AND DATA SCIENCES
BULLETIN OF COMPUTER

Threshold Temporal Reachability Domination for Resilient Diffusion in Temporal Graphs

Jia-Bao Liu[1] and Waqas Nazeer[2]

[1] School of Mathematics and Physics, Anhui Jianzhu University, Hefei 230601, P.R. China

[2] Department of Mathematics, Government College University, Lahore Pakistan

Abstract

Temporal graphs model systems whose interactions change over time. In such networks, temporal reachability domination seeks a small seed set whose influence can reach all vertices. However, one successful temporal path is often not enough in practical settings, where repeated exposure or backup delivery may be needed. To address this, this paper introduces the q-Temporal Reachability Dominating Set (q-TaRDiS), which requires every vertex to be reachable from at least q distinct seeds, together with a budgeted version that maximizes threshold coverage under a fixed seed limit. The study presents the model, establishes basic properties, shows membership in NP, and notes NP-hardness in the unrestricted case. It also shows that the problem reduces to set multicover once temporal reachability sets are computed, leading to an exact integer programming model and an efficient greedy solution strategy. Experiments on synthetic temporal networks show that the greedy method is near-optimal on small instances and that threshold-based seed sets provide better robustness under random temporal edge deletion. These results show that threshold temporal domination offers a practical and analyzable framework for resilient diffusion in dynamic networks.

Keywords: temporal graphs; temporal reachability; resilient diffusion; multicover; submodular optimization; redundancy-aware seeding

1. Introduction

Dynamic interaction systems are rarely well described by a single static graph. Contact opportunities occur, disappear, and reappear over time; transportation links follow schedules; communication channels become intermittently available; and information spreads through sequences of time-ordered encounters rather than through permanently open edges. A broad survey of time-varying graphs framed this viewpoint and established temporal networks as a distinct algorithmic and modeling domain [1]. Early foundational work on connectivity and inference in temporal networks also made clear that timing alone can change the computational character of propagation problems [2]. A complementary line of work showed that the reachability relation of a temporal graph can itself be summarized as a derived object, namely a temporal reachability graph, which is often a useful intermediary between temporal and static reasoning [3]. The practical usefulness of temporal graph metrics on real systems was further highlighted through applications to mobility and interaction data [4].

Corresponding author: waqas Nazeer (nazeer.waqas@gmail.com)

BCDS 2026, Vol. 7(1), 38-53. DOI: 10.71448/bcds2671-4

Within that broad area, a major family of optimization questions asks how timing can be modified or exploited to alter who reaches whom. One formulation studies how delaying labels can shrink reachability sets [5]. Another asks how times can be assigned so that the maximum reachability of a single vertex becomes as small as possible [6]. A closely related direction compares delaying and deleting operations when the goal is to reduce epidemic-style spread in temporal networks [7]. Temporal labeling questions also arise on the positive side, for example when one wants an optimum labeling that guarantees temporal connectivity [8]. Even dense temporal structures can sometimes be represented more economically, as shown by sparse spanner constructions for temporal cliques [9].

The specific problem that motivates the present paper belongs to this same family but focuses on source selection rather than edge editing. Temporal reachability domination asks for a small set of initial vertices from which the entire network becomes temporally reachable. This problem was formalized directly as the Temporal Reachability Dominating Set problem and studied through classical and parameterized complexity across several temporal models [10]. The broader conceptual setting behind such questions is the general study of dynamic networks and temporal reachability [11]. Subtle differences between simple, strict, proper, and happy temporal graphs show that the semantics of time-respecting paths can materially change reachability behavior [12]. Related structural work on temporally connected components further emphasizes that components in temporal graphs are not mere copies of connected components in static graphs [13].

Yet the classical one-hit requirement is limited. In many realistic diffusion settings, it is not enough to know that a vertex can be reached once by some seed at some time. Reaching a vertex twice by independent or at least distinct seed origins may be operationally more meaningful. In epidemiological reconstruction, temporal contact sequences are often interpreted cautiously because contact does not guarantee effective transmission, and causal inference remains sensitive to uncertainty [14]. Threshold behavior is also central in epidemic modeling because outbreak viability may depend on temporal structure rather than on static averages alone [15]. Control-oriented work on deleting temporal edges to restrict epidemic size similarly reflects the fact that spread processes are naturally evaluated in terms of robustness and suppression rather than bare reachability [16]. In ephemeral networks, random link availability directly motivates designs that remain effective despite missing temporal contacts [17].

Comparable concerns arise beyond epidemiology. Delay-robust routing asks whether arrival guarantees survive temporal perturbation [18]. Temporal covering problems with sliding windows encode the intuition that a single brief coverage event may be insufficient [19]. Temporal network optimization under connectivity constraints shows that scheduling decisions and resource limitations interact strongly with the time dimension [20]. In broadcast and dissemination settings, the practical issue is rarely whether one successful transmission schedule exists in principle; rather, the issue is whether dissemination remains effective under constraints on time, resources, or topology [21]. Hardness results for broadcast scheduling reinforce the gap between conceptual reachability and deployable communication plans [22]. Complexity studies on broadcasting in planar and decomposable graphs provide similar evidence that seemingly simple communication tasks can become structurally difficult [23]. Multiple-originator broadcasting makes the role of several simultaneous or coordinated sources explicit [24]. Broader work on graph broadcasting also repeatedly returns to questions of redundancy, source placement, and time-efficiency [25]. Recent studies on shifting labels to minimize reachability times show once again that carefully adjusting temporal structure can drastically alter the quality of dissemination [26].

These observations motivate the central idea of this paper: a vertex should be considered robustly covered only if it is temporally reachable from at least q distinct seeds. The proposal is deliberately modest. It does not attempt to model full probabilistic dependence, behavioral adoption thresholds, or path-disjoint reliability. Instead, it introduces the smallest natural extension of temporal reachability domination that distinguishes between minimal and redundant temporal influence. This extension already changes both the interpretation and the optimization landscape.

A simple operational picture helps fix ideas. Imagine a hospital issues a same-day antimicrobial recall shortly before a shift change. Infection-control staff contact ward liaisons, ward liaisons brief nurses and junior doctors, pharmacy confirms stock handling, and transport staff relay the update to satellite units. Under a one-hit model, the alert process looks successful as soon as one temporal chain reaches each unit. In practice, however, a missed bridge at midday may leave an entire ward uncovered. A threshold formulation is more faithful to that reality because it asks whether every target can be reached from several distinct alert origins rather than from one lucky temporal route.

The paper develops this idea as a self-contained contribution. The theoretical aim is to define the threshold model carefully and to clarify the immediate consequences of that definition. The algorithmic aim is to show that once temporal reachability sets are computed, the problem becomes amenable to exact and approximate techniques already familiar from covering optimization. The empirical aim is to demonstrate that threshold-aware seed sets offer a measurable robustness advantage under simulated temporal failures. These three aims fit the profile of a hybrid theory-and-application paper: the theoretical layer organizes the model, the algorithmic layer explains how to solve it, and the experimental layer shows why the added threshold requirement matters in practice.

Three design principles guide the development. First, the model should remain close to temporal reachability rather than introducing a completely different propagation rule. Second, the computational pipeline should be explicit enough to implement directly. Third, the experiments should answer a practical design question: when the temporal network degrades, does paying for redundancy on the original graph improve retained coverage? The answer produced by the study is affirmative.

2. Model, Structural Properties, and Algorithmic Framework

Let $\mathcal{G} = (V, E, \lambda)$ be a temporal graph with lifetime τ, where each edge $e \in E$ is active at a nonempty subset $\lambda(e) \subseteq [\tau]$. For each $t \in [\tau]$, let $E_t = \{e \in E : t \in \lambda(e)\}$ and let $G_t = (V, E_t)$ denote the tth snapshot. Throughout the computational part of the paper we use nonstrict temporal paths, meaning that the time labels along a path are nondecreasing, although the threshold framework itself does not depend on that particular choice.

For a seed vertex $u \in V$, let $R(u) \subseteq V$ denote the temporal reachability set of u, that is, the vertices reachable from u by a time-respecting path. If $S \subseteq V$ is a seed set, classical temporal reachability domination requires that every vertex belong to $\bigcup_{u \in S} R(u)$. The threshold formulation replaces this one-hit criterion by a multiplicity requirement.

Definition 1 (q-Temporal Reachability Dominating Set)**.** *Let $q \in \mathbb{N}$. A set $S \subseteq V$ is a* q-Temporal Reachability Dominating Set*, abbreviated* q-TaRDiS*, if every vertex $v \in V$ is temporally reachable from at least q distinct vertices of S. Equivalently,*

$$|\{u \in S : v \in R(u)\}| \geq q \qquad \text{for all } v \in V.$$

The minimum-cardinality optimization problem is denoted Min-q-TaRDiS.

The threshold parameter q controls the strength of the requirement. The case $q = 1$ coincides with ordinary temporal reachability domination. The case $q = 2$ asks for redundant temporal coverage, and larger values of q can be interpreted as stronger redundancy targets.

A second formulation becomes important when the number of seeds is externally capped.

Definition 2 (Budgeted threshold temporal coverage). *For a seed set $S \subseteq V$ and threshold q, define*

$$F_q(S) = \sum_{v \in V} \min(q, |\{u \in S : v \in R(u)\}|) .$$

Given a budget b, the budgeted threshold temporal coverage problem asks for a set $S \subseteq V$ with $|S| \leq b$ maximizing $F_q(S)$.

The function $F_q(S)$ counts thresholded coverage rather than raw repeated coverage. Once a vertex has been reached from q seeds, additional reaching seeds are not rewarded. This cap is conceptually important because it prevents a small region of the graph from dominating the objective merely by being repeatedly reachable from many seeds.

The threshold formulation is meaningful only when the instance is feasible. Define the total-reachability multiplicity of a vertex v by

$$\rho(v) = |\{u \in V : v \in R(u)\}| .$$

If $\rho(v) < q$ for some v, then no seed set can satisfy the threshold at v. Thus a necessary and sufficient instance-level feasibility condition is that $\rho(v) \geq q$ for every vertex. This condition is inexpensive to check after all reachability sets are computed.

If a set S is a q-TaRDiS, then it is also a $(q-1)$-TaRDiS whenever $q \geq 2$. Therefore, the minimum seed requirement is monotone nondecreasing in q. This property matters experimentally because it clarifies how much additional cost is incurred when one moves from minimal coverage to redundant coverage.

Distinct reaching seeds do not necessarily correspond to path-disjoint routes or independent stochastic transmissions. Nevertheless, they provide a simple deterministic proxy for redundancy. If the original graph is perturbed mildly, a vertex reachable from two or three seeds is intuitively less fragile than a vertex reachable from only one seed. The deletion experiments in Section 3 are designed to test exactly this interpretation.

Let $r_{\max} = \max_{u \in V} |R(u)|$. Since each selected seed can contribute at most one unit of threshold coverage to at most $r_{\max}$ vertices, any feasible solution obeys

$$|S| \geq \left\lceil \frac{q|V|}{r_{\max}} \right\rceil .$$

This bound is crude but useful as a first diagnostic. A trivial upper bound is $|V|$, obtained by selecting all vertices. A slightly sharper bound follows from feasibility itself: any set containing every vertex that is not dominated by any other reachability set is automatically feasible whenever the instance is feasible, although this observation is usually too loose to be algorithmically decisive.

The decision problem takes as input a temporal graph $\mathcal{G}$, a threshold q, and an integer k, and asks whether there exists a q-TaRDiS of size at most k.

Proposition 1. *For $q = 1$, Min-q-TaRDiS coincides with the usual temporal reachability domination problem.*

Proof. When $q = 1$, the condition

$$|\{u \in S : v \in R(u)\}| \geq 1$$

for all $v \in V$ is equivalent to requiring that every vertex be temporally reachable from at least one seed in S. This is exactly the one-hit formulation. □

Theorem 1. *The decision version of q-TaRDiS belongs to NP.*

Proof. A certificate is a candidate seed set S. One computes the reachability set of each selected seed, accumulates for every vertex the number of selected seeds that reach it, and verifies that all counts are at least q. Temporal reachability can be computed in polynomial time, so the verification procedure is polynomial. □

Proposition 2. *The decision version of q-TaRDiS is NP-hard in the unrestricted case.*

Proof. The special case $q = 1$ is already the classical temporal reachability domination problem. Hence any unrestricted formulation that allows $q = 1$ inherits NP-hardness immediately. □

The NP-hardness statement just given is intentionally conservative. It avoids claiming a full fixed-threshold complexity classification, which would require additional reductions tailored to the threshold parameter itself. Even so, the result is enough to justify the computational strategy adopted here: exact optimization for moderate instances and greedy methods for larger ones. The use of reductions as a baseline hardness language follows the standard tradition of NP-completeness theory [27]. From a parameterized point of view, it is also natural to ask whether fixed values of q, lifetime, or structural parameters of the footprint graph might support tractable algorithms [31]. The broader language of fixed-parameter tractability remains useful when interpreting such possibilities [32]. Bounded-treewidth formulations are especially attractive because optimization on graph-structured data often becomes more manageable in that regime [33]. Logic-based metatheorems expressed in monadic second-order language provide one of the cleanest formal routes for exploiting that structure [34].

The threshold problem becomes much simpler conceptually after a single preprocessing step: compute all reachability sets $R(u)$. Once those sets are available, the temporal aspect is compressed into a family of static subsets of V. For each vertex u, define the set $\mathcal{R}_u := R(u)$. Then Min-q-TaRDiS asks for the fewest sets among $\{\mathcal{R}_u : u \in V\}$ such that every element of the universe V is covered at least q times.

Proposition 3. *After all temporal reachability sets are computed, Min-q-TaRDiS is exactly a set multicover problem on the family $\{R(u) : u \in V\}$ with uniform demand q.*

Proof. Selecting a seed u contributes one cover to each vertex in $R(u)$. Therefore, selecting a seed set S covers a vertex v exactly $|\{u \in S : v \in R(u)\}|$ times. Requiring at least q covers for each vertex is the multicover requirement with uniform demand q. □

This reduction is the key methodological observation of the paper. It tells us that the genuinely temporal part of the pipeline is reachability computation, whereas the optimization layer is a structured covering problem.

Introduce a binary variable x_u for every candidate seed vertex $u \in V$, where $x_u = 1$ if u is selected and $x_u = 0$ otherwise. Define the coverage matrix $A \in \{0,1\}^{|V|\times|V|}$ by

$$A_{vu} = \begin{cases} 1 & \text{if } v \in R(u), \\ 0 & \text{otherwise.} \end{cases}$$

Then Min-q-TaRDiS admits the exact formulation

$$\min \sum_{u\in V} x_u$$

subject to

$$\sum_{u\in V} A_{vu} x_u \geq q \qquad \text{for all } v \in V,$$

$$x_u \in \{0,1\} \qquad \text{for all } u \in V.$$

The model is compact: it has one binary variable per candidate seed and one covering constraint per vertex. If weighted seeds are later introduced, the objective simply becomes $\min \sum_{u\in V} c_u x_u$.

Some candidate seeds are obviously unnecessary. If $R(u) \subseteq R(w)$, then selecting w is never worse than selecting u in the unweighted minimum-cardinality problem. Therefore u can be removed from the candidate set without changing the optimum. In practice this rule reduces symmetry and shrinks the ILP.

Lemma 1. *If $R(u) \subseteq R(w)$, then there exists an optimal solution that does not select u unless it also selects w.*

Proof. Suppose an optimal seed set S contains u but not w. Replacing u by w cannot reduce the number of times any vertex is covered, because every vertex covered by u is also covered by w. The resulting solution has the same cardinality and is still feasible. □

The multicover interpretation suggests a natural greedy rule. For a current seed set S, define the residual demand of a vertex v by

$$d_S(v) = \max(0,\ q - |\{u \in S : v \in R(u)\}|)\,.$$

At each iteration, choose a new seed $u \notin S$ maximizing the number of vertices in $R(u)$ whose residual demand is still positive. Equivalently, choose

$$u^* \in \arg\max_{u\in V\setminus S} \sum_{v\in R(u)} \mathbf{1}[d_S(v) > 0].$$

The algorithm terminates when all residual demands are zero.

This rule is simple but quite effective. It mimics classical greedy multicover while retaining a direct interpretation in temporal language: choose the vertex whose temporal influence helps the largest number of still-under-covered vertices.

Once the problem has been reduced to multicover, the greedy method inherits the usual logarithmic-style behavior associated with covering heuristics. Greedy set cover itself is classically analyzed through the work of Chvátal [28]. Submodular set covering analyses also support the use of greedy selection for capped or demand-based covering objectives [29]. These results do not eliminate the need for exact methods on moderate instances, but they explain why a greedy solver is a reasonable large-scale baseline.

The fixed-budget version is valuable when the number of deployable seeds is predetermined. In that case, minimizing seed count is not the correct optimization target; instead, one should maximize thresholded coverage under a cardinality cap.

Theorem 2. *For fixed q, the function*

$$F_q(S) = \sum_{v \in V} \min(q, |\{u \in S : v \in R(u)\}|)$$

is monotone and submodular.

Proof. For a fixed vertex v, the map

$$S \mapsto |\{u \in S : v \in R(u)\}|$$

is a modular counting function. The scalar transformation $x \mapsto \min(q, x)$ is nondecreasing and concave on the nonnegative integers. A nondecreasing concave transformation of a modular count is submodular. Summing the resulting vertex-wise contributions preserves monotonicity and submodularity. □

The significance of this theorem is practical as much as theoretical. It means that the budgeted problem falls into the classical cardinality-constrained submodular maximization setting. Standard greedy selection therefore has a firm analytical basis in this formulation. The general approximation analysis for monotone submodular maximization under a cardinality constraint is rooted in the work of Nemhauser, Wolsey, and Fisher [30].

The full computational workflow used in the paper has five stages.

1. Build or import a temporal graph and fix a reachability convention.

2. Compute the reachability set $R(u)$ for every candidate seed $u \in V$.

3. Check feasibility by verifying that every vertex is reached by at least q candidates.

4. Apply dominance pruning and then solve the minimum-threshold problem either exactly by ILP or heuristically by greedy residual demand.

5. Evaluate the obtained seed set on the original graph and, when desired, under perturbed temporal graphs obtained by deleting time-edges.

The main conceptual advantage of this pipeline is modularity. Reachability computation, optimization, and robustness evaluation are separated cleanly. This makes it easy to replace one component without changing the others. For example, one could switch from exhaustive search to ILP, from nonstrict to strict reachability, or from random deletion to targeted deletion while keeping the rest of the workflow unchanged.

The threshold requirement introduces a useful tension between efficiency and resilience. A one-hit solution may be smallest, but it often concentrates influence too aggressively. A threshold solution is forced to distribute influence more broadly, which tends to produce seed sets that remain effective after mild perturbation. This claim is not a theorem in full generality, because specific perturbations can behave adversarially. It is, however, a natural hypothesis, and the experiments are designed precisely to evaluate it on controlled synthetic instances.

3. Experimental Design and Results

The experimental study was designed to answer three questions. First, is the greedy residual-demand algorithm close to exact optimization on instances small enough to solve exhaustively? Second, how much extra seed cost is incurred when moving from the one-hit objective to a threshold objective with $q = 2$? Third, do threshold-aware seed sets preserve more practical coverage when temporal edges are deleted after optimization?

Two families were used so that the evaluation would not depend on a single structural regime.

The first family consists of sparse temporal Erdős–Rényi graphs, abbreviated ER. For each snapshot, every unordered vertex pair appears independently with a fixed probability. This family produces relatively homogeneous temporal structure and serves as a baseline setting in which diffusion opportunities are distributed without strong mesoscopic organization.

The second family consists of temporally clustered two-community graphs, abbreviated COM. The vertex set is partitioned into two equal communities. Intra-community edges appear with higher probability than inter-community edges, and the temporal schedule is staged so that cross-community interaction is weaker early in the lifetime and stronger later. This design produces a simple but useful surrogate for delayed inter-group mixing.

These two graph families deliberately emphasize different obstacles to threshold domination. In ER graphs, the main challenge is sparse random connectivity. In COM graphs, the main challenge is temporal and structural separation between communities.

To make the study less abstract, a third evaluation layer uses a Python-generated application-inspired alert network, abbreviated ALERT. The simulator represents a single operating day in a hospital-style coordination environment with four clinical units, shared support services, and six ordered contact blocks corresponding to local huddles, escalation, cross-unit bridging, unit relay, service confirmation, and late fallback relay.

The ALERT traces contain 32 vertices in total: four liaison nurses, four medical leads, twelve nurses, four administrative assistants, and eight support or coordination staff. Across 20 generated days, the simulator produces between 103 and 111 undirected temporal contacts per day, with a mean of 107.2. For this application layer, only operational broadcasters are treated as eligible seeds: the four unit liaisons, the two infection-control coordinators, and the operations coordinator. This restricted candidate set reflects a realistic deployment rule in which not every staff member is authorized to originate a system-wide alert.

All experiments used nonstrict temporal reachability. The choice was made for two reasons. First, nonstrict reachability is a common and computationally straightforward convention. Second, it isolates the threshold effect without introducing an additional layer of strict-versus-nonstrict comparison. Every reported coverage quantity should therefore be interpreted under nondecreasing time labels along temporal paths.

Three solution mechanisms were used.

The first is exhaustive search, used only on small instances. It enumerates all seed subsets in increasing size until it finds an optimal feasible solution.

The second is the greedy residual-demand method introduced in the previous section. Because the paper focuses on a threshold generalization rather than on solver engineering, the greedy method was intentionally kept simple and transparent.

The third is a static footprint-degree baseline. This baseline ignores temporal ordering. It aggregates all temporal edges into a footprint graph, ranks vertices by static degree, and adds them until the threshold condition is satisfied. The baseline is not expected to be competitive on a genuinely temporal problem, but it provides a useful comparison point because it represents what a practitioner might do if temporal structure were ignored.

For the exact-versus-greedy comparison, graph sizes were $n \in \{12, 14, 16\}$ and lifetime was fixed at $T = 4$. For each graph family and each threshold $q \in \{1, 2\}$, 60 feasible instances were collected, with 20 feasible instances at each graph size. The outcome measures were mean optimum, mean greedy solution size, and mean baseline solution size.

For the robustness study, graph size was increased to $n = 60$ and lifetime was set to $T = 5$. For each family, 20 base graphs were generated. On every base graph, one greedy seed set was computed for $q = 1$ and another for $q = 2$. After optimization, 10%, 20%, or 30% of temporal edges were deleted uniformly at random. For each deletion level, 30 independent deletion trials were applied to each base graph. Coverage was then reevaluated at threshold 1 using the original seed sets.

This evaluation choice deserves emphasis. The post-deletion performance metric is one-hit coverage even for seed sets originally optimized under $q = 2$. The purpose is not to test whether the deleted graph still satisfies the stricter threshold. Instead, the purpose is to measure whether paying for redundancy on the original graph increases the probability that basic diffusion remains intact after perturbation. This makes the interpretation of the results especially transparent for applications.

Two robustness metrics were reported: the probability that complete one-hit coverage is preserved and the mean fraction of vertices that remain one-hit covered. The first is a stringent all-or-nothing indicator. The second is smoother and reveals whether losses are localized or widespread. The same metrics were also applied to the ALERT traces so that the generic random families and the application-inspired case could be compared on a common scale.

Table 1 summarizes the small-instance results. The greedy method matches the optimum exactly in three of the four tested settings and exceeds it only slightly in the remaining one. The static footprint-degree baseline is consistently much worse.

Table 1. Exact versus greedy comparison on small temporal graphs. Values are means over 60 feasible instances per family and threshold.

Family	q	Mean optimum	Mean greedy	Mean degree baseline	Greedy/Opt.	Degree/Opt.
ER	1	1.717	1.717	5.583	1.000	2.846
ER	2	2.300	2.300	3.383	1.000	1.404
COM	1	1.650	1.650	5.483	1.000	2.965
COM	2	3.133	3.167	6.350	1.008	2.053

Several observations follow immediately. First, the threshold objective does not seem to make small instances algorithmically unstable for greedy optimization. Even at $q = 2$, the heuristic remains essentially optimal in the tested regimes. Second, the gap between greedy and static degree is large. This matters because it shows that temporal order, not merely aggregate exposure volume, determines which seeds are useful. A vertex with many footprint neighbors may still be a poor temporal seed if its active edges appear too early, too late, or in an unhelpful order.

Third, the change from $q = 1$ to $q = 2$ is modest but noticeable in seed cost. In ER graphs the mean optimum increases from roughly 1.7 to 2.3. In COM graphs the increase is larger, from roughly 1.65 to 3.13. This difference is itself informative. Community structure creates temporal bottlenecks, and redundancy becomes more expensive precisely because influence must cross those bottlenecks more than once.

Table 2 reports the main deletion results. Threshold-aware seed sets almost always preserve more one-hit coverage after random temporal edge deletion than one-hit-optimized seed sets. The gains are especially visible in ER graphs, where random deletion more directly disrupts already sparse temporal routes.

Table 2. Robustness of greedy seed sets under random deletion of temporal edges. Coverage is evaluated at threshold 1 after deletion, even when the original seed set was designed for $q = 2$.

Family	Del.	Seeds q=1	Seeds q=2	Full cov. q=1	Full cov. q=2	Mean frac. q=1	Mean frac. q=2
ER	10%	1.00	2.00	0.753	0.847	0.992	0.997
ER	20%	1.05	2.10	0.370	0.527	0.965	0.984
ER	30%	1.05	2.10	0.160	0.277	0.915	0.970
COM	10%	1.00	2.00	0.905	0.923	0.998	0.998
COM	20%	1.00	2.00	0.763	0.775	0.995	0.995
COM	30%	1.00	2.00	0.437	0.532	0.965	0.987

The most revealing comparison occurs at intermediate and high deletion levels. In ER graphs, the probability of preserving complete one-hit coverage rises from 0.370 to 0.527 at 20% deletion and from 0.160 to 0.277 at 30% deletion when one switches from $q = 1$ seeds to $q = 2$ seeds. Mean retained coverage also rises noticeably, from 0.965 to 0.984 at 20% deletion and from 0.915 to 0.970 at 30% deletion. In COM graphs the same pattern appears, although the gain is more moderate because the underlying graphs are already more structured and stable at low deletion rates.

The generic random families establish the basic robustness pattern, but they remain stylized. For that reason, the study also included the ALERT case generated directly in Python. The ALERT network is not presented as a real observational dataset. Instead, it functions as an application-shaped testbed in which vertices correspond to recognizable staff roles and the temporal schedule mirrors a plausible same-day alert workflow. The summary of the Python-generated ALERT case is presented in Table 3. The simulator generates one temporal graph per day, with role-labeled vertices and a fixed six-block coordination schedule.

Table 3. Summary of the Python-generated ALERT case. The simulator produces one temporal graph per day with role-labeled vertices and a fixed six-block coordination schedule.

Attribute	Value
Clinical units	4 (ED, Ward A, Ward B, Diagnostics)
Vertices per day	32
Eligible broadcaster candidates	7
Time blocks per day	6
Contacts per day	103–111 (mean 107.2)
Generated days	20
Deletion trials per day and level	100

Operationally, the ALERT traces are easy to read. Time block 1 represents local unit huddles, time block 2 represents initial escalation to coordination staff, time block 3 represents the bridge between coordination teams and unit leads, time block 4 represents local relay inside each unit, time block 5 represents service confirmations, and time block 6 provides a fallback liaison chain. In the generated traces, the greedy solver typically selects the emergency-department liaison as the unique $q = 1$ broadcaster and adds one ward liaison when $q = 2$ is required. That behavior is intuitive: the first seed exploits the earliest alert bridge, while the second seed creates a second operational origin on the ward side of the schedule. Average performance of greedy seed sets on the Python-generated ALERT traces is shown in Table 4. Coverage is reevaluated at threshold 1 after random deletion of temporal edges.

Table 4. Average performance of greedy seed sets on the Python-generated ALERT traces. Coverage is reevaluated at threshold 1 after random temporal edge deletion.

Deletion	Seeds q=1	Seeds q=2	Full cov. q=1	Full cov. q=2	Mean frac. q=1	Mean frac. q=2
10%	1.00	2.00	0.475	0.571	0.976	0.982
20%	1.00	2.00	0.166	0.251	0.942	0.954
30%	1.00	2.00	0.044	0.081	0.895	0.917

The ALERT results reinforce the same message already visible in the ER and COM experiments, but they do so in a setting that looks more like a human workflow than a purely abstract graph family. At every deletion level, the two-seed threshold solution preserves more one-hit functionality than the one-seed solution. The gain in complete post-failure coverage is especially clear at 10% and 20% deletion, where the probability of preserving full alert reach rises from 0.475 to 0.571 and from 0.166 to 0.251, respectively. The mean retained coverage also improves steadily.

Figure 1 makes the trend visually clear. The two-seed threshold solution does not eliminate fragility, because the temporal schedule still contains vulnerable bridges, but it consistently shifts the retained-coverage curve upward. This is exactly the type of practical improvement that motivates the threshold formulation: a modest increase in the number of operational broadcasters can translate into noticeably better resilience when the observed contact process is incomplete or disrupted.

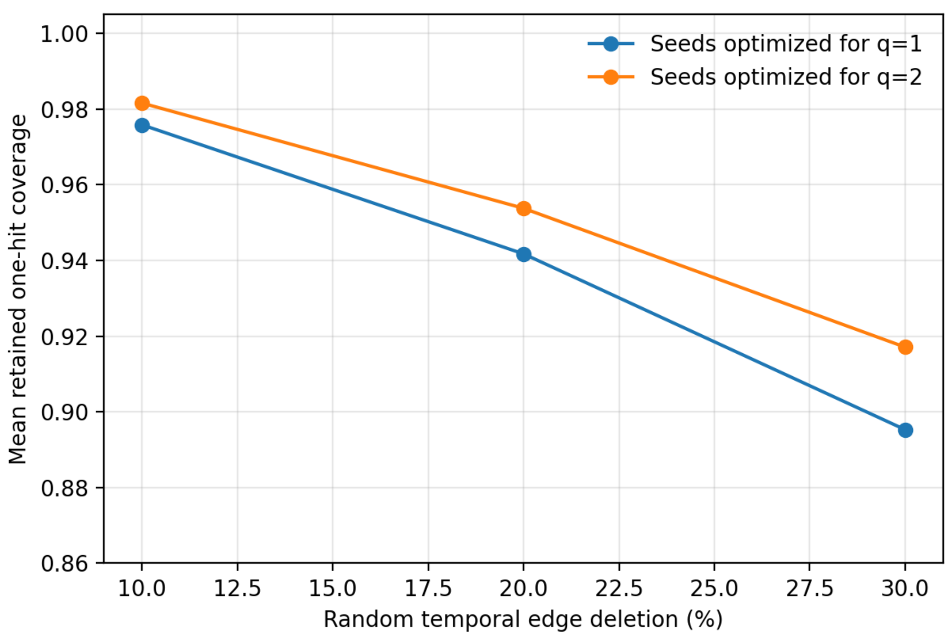

Fig. 1. Mean retained one-hit coverage on the Python-generated ALERT traces after random temporal edge deletion.

The application-shaped case also sharpens the interpretation of the static degree baseline. In a role-constrained environment, a naive footprint heuristic may recommend people who look central in aggregate but are not actually positioned at the right times to originate a reliable alert. The threshold problem therefore points to a broader lesson for deployment settings: temporal dissemination plans should be evaluated in temporal terms and against the operational candidate set that would exist in practice.

4. Discussion and Conclusion

The main conceptual contribution of this paper is the observation that temporal reachability domination becomes substantially more realistic when one distinguishes between single exposure and redundant exposure. The resulting threshold model remains close to the classical temporal-reachability language, but it changes the practical meaning of a good solution. A minimum one-hit seed set answers the question, “What is the smallest group that can reach everyone at least once?” A minimum threshold seed set answers the stronger question, “What is the smallest group that can reach everyone robustly enough that each vertex has several independent seed origins?”

That difference matters in applications for several reasons. In epidemic alerting or public communication, repeated exposure can be interpreted as reinforcement. In network monitoring, it can be interpreted as backup dissemination. In distributed control, it can be interpreted as insurance against localized temporal failures. None of these interpretations requires a full probabilistic model to be useful. A deterministic threshold already captures the first layer of redundancy that practitioners often care about.

From a theoretical standpoint, the model has two attractive features. The first is definitional economy: it extends temporal reachability domination by changing only the demand placed on each target vertex. The second is algorithmic modularity: after reachability preprocessing, the problem becomes an instance of multicover, and the budgeted version becomes a standard monotone submodular maximization problem. This means that a threshold formulation can be studied with familiar tools while remaining genuinely temporal.

The experiments complement this theoretical picture. On small instances, greedy residual-demand selection is essentially exact. On larger instances, threshold-aware seed sets preserve more useful coverage after failure. The joint reading of these facts is important. If threshold-aware optimization were only more robust but much harder to compute, its practical value would be limited. If it were easy to compute but provided no robustness gains, it would add little beyond the one-hit model. The paper suggests that neither concern is decisive in the tested regimes.

At the same time, several limitations should be acknowledged clearly.

First, the model counts distinct seeds rather than path-disjoint routes. A vertex reached twice from two seeds may still depend on overlapping temporal substructures. This means that q-coverage is a proxy for redundancy rather than a full reliability guarantee.

Second, graphs are appropriate for controlled comparison, but they cannot capture all the behavioral and structural peculiarities of real contact systems. The present study therefore demonstrates the mechanism rather than claiming domain-specific predictive accuracy.

Third, the complexity analysis is intentionally preliminary. Membership in NP and unrestricted NP-hardness are immediate and important, but they do not exhaust the theoretical questions raised by the model. Fixed-threshold hardness, approximation lower bounds, parameterized complexity with respect to lifetime or structural width, and special cases on restricted temporal graph classes all remain open.

Fourth, the robustness protocol is only one among many possible perturbation models. Uniform random deletion is a clean first test, but some applications may require burst failures, adversarial deletions, or time-window disruptions. Those scenarios could change the relative value of threshold seeding and deserve separate study.

These limitations lead directly to a substantial agenda for future work.

A weighted version of the model would allow different deployment costs for seeds and different importance weights for target vertices. This would be useful in settings where certain nodes are expensive to activate or particularly important to protect.

A heterogeneous-threshold version would permit each vertex v to carry its own demand q_v. Such a formulation could represent mixed populations in which some vertices require more redundancy than others.

A probabilistic extension could attach success probabilities to time-edges and replace deterministic reachability sets by expected or high-confidence coverage quantities. The threshold viewpoint developed here would still be relevant, but the optimization layer would need to account for dependence and uncertainty more explicitly.

A path-based extension could require each target to be reachable through multiple temporally disjoint or edge-disjoint paths rather than merely from multiple seeds. This would create a stronger notion of resilience at the cost of a more complex feasibility test.

A structural program of research could ask which temporal graph classes admit efficient exact algorithms. Bounded lifetime, bounded treewidth of the footprint graph, bounded number of locally earliest edges, or strong restrictions on temporal labeling all seem plausible starting points.

A data-driven continuation should test the methodology on real temporal interaction datasets from public health, transportation, mobility, or wireless communication. The strongest justification for threshold temporal domination will ultimately come from settings in which a small increase in seed cost produces a large and domain-relevant increase in operational robustness.

Despite these open questions, the main message of the paper is already clear. Threshold temporal reachability domination offers a simple and useful extension of temporal domination. It is theoretically natural, algorithmically structured, and empirically meaningful. The model captures a practical idea that the one-hit formulation suppresses: in dynamic systems, it is often not enough that influence can reach a vertex once; it matters whether that reachability has some redundancy behind it.

Reproducibility Note

All numerical values reported in this manuscript were generated from a Python prototype that computes temporal reachability sets, solves the small-instance exact problem by exhaustive search, applies the greedy multicover heuristic, and evaluates robustness under random temporal edge deletion. The parameter settings reported in Section 3 are sufficient to reproduce the tables from the same code base.

References

[1] Casteigts, A., Flocchini, P., Quattrociocchi, W., & Santoro, N. (2012). Time-varying graphs and dynamic networks. *International Journal of Parallel, Emergent and Distributed Systems, 27*(5), 387-408.

[2] Kempe, D., Kleinberg, J., & Kumar, A. (2000, May). Connectivity and inference problems for temporal networks. In *Proceedings of the thirty-second annual ACM symposium on Theory of computing* (pp. 504-513).

[3] Whitbeck, J., Dias de Amorim, M., Conan, V., & Guillaume, J. L. (2012, August). Temporal reachability graphs. In *Proceedings of the 18th annual international conference on Mobile computing and networking* (pp. 377-388).

[4] Tang, J., Leontiadis, I., Scellato, S., Nicosia, V., Mascolo, C., Musolesi, M., & Latora, V. (2013). Applications of temporal graph metrics to real-world networks. In *Temporal Networks* (pp. 135-159). Berlin, Heidelberg: Springer Berlin Heidelberg.

[5] Deligkas, A., & Potapov, I. (2022). Optimizing reachability sets in temporal graphs by delaying. *Information and Computation, 285*, 104890.

[6] Enright, J., Meeks, K., & Skerman, F. (2021). Assigning times to minimise reachability in temporal graphs. *Journal of Computer and System Sciences, 115*, 169-186.

[7] Molter, H., Renken, M., & Zschoche, P. (2024). Temporal reachability minimization: Delaying vs. deleting. *Journal of Computer and System Sciences, 144*, 103549.

[8] Klobas, N., Mertzios, G. B., Molter, H., & Spirakis, P. G. (2024). The complexity of computing optimum labelings for temporal connectivity. *Journal of Computer and System Sciences, 146*, 103564.

[9] Casteigts, A., Peters, J. G., & Schoeters, J. (2021). Temporal cliques admit sparse spanners. *Journal of Computer and System Sciences, 121*, 1-17.

[10] Kutner, D. C., & Larios-Jones, L. (2025). Temporal reachability dominating sets: contagion in temporal graphs. *Journal of Computer and System Sciences,* 103701.

[11] Casteigts, A. (2018). *A journey through dynamic networks (with excursions)* (Doctoral dissertation, Université de Bordeaux).

[12] Casteigts, A., Corsini, T., & Sarkar, W. (2024). Simple, strict, proper, happy: A study of reachability in temporal graphs. *Theoretical Computer Science,* 991, 114434.

[13] Balev, S., Sanlaville, E., & Schoeters, J. (2024). Temporally connected components. *Theoretical Computer Science, 1013*, 114757.

[14] Braunstein, A., & Ingrosso, A. (2016). Inference of causality in epidemics on temporal contact networks. *Scientific Reports, 6*(1), 27538.

[15] Valdano, E., Ferreri, L., Poletto, C., & Colizza, V. (2015). Analytical computation of the epidemic threshold on temporal networks. *Physical Review X, 5*(2), 021005.

[16] Enright, J., Meeks, K., Mertzios, G. B., & Zamaraev, V. (2021). Deleting edges to restrict the size of an epidemic in temporal networks. *Journal of Computer and System Sciences, 119*, 60-77.

[17] Akrida, E. C., Gąsieniec, L., Mertzios, G. B., & Spirakis, P. G. (2016). Ephemeral networks with random availability of links: The case of fast networks. *Journal of Parallel and Distributed Computing, 87*, 109-120.

[18] Füchsle, E., Molter, H., Niedermeier, R., & Renken, M. (2022). Delay-robust routes in temporal graphs. *arXiv preprint arXiv:2201.05390.*

[19] Akrida, E. C., Mertzios, G. B., Spirakis, P. G., & Zamaraev, V. (2020). Temporal vertex cover with a sliding time window. *Journal of Computer and System Sciences, 107,* 108-123.

[20] Mertzios, G. B., Michail, O., Chatzigiannakis, I., & Spirakis, P. G. (2013, July). Temporal network optimization subject to connectivity constraints. In *International Colloquium on Automata, Languages, and Programming* (pp. 657-668). Berlin, Heidelberg: Springer Berlin Heidelberg.

[21] Peleg, D. (2007, December). Time-efficient broadcasting in radio networks: A review. In *International Conference on Distributed Computing and Internet Technology* (pp. 1-18). Berlin, Heidelberg: Springer Berlin Heidelberg.

[22] Erlebach, T., & Hall, A. (2004). NP-hardness of broadcast scheduling and inapproximability of single-source unsplittable min-cost flow. *Journal of Scheduling, 7*(3), 223-241.

[23] Jakoby, A., Reischuk, R., & Schindelhauer, C. (1998). The complexity of broadcasting in planar and decomposable graphs. *Discrete Applied Mathematics, 83*(1-3), 179-206.

[24] Chia, M. L., Kuo, D., & Tung, M. F. (2007). The multiple originator broadcasting problem in graphs. *Discrete Applied Mathematics, 155*(10), 1188-1199.

[25] Grigoryan, H. (2013). *Problems related to broadcasting in graphs* (Doctoral dissertation, Concordia University).

[26] Deligkas, A., Eiben, E., & Skretas, G. (2021). Minimizing reachability times on temporal graphs via shifting labels. *arXiv preprint arXiv:2112.08797.*

[27] R.M. Karp, Reducibility among combinatorial problems, in: R.E. Miller, J.W. Thatcher, J.D. Bohlinger (Eds.), *Complexity of Computer Computations*, Springer, Boston, MA, 1972, pp. 85–103.

[28] Chvatal, V. (1979). A greedy heuristic for the set-covering problem. *Mathematics of Operations Research, 4*(3), 233-235.

[29] Wolsey, L. A. (1982). An analysis of the greedy algorithm for the submodular set covering problem. *Combinatorica, 2*(4), 385-393.

[30] Nemhauser, G. L., Wolsey, L. A., & Fisher, M. L. (1978). An analysis of approximations for maximizing submodular set functions—I. *Mathematical programming, 14*(1), 265-294.

[31] Cygan, M., Fomin, F. V., Kowalik, Ł., Lokshtanov, D., Marx, D., Pilipczuk, M., ... & Saurabh, S. (2015). *Parameterized algorithms* (Vol. 5, No. 4, p. 16). Cham: Springer.

[32] Downey, R. G., & Fellows, M. R. (1995). Fixed-parameter tractability and completeness I: Basic results. *SIAM Journal on computing, 24*(4), 873-921.

[33] Courcelle, B. (1997). The Expression of Graph Properties and Graph Transformations in Monadic Second-Order Logic. *Handbook of graph grammars, 1*, 313-400.

[34] Courcelle, B., & Engelfriet, J. (2012). Graph structure and monadic second-order logic: a language-theoretic approach (Vol. 138). Cambridge University Press.

How to cite this article: Jia-Bao Liu and Waqas Nazeer (2026). Threshold Temporal Reachability Domination for Resilient Diffusion in Temporal Graphs. *Bulletin of Computer and Data Sciences*, 7(1), 38-53. DOI: 10.71448/bcds2671-4

Received: 05/1/2026 **Revised:** 27/2/2026 **Accepted:** 10/3/2026 **Published:** 31/03/2026

BULLETIN OF COMPUTER AND DATA SCIENCES *Bulletin of Computer and Data Sciences* is a peer-reviewed open access journal.