concerning failure sets. Among the sources we examined, we did not find the same critical-neighbourhood failure criterion under the same definitions and assumptions. This observation is limited to the literature reviewed and does not establish that the criterion is absent elsewhere. Appendix B describes the searches and the areas we could not resolve.

The model and threshold reduction are developed in Sections 2 and 3. Section 4 presents the critical certificate and algorithms, and Section 5 proves the multipartite equivalence and its consequences. Reliability applications and universal-centre compositions appear in Sections 6 and 7. Sections 8 and 9 give exact examples and finite checks. Section 10 discusses the limitations of the model and the questions that remain.

2. Definitions and basic structure

Throughout, $G = (V, E)$ denotes a finite simple undirected graph, with $n = |V| \geq 1$. The open neighbourhood of x is $N_G(x)$, and its closed neighbourhood is $N_G[x] = N_G(x) \cup \{x\}$. We omit the graph subscript when the graph is clear. The minimum and maximum degrees are $\delta(G)$ and $\Delta(G)$. A graph with an isolated vertex has no total dominating set. Our definitions include such graphs unless stated otherwise; an infeasible minimisation problem has value $+\infty$.

Definition 2.1. A set $S \subseteq V$ is a total dominating set if $N(x) \cap S \neq \varnothing$ for every $x \in V$. It is a secure total dominating set (STDS) if it is total dominating and, for each $u \in V \setminus S$, there is a vertex $v \in S \cap N(u)$ for which the set

$$S_{v \to u} = (S \setminus \{v\}) \cup \{u\} \tag{1}$$

is total dominating. We call v a defender of u relative to S. The minimum size of an STDS is the usual secure total domination number $\gamma_{st}(G)$.

Every total-domination condition uses open neighbourhoods. In particular, the newly occupied vertex u does not support itself. This point leads to several of the stronger conditions used below.

Definition 2.2. A deployment is a set $D \subseteq V$. A failure set is a set $F \subseteq D$, and the guards that remain occupy $D \setminus F$. For an integer $f \geq 0$, we say the deployment is *resilient to f guard failures* if $D \setminus F$ is an STDS for every $F \subseteq D$ satisfying $|F| \leq f$. Define

$$\gamma_{st}^{(f)}(G) = \min\{|D| : D \text{ is resilient to } f \text{ guard failures}\}. \tag{2}$$

Thus $\gamma_{st}^{(0)}(G) = \gamma_{st}(G)$. The position of a failed guard remains a vertex of G. It can still serve as a response location, and a surviving guard may move to it.

For each deployment, define

$$A_D(x) = N(x) \cap D, \qquad d_D(x) = |A_D(x)|, \qquad \mu_D = \min_{x \in V} d_D(x). \tag{3}$$

Definition 2.3. The destructive-failure threshold is the minimum number of failures that makes the deployment insecure:

$$\tau_G(D) = \min\{|F| : F \subseteq D,\ D \setminus F \text{ is not an STDS of } G\}. \tag{4}$$

This minimum exists because the empty surviving deployment cannot totally dominate a nonempty graph. If a deployment is already insecure, its threshold is zero. More generally, D is resilient to f failures exactly when $\tau_G(D) \geq f + 1$.

2.1 Local defence and monotonicity

For a total dominating set S and a guard $v \in S$, define the open private-neighbour set

$$P_S(v) = \{x \in V : N(x) \cap S = \{v\}\}. \tag{5}$$

Lemma 2.4 (local defence criterion; after Castillano et al. [5]). Suppose S is total dominating, $u \notin S$, and $v \in S \cap N(u)$. Then v defends u exactly when $P_S(v) \subseteq N(u)$.

Proof. Removing v leaves a vertex unsupported exactly when v was its only neighbour in S; these vertices form $P_S(v)$. For such a vertex x, adding u restores support exactly when $x \in N(u)$. Every vertex outside $P_S(v)$ keeps a supporting neighbour from the original set. The containment is therefore necessary and sufficient. In particular,

this includes $x = u$. If v was the only neighbour of u in S, then $u \in P_S(v)$ but $u \notin N(u)$, and the response fails. □

Lemma 2.5 (upward closure). If S is an STDS and $S \subseteq T \subseteq V$, then T is an STDS. If D is resilient to f failures and $D \subseteq D' \subseteq V$, then D' is resilient to f failures.

Proof. Total domination is preserved when vertices are added. Given $u \notin T$, the security of S provides a defender $v \in S$. The replacement set $T_{v \to u}$ contains $S_{v \to u}$ and is therefore also total dominating. For the second claim, let $F' \subseteq D'$ have size at most f. The secure set $D \setminus (F' \cap D)$ is contained in $D' \setminus F'$, so the first claim applies. □

Lemma 2.6 (two neighbours suffice). If $d_S(x) \geq 2$ holds for every vertex x, then S is an STDS. Also, every vertex in $S \cap N(u)$ defends each adjacent $u \notin S$.

Proof. Removing any one guard leaves every vertex with at least one supporting neighbour. Adding the response vertex cannot destroy total domination. □

Two neighbours are not necessary at every occupied vertex. For instance, the two vertices of K_2 form an STDS, and each has only one neighbour in the deployment.

2.2 Passive satellites

The single-universal-vertex case also follows from the corona characterisation of Go and Canoy [9]. Here is a direct proof, which allows any edges among the added vertices, all of which are unoccupied.

Proposition 2.7 (passive universal vertices). Let H be formed from G by adding a nonempty set U of vertices, each adjacent to every vertex in G. Edges inside U are unrestricted. For a deployment $D \subseteq V(G)$, D is an STDS of H exactly when it is an STDS of G.

Proof. Suppose D is an STDS of G. It has at least two vertices, so each new vertex has a neighbour in D. For an original response vertex, a defender in D gives an original total dominating set after the move. This set also supports every vertex of U. Now let the response vertex be $u \in U$ and move any $v \in D$ to u. The guard at u supports all original vertices. At least one original guard remains; it supports u and every other vertex in U. The resulting set is therefore total dominating in H.

Conversely, suppose D is an STDS of H. No guard lies in U, so D must totally dominate G. For an original response vertex $u \in V(G) \setminus D$, the defender and the replacement set are both contained in G. Total domination of the original vertices in H is therefore the same as total domination in G. The same defender works in G. □

The proposition also applies after any guards have failed. Hence passive universal vertices do not change $\tau_G(D)$ when the deployment is confined to the original graph. With this allocation rule, adding one or two satellites gives no separate redundancy benefit. Section 7 uses a different allocation rule in which satellites do affect protection.

3. The destructive-failure threshold

Theorem 3.1 (two possible threshold values). Let $D \subseteq V$. If $\mu_D = 0$, then $\tau_G(D) = 0$. If $\mu_D \geq 1$, then

$$\mu_D - 1 \leq \tau_G(D) \leq \mu_D, \qquad \text{hence} \qquad \tau_G(D) \in \{\mu_D - 1, \mu_D\}. \tag{6}$$

Proof. If $\mu_D = 0$, some vertex has no neighbouring guard, so D is not total dominating. Now suppose $\mu_D \geq 1$. Choose x with $d_D(x) = \mu_D$. If all guards in $A_D(x)$ fail, x has no support. Hence $\tau_G(D) \leq \mu_D$.

If $\mu_D \geq 2$ and $|F| \leq \mu_D - 2$, every vertex has at least $\mu_D - |F| \geq 2$ neighbours in $D \setminus F$. Lemma 2.6 then shows that the surviving deployment is secure. So no failure set smaller than $\mu_D - 1$ is destructive. When $\mu_D = 1$, the lower bound follows just from the threshold being nonnegative. □

This theorem describes the first possible ways to lose protection. With μ_D failures, all guards in a minimum guarded neighbourhood can fail. With $\mu_D - 1$ failures, total domination still holds, but a response might remove the only remaining support of another vertex. The next section gives the exact condition for this earlier failure.

Example 3.2 (both values occur). In K_n, let D have $d \geq 2$ vertices. Every set of at least two vertices is an STDS, so $\tau_{K_n}(D) = d - 1$. Here $\mu_D = d - 1$, and the upper value occurs. In $K_{3,3}$, choose two vertices from each part. Then $\mu_D = 2$. If one guard in either part fails, an outside vertex in the opposite part has only one possible supporting guard. That guard cannot move to it and still leave it supported. Thus $\tau_G(D) = 1 = \mu_D - 1$.

Corollary 3.3 (automatic resilience ranges). Suppose $\mu_D \geq 1$. The deployment is resilient for every budget $f \leq \mu_D - 2$ and fails to be resilient for every budget $f \geq \mu_D$. Only the budget $f = \mu_D - 1$ needs a further security test.

Proof. This follows from Definition 2.3 and Theorem 3.1. The first range is empty if $\mu_D = 1$. □

3.1 Responses to failed original guards

Lemma 3.4 (responses to failed original guards). Suppose $d_D(x) \geq f + 1$ for every $x \in V$. Let $F \subseteq D$ satisfy $|F| \leq f$, and take $u \in F$. Then u has a defender in $D \setminus F$. In fact, every surviving neighbouring guard is a defender.

Proof. Since $u \notin N(u)$, at most $|F| - 1 \leq f - 1$ guards from $A_D(u)$ have failed. So u has at least two surviving neighbours in the deployment. Choose any one of them, say v, and define

$$F' = (F \setminus \{u\}) \cup \{v\}. \tag{7}$$

The set F' has size $|F| \leq f$, and the replacement deployment is $D \setminus F'$. Every vertex still has at least $f + 1 - |F'| \geq 1$ supporting neighbours, so the replacement is total dominating. □

By this lemma, the borderline failure case need only be tested at vertices in $V \setminus D$. A failed original guard lies outside the surviving deployment and may need a response, but restoring it exchanges one failed guard for another. This exchange within the original deployment is unavailable for a vertex that was unoccupied from the start.

4. Critical-neighbourhood certificates

Assume $\mu = \mu_D \geq 1$. Call x a *critical vertex* when $d_D(x) = \mu$. Define the family

$$\mathcal{F}_D = \{A_D(x) \setminus \{v\} : d_D(x) = \mu,\ v \in A_D(x)\}. \tag{8}$$

Each member has size $\mu - 1$, and there are at most $n\mu$ members before duplicates are removed. For $F \in \mathcal{F}_D$ and $u \in V \setminus D$, define the set of blocked guard labels

$$B_D(u, F) = \{v \in D \setminus F : \exists x \in V \setminus N(u),\ A_D(x) = F \cup \{v\}\}. \tag{9}$$

Note that u belongs to $V \setminus N(u)$. This is needed because, if an outside response vertex has only one surviving neighbouring guard, the vertex itself must be counted as an obstruction. The possible defenders of u are precisely the guards in $A_D(u) \setminus F$. The set $B_D(u, F)$ records guard labels with a private-support obstruction, although a recorded guard need not be adjacent to u. The test below applies these labels to the possible defenders.

Theorem 4.1 (critical-neighbourhood characterisation). Suppose $\mu_D \geq 1$. The following are equivalent:

1. $\tau_G(D) = \mu_D - 1$.
2. There are $F \in \mathcal{F}_D$ and $u \in V \setminus D$ such that

$$A_D(u) \setminus F \subseteq B_D(u, F). \tag{10}$$

If no such pair exists, then $\tau_G(D) = \mu_D$.

Proof. Take $F \subseteq D$ with size $\mu - 1$, and write $S = D \setminus F$. Every vertex still has a neighbouring guard, so S is total dominating. Let $u \in V \setminus D$ be an initially unoccupied vertex and $v \in A_D(u) \setminus F$ a possible defender. By Lemma 2.4, v fails as a defender exactly when some $x \notin N(u)$ satisfies

$$A_D(x) \setminus F = \{v\}. \tag{11}$$

Since $|A_D(x)| \geq \mu$ and $|F| = \mu - 1$, equation (11) implies $A_D(x) = F \cup \{v\}$ and $d_D(x) = \mu$. Conversely, this equality means x loses all support if v moves to u. Thus a possible defender is invalid exactly when it belongs to $B_D(u, F)$. Every possible defender of u is blocked exactly when (10) holds.

If (10) holds for one of the listed pairs, the surviving deployment remains total dominating but cannot respond to u. It is therefore insecure, and Theorem 3.1 gives $\tau_G(D) = \mu - 1$.

Conversely, let F be a destructive set of size $\mu - 1$. The surviving deployment is still total dominating. Lemma 3.4, applied with $f = \mu - 1$, shows that a failed vertex in F cannot cause the loss of security. Therefore some $u \in V \setminus D$ has no valid defender. It has at least one possible defender because $|A_D(u)| \geq \mu > |F|$. Choose one, v, and an obstructing vertex x from (11). Then $d_D(x) = \mu$ and $F = A_D(x) \setminus \{v\}$, so $F \in \mathcal{F}_D$. All possible defenders are blocked, and hence (10) holds. If no listed obstruction exists, Theorem 3.1 leaves only the value μ. □

If $\mu = 1$, the empty set is the only member of the candidate family, and the theorem checks whether D is already secure. If $D = V$, there are no initially unoccupied vertices and the obstruction cannot occur. Both cases are covered by the theorem.

4.1 A certificate explaining the failure

For each $v \in A_D(u) \setminus F$, the pair (F, u) can be supplemented by an obstructing vertex x_v. The certificate consists of

$$x_v \notin N(u), \qquad A_D(x_v) = F \cup \{v\}. \tag{12}$$

Once F fails, moving v would leave x_v without support. Each failed response can be checked using the adjacency information. The certificate lists the failed guards, the response vertex and the vertex that would be left unsupported by each possible response.

Theorem 4.2 (polynomial-time threshold evaluation). Given the adjacency matrix of G and a deployment D, we can compute $\tau_G(D)$ in $O(n^4)$ elementary operations and $O(n^2)$ working space. If the value is $\mu_D - 1$, the

algorithm also returns a certificate as in (12). Thus the resilience of a fixed deployment can be decided in polynomial time, even if the failure budget is part of the input.

Proof. Compute the sets $A_D(x)$ and μ_D in $O(n^2)$ time. Return zero if $\mu_D = 0$. Otherwise, for each ordered pair (x, v) with $d_D(x) = \mu_D$ and $v \in A_D(x)$, form $F = A_D(x) \setminus \{v\}$. Duplicates may be kept; there are at most n^2 candidates.

Fix one candidate F and scan the vertices y. Check whether $A_D(y)$ contains F plus exactly one other element. For each critical vertex that passes this check, record the extra element as a label b_y. With Boolean membership arrays, this takes $O(n^2)$ time. For each $u \in V \setminus D$, scan the labelled vertices y that are not adjacent to u and mark their labels as blocked. Then check whether any defender in $A_D(u) \setminus F$ remains unmarked. The work for this candidate, summed over all u, is $O(n^2)$.

If all possible defenders of a response vertex are marked as blocked, return $\mu_D - 1$ together with F, that response vertex u, and one labelled witness for each blocked defender. If every candidate passes the test, return μ_D. Theorem 4.1 proves correctness. There are at most n^2 candidates, each requiring $O(n^2)$ work, giving the claimed time. Candidates are processed one at a time, so the adjacency and neighbourhood arrays determine the working space. Finally, compare the returned threshold with $f + 1$. □

The input to this result is a given deployment, whose threshold is evaluated by the algorithm. It does not provide a polynomial-time method for finding a minimum deployment in an arbitrary graph. This distinction matters because the optimisation problem already includes conventional secure total domination when $f = 0$ [8].

4.2 Recognition and independent certificate checks

The stated operation bounds use an adjacency matrix whose entries can be accessed in constant time. Subsets are stored as Boolean membership arrays, and a neighbourhood scan examines at most n entries. The following membership test is used to evaluate the threshold and to filter candidate minimum failure sets.

Proposition 4.3 (recognition and verification). Given $S \subseteq V$, we can decide whether S is an STDS in $O(n^2)$ time, using $O(n)$ additional working space beyond the adjacency matrix. If S is insecure, the procedure returns either an unsupported vertex or an unoccupied response vertex, with one blocking vertex for each possible defender. A certificate for the lower threshold in Theorem 4.1 can also be checked independently in $O(n^2)$ time.

Proof. First scan each neighbourhood and count the surviving guards. If the count is zero, the vertex certifies failure of total domination. Otherwise, when x has exactly one neighbour in S, store that guard as $\mathrm{sole}(x)$; leave the label undefined when there are at least two. For each $u \notin S$, mark v as blocked whenever a labelled vertex $x \notin N(u)$ satisfies $\mathrm{sole}(x) = v$. By Lemma 2.4, a neighbouring guard defends u exactly when it is not blocked. Two scans of at most n entries are made for each response vertex. The initial scan uses at most n^2 matrix entries, giving the stated time and space bounds. Retaining a blocking vertex for each marked label explains why the response fails.

To check a lower-threshold certificate, keep a generating anchor (x_0, v_0) along with F, u, and the witnesses x_v in (12). Recompute $A_D(x)$ and μ_D, then verify $d_D(x_0) = \mu_D$, $v_0 \in A_D(x_0)$, $F = A_D(x_0) \setminus \{v_0\}$, and $u \notin D$. For each $v \in A_D(u) \setminus F$, check $x_v \notin N(u)$ and $A_D(x_v) = F \cup \{v\}$. There are at most n such checks, each using at most n entries. They certify destruction with $\mu_D - 1$ failures, and Theorem 3.1 shows that fewer failures cannot suffice. □

Algorithm 1. Recognising a surviving deployment. The value none means that a label is undefined. A false result includes the witnesses stored in the marked arrays.

```
        STDS_MEMBER(G, S)
  for each vertex x
    L := N(x) intersect S
    if L is empty: return false, unsupported vertex x
    sole[x] := the member of L if |L| = 1; otherwise none
  for each vertex u outside S
    clear blocked and witness arrays
    for each vertex x
      if sole[x] is defined and x is not adjacent to u
        v := sole[x]
        blocked[v] := true
        witness[v] := x
    if every v in N(u) intersect S is blocked
      return false, response u, witness array
  return true
```

The nonadjacent scan includes $x = u$. A simple graph has no loop at u, so a response vertex with only one surviving neighbour itself blocks that sole guard. The algorithm would be incorrect if this diagonal case were omitted.

Algorithm 2. Exact threshold for a fixed deployment. Algorithm 1 checks each candidate in $O(n^2)$ time. This version gives the same result as the blocked-label procedure in the proof of Theorem 4.2.

```
        EXACT_THRESHOLD(G, D)
  compute A_D(x) for every x and mu := min_x |A_D(x)|
  if mu = 0
    return 0, empty failure set, an unsupported vertex
  for each vertex x with |A_D(x)| = mu
    for each v in A_D(x)
      F := A_D(x) minus {v}
      result := STDS_MEMBER(G, D minus F)
      if result is false
        return mu - 1, F, anchor (x,v), result witnesses
  choose x with |A_D(x)| = mu
  return mu, F := A_D(x), unsupported vertex x after F
```

Total domination survives every candidate failure set of size $\mu_D - 1$. By Lemma 3.4, any undefendable vertex returned must also be in $V \setminus D$, even though Algorithm 1 checks every vertex outside the surviving deployment. The data returned for the lower threshold therefore have the form required by Theorem 4.1. Retaining duplicate candidates does not change the $O(n^4)$ time bound.

In the upper-threshold branch, the returned failure set destroys total support and is therefore destructive. Its size is minimum only when the earlier checks have also ruled out a lower-threshold obstruction. A destructive set of size μ_D by itself does not certify optimality.

4.3 All minimum destructive failure sets

The threshold algorithm returns one destructive set. Reliability calculations need to count all destructive sets of the smallest possible size. Define

$$\mathcal{M}_G(D) = \{F \subseteq D : |F| = \tau_G(D),\ D \setminus F \text{ is not an STDS}\}. \tag{13}$$

We call this the family of minimum-cardinality destructive sets. Every member is inclusion-minimal: any proper subset has fewer than the threshold number of failures and so cannot be destructive. The family does not include inclusion-minimal destructive sets of larger size. This distinction will matter below.

Suppose $\mu = \mu_D \geq 1$. Alongside the critical family $\mathcal{F}_D$ from (8), define three families, each consisting of sets of size μ:

$$\mathcal{H}_0 = \{A_D(x) : x \in V,\ d_D(x) = \mu\}, \tag{14}$$

$$\mathcal{H}_1 = \bigcup_{\substack{x \in V \\ d_D(x) = \mu}} \bigcup_{v \in A_D(x)} \{(A_D(x) \setminus \{v\}) \cup \{w\} : w \in D \setminus A_D(x)\}. \tag{15}$$

$$\mathcal{H}_2 = \{A_D(x) \setminus \{v\} : x \in V,\ d_D(x) = \mu + 1,\ v \in A_D(x)\}. \tag{16}$$

Let $\mathcal{H}_D = \mathcal{H}_0 \cup \mathcal{H}_1 \cup \mathcal{H}_2$ be their union, with duplicate subsets kept only once. Different generators may give the same set.

Theorem 4.4 (generating all minimum destructive sets). If $\tau_G(D) = 0$, then $\mathcal{M}_G(D) = \{\varnothing\}$. If $\mu_D \geq 1$ and $\tau_G(D) = \mu_D - 1$, then

$$\mathcal{M}_G(D) = \{F \in \mathcal{F}_D : D \setminus F \text{ is not an STDS}\}. \tag{17}$$

If $\mu_D \geq 1$ and $\tau_G(D) = \mu_D$, then

$$\mathcal{M}_G(D) = \{F \in \mathcal{H}_D : D \setminus F \text{ is not an STDS}\}. \tag{18}$$

Thus, when the threshold has its upper value, guarded neighbourhoods of sizes μ_D and $\mu_D + 1$ generate every minimum destructive set.

Proof. If the threshold is zero, the empty set is the only set of size zero. For the lower threshold, the reverse direction of Theorem 4.1 applies to every destructive set of size $\mu - 1$. Therefore each such set is in $\mathcal{F}_D$, and filtering that family for insecure survivors gives exactly the minimum destructive sets.

Now suppose $\tau_G(D) = \mu$, and take $F \in \mathcal{M}_G(D)$. Write $S = D \setminus F$. If S is not total dominating, some x satisfies $A_D(x) \subseteq F$. Since $d_D(x) \geq \mu = |F|$, equality must hold: $A_D(x) = F$ and $d_D(x) = \mu$. Hence $F \in \mathcal{H}_0$.

Otherwise, S is total dominating but insecure. Some $u \notin S$ has no defender. By total domination, choose $v \in N(u) \cap S$. Lemma 2.4 gives a vertex $x \notin N(u)$ with $N(x) \cap S = \{v\}$. Therefore

$$A_D(x) \setminus F = \{v\}, \qquad \mu \leq d_D(x) \leq |F| + 1 = \mu + 1. \tag{19}$$

If $d_D(x) = \mu + 1$, every guard in $A_D(x) \setminus \{v\}$ must fail, and there are exactly μ such guards. Hence $F = A_D(x) \setminus \{v\} \in \mathcal{H}_2$. If $d_D(x) = \mu$, $A_D(x) \setminus \{v\}$ accounts for $\mu - 1$ members of F. The one remaining member w cannot be in $A_D(x)$, since v survives. It follows that $F = (A_D(x) \setminus \{v\}) \cup \{w\} \in \mathcal{H}_1$.

So every minimum destructive set is generated by the stated families. Conversely, every member of $\mathcal{H}_D$ has size μ. Under the assumption $\tau_G(D) = \mu$, each destructive member therefore has minimum cardinality. The same size argument applies to the lower-threshold family. □

When the threshold is at its upper value, u in this proof may be a failed original guard. Lemma 3.4 rules out such a witness only when at most $\mu_D - 1$ guards fail. Therefore, when filtering $\mathcal{H}_D$, we must check every vertex outside $D \setminus F$. Algorithm 1 does this.

Corollary 4.5 (polynomial enumeration and counting). Let $d = |D|$. Let q_0 be the number of vertices with $d_D(x) = \mu_D$, and let q_1 count those with $d_D(x) = \mu_D + 1$. For $\mu_D \geq 1$,

$$\begin{aligned} |\mathcal{M}_G(D)| &\leq q_0\mu_D \quad \text{if } \tau_G(D) = \mu_D - 1, \\ |\mathcal{M}_G(D)| &\leq q_0\left(1 + \mu_D(d - \mu_D)\right) + q_1(\mu_D + 1) \quad \text{if } \tau_G(D) = \mu_D. \end{aligned} \tag{20}$$

We can list and count all distinct members of $\mathcal{M}_G(D)$ in $O(n^5)$ elementary operations, using $O(n^4)$ Boolean entries to store candidates, in addition to the adjacency matrix and oracle workspace. The lower-threshold case takes $O(n^4)$ time. These are upper bounds; we do not claim that the exponents are optimal.

Proof. Each of the q_0 critical vertices generates at most μ_D candidates for the lower branch. For the upper branch, it generates one set in $\mathcal{H}_0$ and at most $\mu_D(d - \mu_D)$ in $\mathcal{H}_1$. Each of the q_1 vertices counted above generates at most $\mu_D + 1$ sets in $\mathcal{H}_2$. This proves (20) before deduplication, and hence also after it. The upper family has $O(nd^2) = O(n^3)$ candidates.

First compute $\tau_G(D)$ using Theorem 4.2. If the threshold is zero, return the family containing only the empty set. Otherwise generate the relevant candidates and store each as an n-entry Boolean vector. Use lexicographic radix sorting on the n binary coordinates to remove duplicate vectors. For K candidates, sorting takes $O(nK)$ operations and $O(nK)$ storage and does not depend on hash-table performance. For each remaining candidate failure set, remove that set from the deployment and apply Algorithm 1 to the survivors. Theorem 4.4 ensures that this procedure retains exactly the required family. Filtering requires $O(Kn^2)$ operations: $O(n^5)$ for the upper branch and $O(n^4)$ for the lower branch. At most $O(n^4)$ storage is needed for the candidates. Counting the distinct vectors that remain gives the exact number of sets. □

Algorithm 3. Listing all minimum destructive sets. Remove duplicate failure vectors before counting the sets or summing their reliability weights.

```
    MINIMUM_DESTRUCTIVE_SETS(G, D)
t := EXACT_THRESHOLD(G, D).threshold
if t = 0: return the family containing only the empty set
compute mu and all guarded neighbourhoods A_D(x)
if t = mu - 1
  generate the family F_D from equation (8)
else
  generate H_0, H_1, H_2 from equations (14) to (16)
sort the Boolean failure vectors and remove duplicates
return each generated F for which
    STDS_MEMBER(G, D minus F) is false
```

The result listing every minimum destructive set extends threshold evaluation without enumerating every inclusion-minimal destructive set. Section 6 uses the enumeration to obtain the exact first unreliability coefficient. It also gives a graph family for which the larger family of inclusion-minimal destructive sets has exponential size.

5. Structural bounds and exact graph classes

5.1 Feasibility and forced guards

Theorem 5.1 Feasibility criterion. A deployment resilient to f guard failures exists if and only if

$$\delta(G) \geq f + 1. \tag{21}$$

Proof. Suppose some vertex has at most f neighbouring guards. All of those guards can fail, so this deployment is not feasible. Thus every feasible deployment must satisfy $d_D(x) \geq f + 1$ for every x, which gives (21). Conversely, under (21), choose $D = V$. After any set of at most f failures, every vertex still has a neighbouring guard, so total domination is preserved. Every vertex that is unoccupied after the failures was originally a guard. Lemma 3.4 therefore gives a valid defender. □

It follows that $\tau_G(V) = \delta(G)$ whenever G has no isolated vertices. Under this model, no choice of guards can withstand more than $\delta(G) - 1$ arbitrary guard failures.

Theorem 5.2 Local necessities and a sufficient margin. If D is resilient to f failures, then

$$d_D(x) \geq f + 1 \quad (x \in D), \qquad d_D(u) \geq f + 2 \quad (u \in V \setminus D). \tag{22}$$

Conversely, the stronger uniform condition $d_D(x) \geq f + 2$ for every $x \in V$ is sufficient for resilience to f failures.

Proof. If the supports of a selected vertex are all removed, resilience fails. This gives the first inequality, and the same argument gives the weaker bound $f + 1$ at every vertex. Now take an initially unoccupied vertex u. Suppose $d_D(u) = f + 1$. Fail any f of its neighbouring guards. The one remaining guard cannot move to u: after the move, u would have no neighbouring guard. This argument also applies when $f = 0$. Thus an initially unoccupied vertex must have at least $f + 2$ neighbouring guards. Conversely, after any allowed failures, each vertex still has at least two neighbouring guards. Resilience then follows from Lemma 2.6. □

The two conditions in (22) alone are not sufficient for resilience on every graph. For example, consider the path $1,2,3,4,5$, with $D = \{1,2,4,5\}$ and $f = 0$. Every selected vertex has a selected neighbour, and the only outside vertex, namely 3, has two neighbouring guards. So (22) holds. But if guard 2 moves to 3, vertex 1 loses all support; if guard 4 moves to 3, vertex 5 loses all support. Hence D is insecure. Theorem 4.1 gives the additional relational condition needed here.

Corollary 5.3 Cardinality and forced-vertex bounds. Whenever a finite feasible deployment exists,

$$\gamma_{st}^{(f)}(G) \geq \max\left\{f + 2, \left\lceil \frac{n(f+2)}{\Delta(G)+1} \right\rceil, |N[L_f]|\right\}, \tag{23}$$

where $\lceil t \rceil$ is the least integer at least t, $L_f = \{x \in V : \deg_G(x) = f + 1\}$, and $N[L_f] = \bigcup_{x \in L_f} N[x]$.

Proof. Every selected vertex needs at least $f + 1$ other selected neighbours. Thus $|D| \geq f + 2$. Summing the bounds in (22) over all vertices gives

$$\sum_{x \in V} d_D(x) \geq |D|(f+1) + (n - |D|)(f+2) = n(f+2) - |D|. \tag{24}$$

The same sum equals $\sum_{v \in D} \deg_G(v) \leq \Delta(G)|D|$, proving the second bound. Finally, a vertex of degree $f + 1$ must itself be selected, since an unselected vertex would need $f + 2$ neighbouring guards. All its neighbours must also be selected to give it the required $f + 1$ supports. Therefore $N[L_f] \subseteq D$. □

Corollary 5.4 Regular graphs at their maximum budget. If G is a nonempty r-regular graph with $r \geq 1$, then $\gamma_{st}^{(r-1)}(G) = n$. In particular, $\gamma_{st}^{(1)}(C_n) = n$ for every $n \geq 3$, and no path of order at least two admits resilience to one guard failure.

Proof. When $f = r - 1$, every vertex belongs to L_f. Corollary 5.3 therefore forces $D = V$, and Theorem 5.1 shows that this deployment is feasible. A cycle is 2-regular, while every nontrivial path has minimum degree one. □

We compare these bounds with standard multiple domination [4,10,11]. Let $\gamma_{\times k}(G)$ be the minimum size of a closed k-tuple dominating set, and let $\gamma_{\times k,t}(G)$ be the minimum size of a k-tuple total dominating set. Equation (22) is equivalent to $|N[x] \cap D| \geq f + 2$ for every x: a selected vertex contributes itself to its closed neighbourhood. Conversely, closed $(f + 2)$-tuple domination gives exactly the two inequalities in (22). It follows that

$$\gamma_{\times(f+1),t}(G) \leq \gamma_{\times(f+2)}(G) \leq \gamma_{st}^{(f)}(G) \leq \gamma_{\times(f+2),t}(G). \tag{25}$$

We use extended-real values if any required set does not exist. The middle lower bound includes the vertex itself in the closed-neighbourhood count, so it is at least as strong as the first lower bound. In the path example above, the neighbourhoods of the two leaves force vertices 1, 2, 4, and 5 into every closed double dominating set. Hence $\gamma_{\times 2}(P_5) = 4$. This is the only possible set of size four, and it is insecure. The full vertex set is secure, so $\gamma_{st}(P_5) = 5$. Thus the middle inequality can be strict.

The tuple-total domination quantity on the right may be infinite even when the resilient secure-total parameter is finite. For example, $D = V$ attains the maximum feasible budget in an $(f + 1)$-regular graph.

5.2 Complete multipartite graphs

Let $G = K_{n_1,\ldots,n_r}$, where $r \geq 2$ and each $n_i \geq 1$. Write the parts as $V_1, \ldots, V_r$, set $x_i = |D \cap V_i|$, and let $d = \sum_i x_i$. Each vertex in V_i has exactly $d - x_i$ neighbouring guards.

Theorem 5.5 Resilience and closed tuple domination on complete multipartite graphs. For $G = K_{n_1,\ldots,n_r}$, the deployment D is resilient to f failures if and only if it is a closed $(f + 2)$-tuple dominating set. Equivalently,

$$\begin{aligned} d - x_i \geq f + 1 \quad \text{for every } i, \qquad d - x_i \\ \geq f + 2 \quad \text{whenever } x_i < n_i. \end{aligned} \tag{26}$$

Proof. Theorem 5.2 gives the necessary conditions. To prove sufficiency, take any allowed failure set F and write $S = D \setminus F$, $s = |S|$, and $y_i = |S \cap V_i|$. The first set of inequalities in (26) ensures that S totally dominates the graph.

Now take an outside response vertex $u \in V_i \setminus S$. If $x_i < n_i$, at most f of the $d - x_i$ original guards outside V_i can have failed, so $s - y_i \geq 2$. If $x_i = n_i$, at least one failure was inside V_i. Therefore at most $f - 1$ failures were outside it, and again $s - y_i \geq 2$. Choose any surviving guard v outside V_i, say in V_j. Moving v to u reduces the number of guards outside V_i by one, increases the number outside V_j by one, and leaves the count for every other part unchanged. Every part still has at least one external guard. Thus the replacement is total dominating. □

Closed $(f + 2)$-tuple domination has exactly the part conditions in (26), by the definition of $N[x]$. The equivalence we have proved applies to every deployment in this graph family, not only to a minimum deployment. In particular,

$$\gamma_{st}^{(f)}(K_{n_1,\ldots,n_r}) = \gamma_{\times(f+2)}(K_{n_1,\ldots,n_r}). \tag{27}$$

The additional structural result is that, for this graph class, these local part counts guarantee the failure-and-response condition. The tuple-domination counts themselves are established notions. Here, part counts track how support changes after a response. Consequently, the obstruction in the path example does not occur.

Corollary 5.6 Exact capacity formula. For an integer $d \geq f + 2$, define

$$\begin{aligned} c_i(d) &= n_i \quad \text{if } n_i \leq d - f - 1, \\ c_i(d) &= d - f - 2 \quad \text{if } n_i > d - f - 1. \end{aligned} \tag{28}$$

Then

$$\gamma_{st}^{(f)}(K_{n_1,\ldots,n_r}) = \min\left\{d \in \{f+2, \ldots, n\} : \sum_{i=1}^{r} c_i(d) \geq d\right\}, \tag{29}$$

and the value is $+\infty$ if the set displayed here is empty. We can evaluate the formula by checking $O(nr)$ scalar inequalities, and the procedure also constructs an optimal deployment.

Proof. Fix d. If a part is not full, (26) requires $x_i \leq d - f - 2$. A full part is allowed exactly when $n_i \leq d - f - 1$. If $n_i \leq d - f - 1$, every integer from zero to n_i is permitted. At the boundary $n_i = d - f - 1$, the largest incomplete value is $n_i - 1 = d - f - 2$. Otherwise, the permitted values run from zero to $d - f - 2$. Thus the allowed values of x_i are exactly the integers in $[0, c_i(d)]$.

There are nonnegative integer choices within these upper bounds that sum to d exactly when the upper bounds sum to at least d. To construct such a choice, fill each part up to its capacity until d guards have been assigned. Theorem 5.5 then gives feasibility. Checking all possible values of d proves (29) and the stated operation bound. □

Corollary 5.7 Complete and complete bipartite values. For $n \geq 2$,

$$\gamma_{st}^{(f)}(K_n) = f + 2 \quad \text{if } 0 \leq f \leq n - 2, \tag{30}$$

and the value is infinite otherwise. For $1 \leq a \leq b$,

$$\begin{aligned} \gamma_{st}^{(f)}(K_{a,b}) &= \infty \quad \text{if } a \leq f, \\ \gamma_{st}^{(f)}(K_{a,b}) &= a + b \quad \text{if } a = f + 1, \\ \gamma_{st}^{(f)}(K_{a,b}) &= 2f + 3 \quad \text{if } a = f + 2, \\ \gamma_{st}^{(f)}(K_{a,b}) &= 2f + 4 \quad \text{if } a \geq f + 3. \end{aligned} \tag{31}$$

Proof. In a complete graph, a set is secure total dominating exactly when it has at least two vertices. After f failures, at least two guards must remain, which proves (30). For the bipartite graph, let x and y be the numbers selected from the two parts. Theorem 5.5 requires $x, y \geq f + 1$. It also requires $y \geq f + 2$ if $x < a$, and $x \geq f + 2$ if $y < b$.

If $a \leq f$, no feasible deployment exists. If $a = f + 1$, necessarily $x = a$. Leaving out a vertex from the other part would require $x \geq f + 2$, so $y = b$ is also necessary. If $a = f + 2$, we obtain a feasible deployment of size $2f + 3$ by selecting all a vertices in the first part and $f + 1$ in the second. A smaller set would require $x = y = f + 1$, leaving both parts incomplete and violating the conditions. Finally, let $a, b \geq f + 3$. A feasible

deployment is obtained by selecting $f+2$ vertices from each part. In any feasible set of size at most $2f+3$, one count would have to equal $f+1$. This forces the other part to be full, and hence to contain at least $f+3$ vertices. The total size would then be at least $2f+4$, a contradiction. □

Relation of the exact values to known tuple domination formulas

The capacity formula follows from Theorem 5.5 and the integer allocation of guards among the part capacities. Our literature check did not settle whether it was already known as a closed tuple-domination formula for arbitrary part sizes. When $r=f+2$, the parameter matches closed k-tuple domination on a complete k-partite graph, with $k=f+2$; Kazemi [11, Theorem 6] gives exact values for that setting. In particular, the cases with $f=0$ for complete bipartite graphs reproduce the closed double-domination values. The complete-graph value $f+2$ is also the immediate closed tuple-domination value.

For the third branch of (31), Cabrera Martínez [4, the example following Theorem 2.2] already gives $\gamma_{\times k}\left(K_{k,k'}\right)=2k-1$ for $k'\geq k$. Substituting $k=f+2$ gives the numerical value $2f+3$ when $a=f+2$. The resilience interpretation comes from Theorem 5.5. These identifications mean that this numerical value should not be counted again as a separate discovery under a different parameter name.

5.3 Independent components

Proposition 5.8 Component decomposition. Let G be the disjoint union of nonempty graphs $G_1,\ldots,G_k$, and put $D_i=D\cap V(G_i)$. Then

$$\tau_G(D)=\min_i \tau_{G_i}(D_i), \qquad \gamma_{st}^{(f)}(G)=\sum_i \gamma_{st}^{(f)}(G_i). \tag{32}$$

Proof. Guard support and any possible response stay within one component. Therefore, a surviving deployment is secure in G exactly when its restriction is secure in every component. A smallest destructive set can be placed entirely in a component with the smallest threshold. Global resilience also holds exactly when each component deployment is resilient to the full budget f, because the adversary can use all failures in any one component. Minimising the sum of the component deployment sizes proves the second equality, with the usual convention for infinity. □

6. Reliability under independent heterogeneous failures

For each $v\in D$, let $p_v\in[0,1]$ be its guard-failure probability. Failures are independent. No failure probability is assigned to an unoccupied vertex, and graph vertices do not disappear. Let S be the random set of guards that survive. Define

$$R_{st}(G,D;\mathrm{p})=\Pr(S \text{ is an STDS of } G). \tag{33}$$

The event in (33) requires both total support and a valid defensive relocation. It is therefore stronger than the event that S is total dominating. The state-sum and inclusion-exclusion arguments below apply established reliability methods [6,7] to this particular monotone event. We do not claim the elementary probability steps themselves as new.

Proposition 6.1 Exact state sum and monotonicity. The reliability is

$$R_{st}(G,D;\mathrm{p})=\sum_{\substack{S\subseteq D\\ S \text{ is an STDS}}}\prod_{v\in S}(1-p_v)\prod_{v\in D\setminus S}p_v. \tag{34}$$

It is a polynomial affine in each individual p_v, and it is nonincreasing in each failure probability.

Proof. Each surviving set has the product probability appearing in (34), and these states form a disjoint partition. Each product contains either p_v or $1 - p_v$ once, which proves affinity in each variable. For monotonicity, condition on the states of all other guards. If guard v survives, the available set is larger; Lemma 2.5 says that adding guards to a secure set preserves security. Increasing its failure probability therefore cannot increase the success probability. □

6.1 Minimal destructive sets

Let $\mathcal{C}_G(D)$ be the family of inclusion-minimal sets $C \subseteq D$ for which $D \setminus C$ is insecure. If D is already insecure, this family contains the empty set. Otherwise all its members are nonempty. Upward closure and finiteness imply that a failure set is destructive if and only if it contains a member of $\mathcal{C}_G(D)$.

Theorem 6.2 Exact cut representation and elementary bounds. For independent failures,

$$1 - R_{st}(G, D; \mathrm{p}) = \sum_{\varnothing \neq \mathcal{J} \subseteq \mathcal{C}_G(D)} (-1)^{|\mathcal{J}|+1} \prod_{v \in \bigcup_{C \in \mathcal{J}} C} p_v. \tag{35}$$

If D is secure, then

$$\max_{C \in \mathcal{C}_G(D)} \prod_{v \in C} p_v \leq 1 - R_{st}(G, D; \mathrm{p}) \leq \min\left\{1, \sum_{C \in \mathcal{C}_G(D)} \prod_{v \in C} p_v\right\}. \tag{36}$$

Proof. For each minimal destructive set C, let E_C be the event that every guard in that set fails. The protection property fails exactly when at least one of these events occurs. An intersection of events for a subfamily requires every guard in the union of its sets to fail. Independence gives the product in (35), and inclusion-exclusion gives the equality. The largest individual event probability gives the lower bound. The union bound gives the upper bound. □

Example 6.7 shows that the number of inclusion-minimal destructive sets can be exponential. Thus (35) is an exact representation, but it does not claim that exact reliability can always be computed in polynomial time. Corollary 4.5 lists every destructive set of minimum cardinality; it does not list the full family in (35).

Corollary 6.3 Small failure probabilities. Suppose D is secure, put $t = \tau_G(D) \geq 1$, and let $\mathcal{C}_t = \{C \in \mathcal{C}_G(D) : |C| = t\}$. If $p_v = \varepsilon a_v$ with fixed positive constants a_v and $\varepsilon \downarrow 0$, then

$$1 - R_{st}(G, D; \varepsilon \mathrm{a}) = \varepsilon^t \sum_{C \in \mathcal{C}_t} \prod_{v \in C} a_v + O(\varepsilon^{t+1}). \tag{37}$$

For homogeneous failure probabilities, $p_v = p$, the leading coefficient is $|\mathcal{C}_t|$.

Proof. Every minimal destructive set has at least t members. A single-event term for a set of size t has degree exactly t; all other single-event terms have degree at least $t + 1$. For two distinct minimal destructive sets of size at least t, the union has size at least $t + 1$ unless one set contains the other. Minimality excludes containment. Therefore every intersection term involving two or more distinct sets has degree at least $t + 1$. Substitution in the finite sum (35) proves (37). □

The deterministic threshold gives the first possible power of the failure probability in the unreliability expansion. Its coefficient counts the distinct weakest failure patterns, with each pattern weighted by its individual failure rates.

6.2 Probability guarantees that use the threshold

Proposition 6.4 Certified survivor-count bound. If $t = \tau_G(D) \geq 1$ and Z is the number of failed guards, then

$$R_{st}(G, D; \mathrm{p}) \geq \Pr(Z \leq t - 1). \tag{38}$$

For equal failure probability p and $d = |D|$, this becomes

$$R_{st}(G, D; p) \geq \sum_{j=0}^{t-1} \binom{d}{j} p^j (1-p)^{d-j}. \tag{39}$$

Proof. Every failure set of size at most $t - 1$ preserves security. Hence, the event on the right of (38) is contained in the success event. The binomial formula follows when the independent failure probabilities are equal. □

The total failure count can be used in this way because the graph structure has already certified the threshold. An expectation inequality alone does not establish the event containment used in the proof. When the probabilities differ, multiply the factors $(1 - p_v) + p_v z$. The coefficients of the resulting polynomial give the distribution of Z. Adding the coefficients of degrees at most $t - 1$ gives $\Pr(Z \leq t - 1)$.

The next bound uses local support directly. For $A_x = A_D(x)$, define

$$b_x = \prod_{v \in A_x} p_v + \sum_{v \in A_x} (1 - p_v) \prod_{w \in A_x \setminus \{v\}} p_w. \tag{40}$$

The quantity b_x is the probability that at most one neighbouring guard of x survives. An empty product is one, and an empty sum is zero.

Proposition 6.5 Local support bounds. For every deployment,

$$\max\left\{0, 1 - \sum_{x \in V} b_x\right\} \leq R_{st}(G, D; \mathrm{p}) \leq 1 - \max_{x \in V} \prod_{v \in A_D(x)} p_v. \tag{41}$$

Proof. If every vertex has at least two surviving neighbouring guards, Lemma 2.6 shows that the surviving set is secure. Applying the union bound to the complementary local events gives the lower bound. For any fixed vertex x, simultaneous failure of all guards in $A_D(x)$ destroys total domination. Taking the largest probability of these events gives the upper bound. □

The local events may overlap substantially, so (41) does not treat them as independent. Independence is used only among the component guard failures within each product. A lower bound that is zero or loose does not show that the deployment is unreliable.

6.3 Computing the first unreliability coefficient

Corollary 6.6 Exact leading coefficient in polynomial time. Suppose D is secure, let $t = \tau_G(D) \geq 1$, and let a_v be fixed positive rational numbers. Define

$$L_D(\mathrm{a}) = \sum_{F \in \mathcal{M}_G(D)} \prod_{v \in F} a_v. \tag{42}$$

Then, as $\varepsilon \downarrow 0$,

$$1 - R_{st}(G, D; \varepsilon \mathrm{a}) = L_D(\mathrm{a}) \varepsilon^t + O(\varepsilon^{t+1}). \tag{43}$$

The coefficient $L_D(\mathrm{a})$ can be computed exactly by the $O(n^5)$ combinatorial enumeration in Corollary 4.5, followed by $O(n^4)$ rational arithmetic operations. In the homogeneous case $a_v = 1$, it is the integer $|\mathcal{M}_G(D)|$.

Proof. Every destructive set of size t is inclusion-minimal, and every inclusion-minimal destructive set of size t has minimum cardinality. Therefore $\mathcal{M}_G(D) = \mathcal{C}_t$ in Corollary 6.3, which gives the expansion. By Corollary 4.5, all distinct members can be listed using polynomially many operations. There are $O(n^3)$ products in (42), each with at most n factors. Computing and adding these products therefore takes $O(n^4)$ arithmetic operations. With rational inputs, the result is exact. □

The quoted count of arithmetic operations is different from a bit-complexity bound: multiplication and addition costs depend on the binary lengths of the rational inputs. Together, the graph algorithm and rational evaluation are polynomial in the graph order and the input rational lengths. For instance, the product of all input denominators is a common denominator for every term in (42), and its bit length is at most the sum of the denominator lengths. The numerator lengths and their sum also grow polynomially. Since every a_v is positive, the coefficient is nonzero. The expansion uses failure probabilities for sufficiently small ε such that $\varepsilon a_v \leq 1$ for every guard.

The expansion follows directly from finite probability. The additional algorithmic result is that we can compute its exact leading coefficient from the specialised generating families, without listing every failure state or all the larger inclusion-minimal cuts.

6.4 Why all inclusion-minimal cuts are a different enumeration problem

Example 6.7 An exponential family of inclusion-minimal destructive sets. For every integer $k \geq 3$, there is a graph on $3k + 3$ vertices with a secure deployment of $3k + 2$ guards, threshold two, and at least $2^k - k - 1$ distinct inclusion-minimal destructive sets of size k.

Construction and proof. Take vertices c, d, u and v_i, w_i, x_i for $1 \leq i \leq k$. Add the edge cd. Join each of c, d to every v_i, w_i; join x_i to v_i, w_i; and join u to every v_i. There are no other edges. Place a guard at every vertex except u. Every vertex has at least two neighbouring guards, so Lemma 2.6 shows that the deployment is secure. Its minimum guarded-neighbourhood size is two, attained at each x_i.

A single failure cannot destroy security. Total domination remains because every vertex originally had at least two supports. Lemma 3.4 gives a defender for any failed original guard position. The only initially unoccupied vertex, u, still has at least $k - 1 \geq 2$ neighbouring guards. Each core vertex still has at least $2k$ neighbouring guards, and each pair vertex has at least two, even if the failed guard is c or d. Therefore only a vertex x_i could be left with exactly one surviving neighbour. After one failure, the only possible obstructing private neighbour for a candidate v_i is x_i, and this happens only when w_i has failed. Hence at most one surviving candidate is blocked, leaving another defender available. Theorem 3.1 now gives $\tau_G(D) = 2$.

For any subset $J \subseteq \{1, \ldots, k\}$ with $|J| \geq 2$, fail the set

$$F_J = \{w_i : i \in J\} \cup \{v_i : i \notin J\}. \tag{44}$$

Exactly one of v_i, w_i survives for each index, so every x_i remains supported. The core guards c, d support each other and all pair vertices. The guards adjacent to u are exactly those v_i with $i \in J$, and there are at least two. Thus the surviving deployment is total dominating. However, moving any possible defender v_i to u leaves x_i unsupported, because w_i failed and x_i is not adjacent to u. Hence F_J is destructive.

Now restore any one failed guard. If the restored guard is w_i for $i \in J$, then v_i is a valid defender of u. Before the move, x_i has both pair guards as supports; after v_i moves, it retains w_i. If the restored guard is v_i for $i \notin J$, that guard is a valid defender for the same reason, since w_i was already present. In either case, u retains another neighbouring guard because $|J| \geq 2$. Guards c, d remain adjacent, each pair position keeps its support, and every other x_j keeps its surviving pair guard. Every other unoccupied vertex is a failed pair vertex, for which c is a valid defender. At this stage, each neighbour of c has another neighbouring guard; no vertex has c as its only guard neighbour. Thus $P_S(c) = \varnothing$, and the deployment is secure after the restoration. Upward closure ensures that security is also preserved when more guards are restored. Every proper subset of F_J is therefore non-destructive, so F_J is inclusion-minimal.

The sets F_J are distinct and each contains k members. There are $2^k - k - 1$ choices with $|J| \geq 2$, proving the claim. □

This example shows that listing all inclusion-minimal destructive sets can require exponentially many outputs. It does not establish hardness for every exact reliability method. A compact symbolic representation, or a computation for a particular graph class, may avoid listing all cuts explicitly. Corollary 4.5 concerns only the minimum-cardinality subfamily, which can have polynomial size even in this example.

7. Active satellite guards

Let G be a nonempty graph and let U be a clique of $r \geq 1$ new vertices. Form the join $H = G \vee K_r$, so that each satellite is adjacent to every original vertex and to every other satellite. Choose $B \subseteq V(G)$ and place guards on

$$D = B \cup U. \tag{45}$$

Each satellite initially has a guard. After failures, let $A \subseteq B$ denote the surviving original guards and $Q \subseteq U$ the surviving satellites. Here, A is an ordinary dominating set of G when every vertex belongs to A or has a neighbour in A; the condition therefore uses closed neighbourhoods. The classification below follows formally from [9, Theorems 4.2 and 5.1] after translating the graph operations as described in Section 1.1. We give a direct proof to state its meaning for this fixed deployment, including disconnected graphs and graphs with isolated vertices.

Proposition 7.1 Satellite state classification after Go and Canoy [9]. The set $A \cup Q$ is an STDS of H if and only if one of these cases holds:

1. $|Q| \geq 2$;
2. $|Q| = 1$ and A is an ordinary dominating set of G;
3. $|Q| = 0$ and A is an STDS of G.

Proof. First, suppose that at least two satellites survive. They support one another and every original vertex. When an unoccupied original vertex is attacked, move a guard from one surviving satellite to it. Another satellite remains and supports every original vertex, while the moved guard supports the remaining satellite. All other satellites also remain supported. When a failed satellite is attacked, move a guard from a surviving satellite to that position; at least two satellites are then occupied. Both responses preserve total domination, so the deployment is secure.

Now suppose that exactly one satellite s survives. If A dominates the nonempty graph G, then A is nonempty. The satellite supports every original vertex, and a guard in A supports the satellite. For an unoccupied original vertex u, ordinary domination gives a neighbouring guard $v \in A$. Move v to u. The satellite s remains in place, and at

least one original guard remains, so total domination is preserved. If a failed satellite is attacked, move an original guard to it. The two occupied satellites then provide total domination.

Conversely, suppose A does not dominate G. Some vertex $u \in V(G) \setminus A$ then has no neighbour in A, so its only possible defender is s. If s moves to u, the vertex u has no neighbouring guard and the response cannot preserve total domination. If the deployment was not total dominating before the response, it was already insecure. Ordinary domination is therefore necessary.

Finally, suppose that no satellite survives. All support and responses at original vertices must come from A. Security in H therefore implies that A is an STDS of G. Conversely, an STDS of G has at least two vertices and supports every satellite. It also preserves responses to attacks at original vertices. For an attack at a satellite, move one guard to that satellite while keeping an original guard in place. This follows the construction in Proposition 2.7. □

This classification explains the satellite states. Two surviving satellites support one another. When one satellite survives, it supports every original vertex, although the original guards must still provide a valid response. When no satellite survives, secure total domination must be provided by the original graph alone.

7.1 Exact reliability composition

Assume that satellite $u \in U$ fails with probability q_u, independently for each satellite and independently of all original-guard failures. Define

$$a_0 = \prod_{u \in U} q_u, \qquad a_1 = \sum_{u \in U} (1 - q_u) \prod_{w \in U \setminus \{u\}} q_w. \tag{46}$$

Let $R_{dom}(G, B; \mathrm{p})$ be the probability that the surviving original guards ordinarily dominate G.

Corollary 7.2 Satellite reliability decomposition. With the independence assumptions above,

$$R_{st}(H, B \cup U; \mathrm{p}, \mathrm{q}) = 1 - a_0 - a_1 + a_1 R_{dom}(G, B; \mathrm{p}) + a_0 R_{st}(G, B; \mathrm{p}). \tag{47}$$

If all satellites have the same failure probability q, then $a_0 = q^r$ and $a_1 = r(1 - q)q^{r-1}$.

Proof. The three disjoint cases are that no satellite survives, exactly one survives, or at least two survive. These cases have probabilities a_0, a_1, and $1 - a_0 - a_1$, respectively. Proposition 7.1 specifies the condition on the original guards in each case. Independence of original and satellite failures then gives the formula by multiplication with the corresponding original reliability. □

The formula permits different failure probabilities for the original guards, and it also applies when G has isolated vertices. In that case the original secure total reliability is zero, while either of the other two terms can still give positive reliability after satellites are added.

7.2 Exact destructive threshold after satellite deployment

For B, define the ordinary domination failure threshold as

$$\kappa_G(B) = \min_{x \in V(G)} |N_G[x] \cap B|. \tag{48}$$

This value is zero exactly when B is not an ordinary dominating set. In general, it is the minimum number of original guards whose removal destroys ordinary domination. All guards in a smallest closed neighbourhood must fail to leave that neighbourhood without a guard; with fewer failures, each closed neighbourhood still contains a surviving guard.

Corollary 7.3 Satellite threshold formula. For $r \geq 1$ and nonempty G,

$$\tau_{G \vee K_r}(B \cup U) = \min\{r + \tau_G(B),\ r - 1 + \kappa_G(B)\}. \tag{49}$$

Proof. Proposition 7.1 shows that a destructive failure set cannot leave two or more satellites alive. In the case where none survives, all r satellites must be removed and secure total domination of the original deployment must also be destroyed. The smallest such set has size $r + \tau_G(B)$. If exactly one satellite survives, the failure set must remove $r - 1$ satellites and destroy ordinary domination by B. The minimum size in this case is $r - 1 + \kappa_G(B)$. The definitions of the two original thresholds show that both bounds can be attained. Their minimum gives the formula. This reasoning also covers an augmented deployment that is insecure initially, where the minimum is zero. □

For example, if $B = \varnothing$, the threshold is $r - 1$. A deployment with only active satellites tolerates f failures exactly when $r \geq f + 2$. One additional guard is needed to give the last responding guard a neighbour; a single surviving universal vertex cannot total dominate itself.

8. Worked comparisons on a bipartite service network

Take $G = K_{3,4}$ as a small abstract service graph. The two parts may stand for two classes of service locations, with every location adjacent to every location in the other class. This is an illustrative graph calculation; it uses no measured transportation data or empirical failure rates.

We compare two deployments of size five. Deployment D_A selects all three vertices in the first part and two in the second. Deployment D_B selects two vertices in the first part and three in the second. Both are secure initially and have $\mu_D = 2$, but their thresholds are different:

$$\tau_G(D_A) = 2, \qquad \tau_G(D_B) = 1. \tag{50}$$

Theorem 5.5 shows that D_A remains secure after one failure. In D_B, if a guard in the first part fails, an originally unoccupied vertex in the second part has only one supporting guard left, so a valid response is impossible. This example illustrates the two cases in Theorem 3.1.

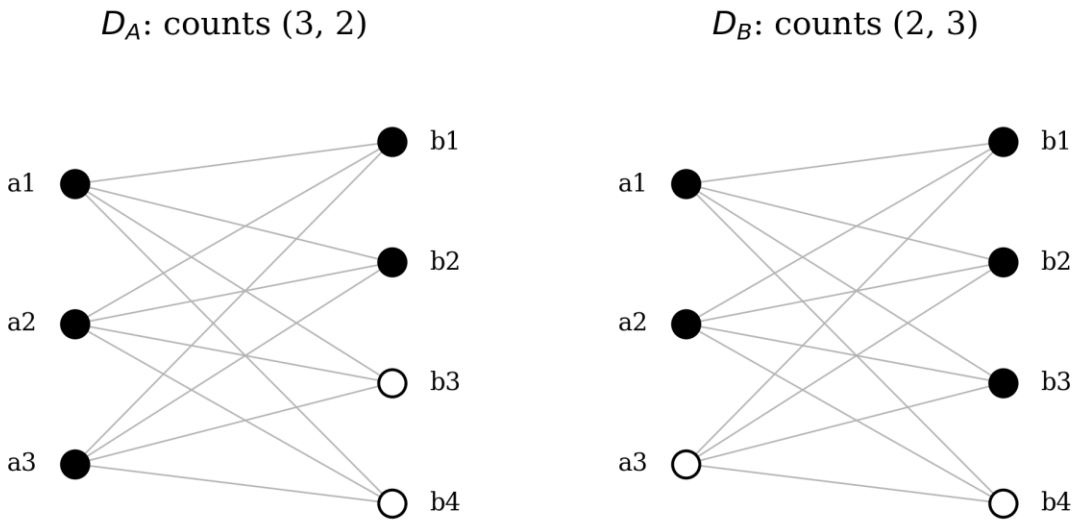


Figure 1. The deployments on $K_{3,4}$. Filled vertices have guards and open vertices do not. Both use five guards, but only the first tolerates every possible single guard failure.

8.1 Exact baseline reliabilities

Assume that every original guard independently fails with probability p, and set $s = 1 - p$. For D_A, every surviving set with at least four guards is secure, while no set with three or fewer guards is secure. It follows that

$$R_A(p) = s^5 + 5ps^4 = s^4(1 + 4p). \tag{51}$$

For D_B, both selected guards in the first part must survive, and at least two of the three guards in the second part must survive. Therefore

$$R_B(p) = s^2\left(s^3 + 3ps^2\right) = s^4(1 + 2p). \tag{52}$$

The difference is $2p(1 - p)^4 > 0$ for $0 < p < 1$. Yet the expected number of surviving guards is $5(1 - p)$ for either deployment. Comparing only expected survivor counts therefore does not distinguish their reliability.

When $p = 0.10$, both have an expected 4.5 surviving guards, but $R_A = 0.91854$ and $R_B = 0.78732$. The absolute difference is 0.13122, equal to 13.122 percentage points. These values come from evaluating the exact polynomials, not from simulation.

For small p, the first deployment has ten minimum destructive pairs, while the second has two minimum destructive singletons. Thus

$$1 - R_A(p) = 10p^2 + O(p^3), \qquad 1 - R_B(p) = 2p + O(p^2), \tag{53}$$

as Corollary 6.3 predicts. The two singleton cuts of D_B are its selected vertices in the part of size three.

All minimum pairs in the upper threshold example

Name the first part of $K_{3,4}$ as a_1, a_2, a_3 and the second as b_1, b_2, b_3, b_4. Let $D_A = \{a_1, a_2, a_3, b_1, b_2\}$. For every first-part vertex, the critical guarded neighbourhood is $\{b_1, b_2\}$. The other guarded neighbourhood is $\{a_1, a_2, a_3\}$, whose size is $\mu_D + 1 = 3$. Every pair of deployed guards is destructive, and each single failure leaves the deployment secure. The three generating families in Theorem 4.4 produce these pairs as shown below.

Failure pattern	**Number**	**Generating family**
Both selected second-part guards	1	$\mathcal{H}_0$
One guard from each part	6	$\mathcal{H}_1$
Two first-part guards	3	$\mathcal{H}_2$

Table 2. The ten minimum destructive pairs of D_A. This example needs all three types in the upper generating family.

The last three pairs have no second-part guard, so the construction using only the critical neighbourhood and one outside failure cannot produce them. They come from the guarded neighbourhood of size $\mu_D + 1$. In the middle row, after the first-part guard fails, only one opposite-part guard remains. The failed guard's position is then an undefendable response location. This shows why the upper-threshold filter must also check failed original positions.

For positive, unequal failure-rate coefficients λ_v, the leading coefficients are

$$L_{D_A}(\lambda) = \sum_{\{v,w\} \subseteq D_A} \lambda_v \lambda_w, \qquad L_{D_B}(\lambda) = \lambda_{a_1} + \lambda_{a_2}, \tag{54}$$

where $D_B = \{a_1, a_2, b_1, b_2, b_3\}$. The sum on the left is over distinct unordered two-element subsets. If every coefficient equals one, the two values are ten and two, respectively, as in (53).

8.2 Adding active satellites

For $B = D_A$, ordinary domination fails only when one part has no surviving selected vertex and the other part does not contain all its original vertices. Counting these states gives

$$R_{dom}(G, D_A; p) = (1 - p^3)(1 - p^2) + p^2(1 - p)^3 = 1 - 4p^3 + 3p^4. \tag{55}$$

Here the ordinary failure threshold is $\kappa_G(D_A) = 3$. Applying Corollary 7.3 gives $\tau_{G \vee K_r}(D_A \cup U) = r + 2$ for every $r \geq 1$. One active satellite increases the number of arbitrary failures that can be tolerated from one to two; two active satellites increase it to three.

At original-guard failure probability $p = 0.10$ and satellite failure probability $q = 0.05$, the exact reliability values are:

Deployment	Total guards	Destructive threshold	Exact reliability
D_B on $K_{3,4}$	5	1	0.78732000
D_A on $K_{3,4}$	5	2	0.91854000
D_A with one active satellite	6	3	0.99241200
D_A with two active satellites	7	4	0.99944485

Table 3. Exact reliability calculations for this example. The threshold is the first number of failures that can destroy security; the guaranteed number of tolerable failures is one less.

The table uses the different original and satellite failure probabilities stated above. The deterministic thresholds do not depend on those probabilities. Figure 2 uses one common failure probability for every deployed guard, so the four architectures can be compared as functions of a single parameter.

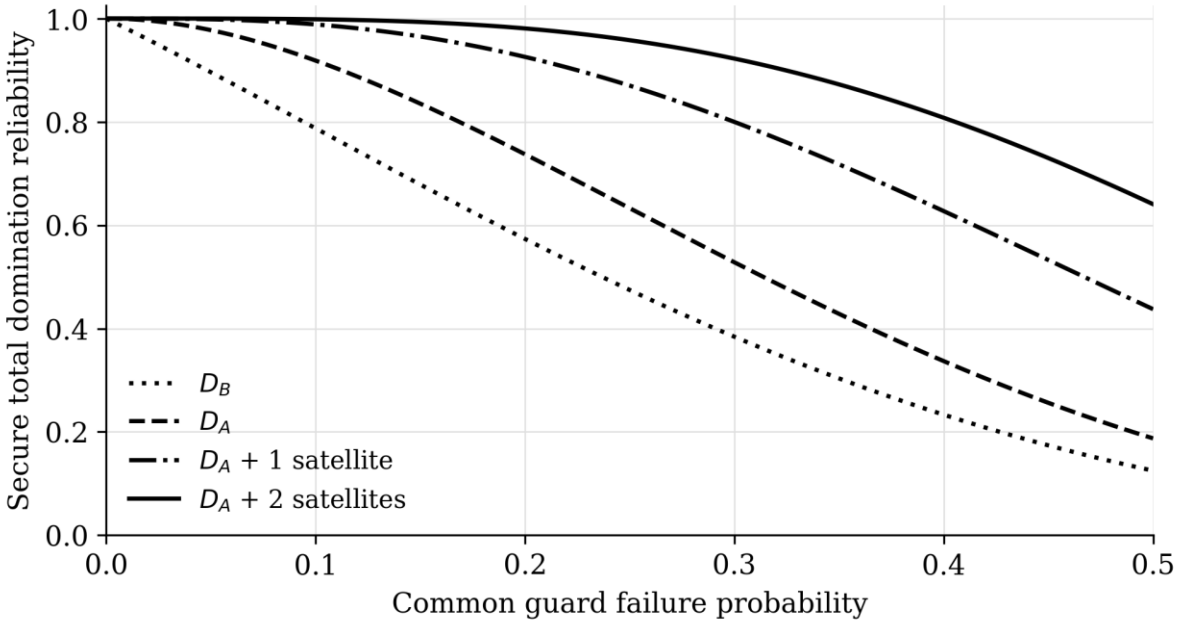


Figure 2. Exact reliability for different values of the common guard-failure probability. The curves are calculated using equations (47), (51), (52), and (55). They are analytical results for the stated graph, not observations from an operating network.

9. Exhaustive finite verification

The general conclusions follow from the proofs. We also performed an exact finite audit to check for possible mistakes involving quantifiers, open-neighbourhood conventions, and boundary cases. For each order from one through six, the audit generated every labelled simple graph on n vertices for $1 \leq n \leq 6$ by checking all $2^{\binom{n}{2}}$ edge sets. It then checked all 2^n deployments on each graph. The audit included graphs with isolated vertices, which correctly gave threshold zero for every deployment.

The direct program began by checking total domination. Then, for every unoccupied vertex, it examined each adjacent guard and checked whether the replacement set was total dominating, as required by Definition 2.1. In this way, it produced a truth table for secure total domination without using the critical-neighbourhood theorem. For each deployment, the destructive threshold was calculated from the exact recurrence

$$\begin{aligned} T(D) &= 0 \quad \text{if } D \text{ is insecure,} \\ T(D) &= 1 + \min_{v \in D} T(D \setminus \{v\}) \quad \text{if } D \text{ is secure.} \end{aligned} \tag{56}$$

The recurrence follows because every nonempty destructive set has a first guard to be removed. An independent program calculated the critical-neighbourhood certificate. For every graph and deployment, the audit compared the threshold obtained from this certificate with the value obtained by the direct program. It also checked upward closure and the identity $\tau_G(V) = \delta(G)$.

Order n	**Labelled graphs**	**All deployments**	**Total dominating deployments**	**Lower value $\mu_D - 1$**	**Upper value μ_D**
1	1	2	0	0	0
2	2	8	1	0	1
3	8	64	13	6	7
4	64	1,024	261	156	105
5	1,024	32,768	9,923	6,280	3,643
6	32,768	2,097,152	744,327	479,970	264,357
Total	33,867	2,131,018	754,525	486,412	268,113

Table 4. Counts from the exact exhaustive verification. Only total dominating deployments are included in the threshold columns. The lower column also includes insecure total dominating sets with $\mu_D = 1$ and threshold zero.

Every comparison agreed. The computations used integer bitsets and involved neither random sampling nor floating-point decisions. Since the enumeration is over labelled graphs, repeated isomorphic graphs were retained. The stated range therefore covers every labelled graph and every deployment, rather than a selection of examples.

We separately checked the satellite state classification for every labelled base graph of orders one through four, with one, two, or three satellites and every surviving-guard state. All 15,372 state comparisons and 3,294 deployment-threshold comparisons passed. For complete multipartite graphs, the audit covered all 58 integer-partition representatives of orders two through eight having at least two nonempty parts. The count characterisation passed all 60,664 deployment-and-budget cases, while the capacity formula agreed with direct minimisation in all 380 graph-and-budget cases. We also recomputed the Table 3 probabilities by direct state enumeration with rational arithmetic.

In order, the results were $19683/25000$, $45927/50000$, $248103/250000$, and $19988897/20000000$.

Finite verification cannot establish historical novelty. Agreement through order six also cannot replace a proof for arbitrary order. The checks support the stated mechanisms and provide reproducible verification of the formulas. The main results follow from a short chain of arguments: Lemma 2.6 supplies the lower threshold bound, minimum guarded neighbourhoods give the upper bound, and Lemmas 2.4 and 3.4 give the exact characterisation at the boundary.

Verification of all minimum failure sets

We also checked Theorem 4.4 and Algorithm 1 for every labelled simple graph of orders one through five. This audit covered 1,099 graphs, all 33,866 deployments, and every failure subset of each deployment. For each graph, a direct swap test first produced a truth table for all surviving sets. All 33,866 membership-oracle comparisons agreed with the direct definition. We then enumerated failure sets by brute force, found the complete family of minimum destructive sets, and compared it as a set with the generated and filtered family after removing duplicates.

The family comparison passed in every case. It compared 20,626 minimum destructive sets for deployments with $\mu_D > 0$ and separately checked all 23,668 zero-support deployments, each of which has the empty set as its unique minimum failure set. We also checked the construction in Example 6.7 through $k = 6$. This included the threshold-two claim for $3 \leq k \leq 6$ and the security obtained by restoring each individual failed guard in every displayed inclusion-minimal set. The range of these checks is limited as stated: the order-six threshold audit in Table 4 is not an order-six audit of the new complete-enumeration theorem.

10. Scope and further questions

The guard-failure model assumes that a location remains available after its assigned guard is lost. If a vertex is physically destroyed, the success condition must be defined on a changed graph, and a guard cannot return to the destroyed location. In Lemma 3.4, the proof restores a failed guard position. This is the exact step that would need a different argument for physical vertex failures.

Our security condition allows one defensive response after the failure set is known. Requiring the replacement state to remain secure would be a stronger condition. Continuing responses indefinitely would require a strategy over states. The theorems here establish neither property. Total domination ensures that each surviving guard has a neighbour, but it does not ensure that the subgraph induced by all surviving guards is connected. If communication must take place through occupied vertices, connectivity must be added and the local certificate reconsidered.

The critical-neighbourhood characterisation suggests several algorithmic questions. For sparse graphs, can adjacency lists or compression of repeated neighbourhoods improve its $O(n^4)$ evaluation bound? For a given failure budget, which graph classes permit polynomial-time optimisation of $\gamma_{st}^{(f)}$? The capacity formula answers the second question for complete multipartite graphs, and the component decomposition extends a class-specific solution to disjoint unions.

Determining exact reliability beyond its first nonzero term is a separate problem. Theorem 4.4 enumerates all minimum-cardinality destructive sets, and Corollary 6.6 gives the exact leading coefficient. However, Example 6.7 shows that the complete family of inclusion-minimal sets can still be exponentially large. A useful direction is to identify graph classes in which larger cuts have a compact representation or exact reliability can be computed without listing them. Unequal failure probabilities also give rise to a weighted placement problem, in which deployment cost, worst-case failure tolerance, and the low-failure expansion coefficients may be optimised together. These objectives can produce different rankings, as the example already shows for deployment size.

Satellite design offers another direction for extending the work. Equation (49) separates the failure of ordinary domination from that of secure total domination in the original graph. A satellite's value therefore depends on both original thresholds, not only on the number of backup vertices. The formulas change if satellites have restricted adjacencies or share a common failure cause. In particular, a shared-cause failure breaks the independence factorisation in equation (47), although the deterministic state classification in Proposition 7.1 still applies.

11. Conclusion

(1) Objective of the study. We study how many guard failures a fixed deployment can withstand while maintaining secure total domination. When a guard fails, its vertex remains in the graph and may still need support. Every vertex must have a neighbouring surviving guard. In addition, an unguarded vertex must be able to receive a guard through a single move that preserves total domination.

(2) Results proved. We proved that the smallest number of guards in an open neighbourhood determines the minimum number of failures that destroys security to within one unit. When this neighbourhood count is positive, the destructive-failure threshold is either the count itself or one less. When the count is zero, the deployment already fails to provide total domination. The critical-neighbourhood criterion distinguishes the two possible values and gives an explicit explanation of failure when the lower value occurs. Theorems 4.1 and 4.2 provide this characterisation and a polynomial-time algorithm for evaluating a given deployment. The related results establish feasibility conditions, identify vertices that must be occupied, and give exact deployment sizes for complete multipartite graphs. We also proved that all destructive failure sets of minimum size can be listed in polynomial time. Under the stated independent-failure assumptions, these sets give the exact leading coefficient of the unreliability expansion. For complete multipartite graphs, the equivalence with the corresponding closed tuple-domination condition explains when established neighbourhood requirements also guarantee a secure response after every permitted failure set. The reliability and universal-centre formulas describe the behaviour of the model under probabilistic failures and the addition of idealised backup vertices. The bipartite example shows that deployments with the same number of guards can have different failure thresholds and reliability values.

(3) New contributions and their scope. The main contribution presented here is the critical-neighbourhood failure criterion and its algorithm in Theorems 4.1 and 4.2. The complete multipartite equivalence is the second structural contribution. Enumerating all minimum-size destructive sets and using them to calculate the leading unreliability coefficient further extends the analysis. Among the sources examined, we did not identify the same critical-neighbourhood failure criterion under the same definitions and assumptions. This assessment is limited to the sources and searches reported in Appendix B and does not establish worldwide priority. Earlier defender criteria, existing domination results, and standard probability methods remain properly attributed. We do not claim that every result used or derived in this paper is new.

(4) Possible practical applications. The results may help in planning guard placements, mobile monitoring arrangements, emergency response resources, and backup services, provided that their support relationships can reasonably be represented by the graph model. A planner could use the failure criterion to compare proposed deployments, identify combinations of failures that make them unsafe, and examine where additional support may be useful. When suitable failure probabilities are available, the reliability formulas may support these comparisons. The universal-centre constructions provide an idealised way to study additional backup resources. These are proposed applications of the mathematical results and have not been demonstrated through field trials.

(5) Limitations. The model uses a fixed, finite, simple undirected graph. All vertices and edges remain after guard failures. It guarantees total domination after one response, but it does not establish security through an unlimited sequence of responses or require the guards to form a connected network. Travel time, service capacity, repair time, changing connections, and deployment costs are not included. The probability calculations assume independent guard failures, so further development is needed for failures caused by a common event. The main algorithm checks a given deployment. It does not find a minimum-cost deployment on an arbitrary graph. Although all minimum-size destructive sets can be listed in polynomial time, the example with exponentially many inclusion-minimal destructive

sets shows why this guarantee cannot be extended to listing all such sets. The finite computational checks support the mathematical proofs. Practical application requires suitable network data, realistic operational constraints, and independent evaluation.

12. Appendix A Reproducible verification procedure

The following procedure gives the two independent calculations used in Table 4. The vertices are numbered from zero to $n-1$, and subsets are represented as bitsets. A neighbourhood bitset does not include the vertex itself. The direct test follows the original definition, while the certificate test uses equations (8)–(10).

1. List all unordered pairs of vertices. For each possible edge set, construct a symmetric adjacency relation whose diagonal is zero.
2. For every subset S, check whether every vertex has a neighbour in S. If this holds, examine each $u \notin S$ by scanning its neighbours $v \in S$ and checking whether $(S \setminus \{v\}) \cup \{u\}$ is total dominating. Record whether at least one replacement is total dominating.
3. Process the deployment subsets in increasing order of size. Give each insecure subset threshold zero, and use recurrence (56) for each secure subset.
4. Independently calculate $A_D(x)$, μ_D, the critical candidate sets, and the sets of blocked guard labels. Return zero when $\mu_D = 0$; if a certificate is found, return $\mu_D - 1$; otherwise, return μ_D.
5. Compare the two threshold values. For each secure subset, check every superset obtained by adding one vertex. Compare the threshold of the full deployment with the minimum degree of the graph.
6. Add the results to the counts in Table 4, retaining disconnected graphs and graphs with isolated vertices. If a comparison fails, stop and record the graph, the deployment, and the two different values.

For the additional audit of Theorem 4.4, enumerate every failure subset of each deployment through order five. From the direct truth table, retain the destructive subsets of minimum size and compare this family with Algorithm 3. Separately check the recognition oracle against the direct swap predicate. For Example 6.7, construct the graph as specified and test every displayed set F_J, as well as every restoration of one of its failed guards.

The source graph supplied with this study is not used as a computational dataset. The enumeration uses only the stated integer order bound and the finite graph axioms. Hence, the scope of the audit does not depend on the selection of an external dataset.

13. Appendix B Prior result mapping and literature scope

We checked the literature on 29 October 2026. Where available, the review used primary publisher or author sources. It also involved targeted searches using secure-total and multiple-domination terminology and examination of relevant cited results. We compared the exact defender rule with Castillano et al. [5, Theorem 2.5], the universal-centre classification with Go and Canoy [9, Theorems 4.2 and 5.1], and the numerical bipartite overlap with the example after Theorem 2.2 in Cabrera Martínez [4]. We also examined the distinctions among closed tuple domination, tuple total domination, and the one-response guard-failure model.

Recent context is provided by the 2024 sources [1,2], the 2025 source [3], and the 2025 preprint and 2026 journal versions of [13]. These do not replace comparison with the closest older results. We treat the versions of [13] as one research work. The original supplied manuscript [14] is an unpublished conceptual source; we do not attribute a journal publication, dataset validation, or DOI to it.

Access to the literature and the coverage of this review are incomplete. Some earlier secure-total papers were available only through bibliographic records or abstracts. As we could not open the publisher PDF, we checked the

exact text of Theorem 6 in Kazemi [11] through an indexed excerpt from the primary article. We did not complete an exhaustive search of MathSciNet, zbMATH, or Scopus, a systematic forward-citation search, or a review of non-English literature. In this targeted pass, we did not obtain the full texts of the 2007 initiating paper by Benecke and coauthors, Duginov [8], or Poureidi's 2022 work. Their bibliographic records and accessible abstracts are used only for context. The full historical priority of the arbitrary-part-size capacity formula also remains unresolved.

Within this scope, we did not find the same critical-neighbourhood failure criterion or the same minimum-cardinality failure-set enumeration under the same definitions and hypotheses. The explicit exponential family shows the scope of the enumeration, without a claim that it is the first such construction. A result not found in the sources checked may still be present elsewhere in the mathematical literature. The paper therefore proves its stated mathematical implications and makes no global first-result claim pending a wider review.

REFERENCES

1. Y. Aita and T. Araki. Secure total domination number in maximal outerplanar graphs. *Discrete Applied Mathematics* **353** (2024), 65–70. https://doi.org/10.1016/j.dam.2024.04.007. Author version: arXiv:2403.03404v2.
2. T. Araki and Y. Aita. An algorithm for the secure total domination problem in proper interval graphs. *Theoretical Computer Science* **1011** (2024), article 114706. https://doi.org/10.1016/j.tcs.2024.114706.
3. I. Boufelgha, M. Ahmia, and M. Guettiche. Secure 2-domination in graphs. *Journal of Discrete Mathematical Sciences and Cryptography* **28**(7) (2025), 2723–2736. https://doi.org/10.47974/JDMSC-2067.
4. A. Cabrera Martínez. A note on the -tuple domination number of graphs. *Ars Mathematica Contemporanea* **22**(4) (2022), article P4.03. https://doi.org/10.26493/1855-3974.2600.dcc. Author version: arXiv:2104.03172v2.
5. E. C. Castillano, R. A. L. Ugbinada, S. R. Canoy Jr., and C. C. Go. Secure domination and secure total domination in the composition *International Journal of Mathematical Analysis* **8**(35) (2014), 1741–1749. https://doi.org/10.12988/ijma.2014.47190.
6. M. Dod. *Domination in graphs with application to network reliability*. Doctoral dissertation, Technische Universität Bergakademie Freiberg, 2015. https://d-nb.info/1220912212/34.
7. K. Dohmen and P. Tittmann. Domination reliability. *The Electronic Journal of Combinatorics* **19**(1) (2012), paper P15. https://doi.org/10.37236/1166.
8. O. Duginov. Secure total domination in graphs: Bounds and complexity. *Discrete Applied Mathematics* **222** (2017), 97–108. https://doi.org/10.1016/j.dam.2016.08.018.
9. C. E. Go and S. R. Canoy Jr. Domination in the corona and join of graphs. *International Mathematical Forum* **6**(16) (2011), 763–771. https://www.m-hikari.com/imf-2011/13-16-2011/goIMF13-16-2011.pdf.
10. M. A. Henning and A. P. Kazemi. -tuple total domination in graphs. *Discrete Applied Mathematics* **158**(9) (2010), 1006–1011. https://doi.org/10.1016/j.dam.2010.01.009.
11. A. P. Kazemi. On the total -domination number of graphs. *Discussiones Mathematicae Graph Theory* **32**(3) (2012), 419–426. https://doi.org/10.7151/dmgt.1616.
12. W. F. Klostermeyer and C. M. Mynhardt. Secure domination and secure total domination in graphs. *Discussiones Mathematicae Graph Theory* **28**(2) (2008), 267–284. https://doi.org/10.7151/dmgt.1405.
13. I. Lamprou, I. Sigalas, I. Vaxevanakis, and V. Zissimopoulos. Approximations for fault-tolerant total and partial positive influence domination. *Discrete Mathematics & Theoretical Computer Science* **28**:2 (2026), article 1. https://doi.org/10.46298/dmtcs.15903. Preprint first posted 15 June 2025 as arXiv:2506.12828; final author version v3 dated 15 January 2026.
14. R. V. Prasad. Advanced Secure Domination Parameters in Graphs: A Unified Framework for Resilience in Transportation Networks under Probabilistic and Multi-Failure Scenarios. Unpublished manuscript dated 20 September 2025.