Diversity and Trust to Increase Structural Robustness in Networks

Waseem Abbas[1], Aron Laszka[2], and Xenofon Koutsoukos[3]

[1]*Information Technology University, Lahore, Pakistan*

[2]*University of Houston, Texas*

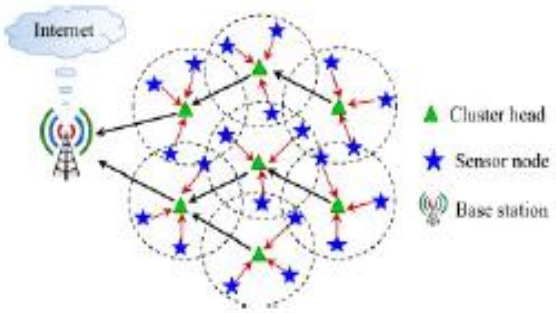
[3]*Vanderbilt University*

Presenter: **Mudassir Shabbir**

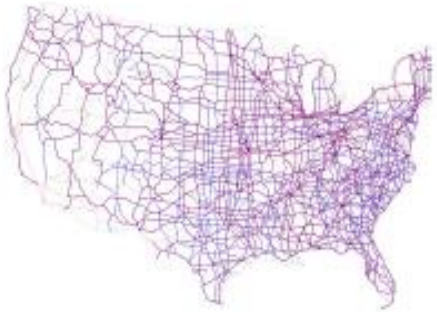
Information Technology University, Lahore, Pakistan

Motivation

internet topology



sensor network



infrastructure

social network

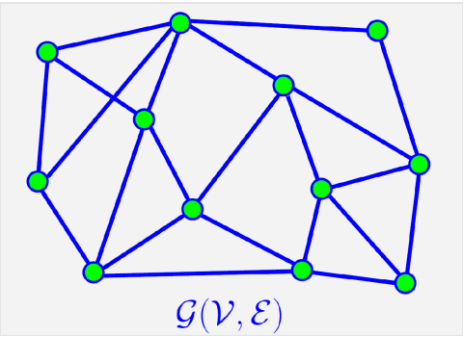
- Networks are failure-prone (vulnerable to node (edge) removals.)
- Thus, connectivity between nodes (required for network operations), might be severely affected.
- We desire networks to be structurally robust, that is, to remain connected under node (edge) removals.

How can we improve structural robustness of networks?

Introduction – Structural Robustness

Structural Robustness:

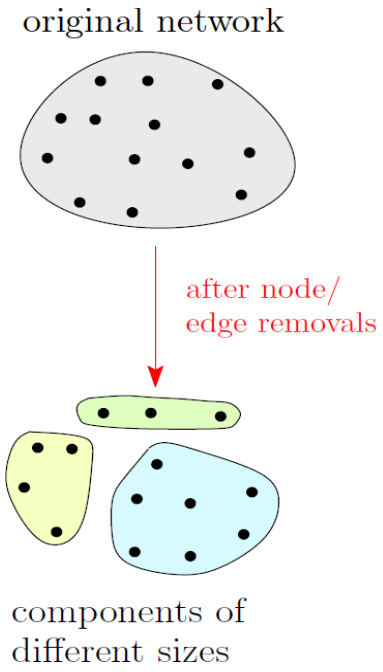
Network's ability to retain and preserve its *structure* as a result of node and edge removals.



Measuring Structural Robustness:

Based on two factors:

1) Effort required to cause the damage (e.g., no. of nodes or edges removed),
2) Extent of damage
    1. Number of resulting components
    2. Size of components



Examples:

Connectivity, Pair-wise connectivity, r-robustness, integrity, toughness, tenacity, expansion ratio, and many more ...

Contributions

Main Problem:

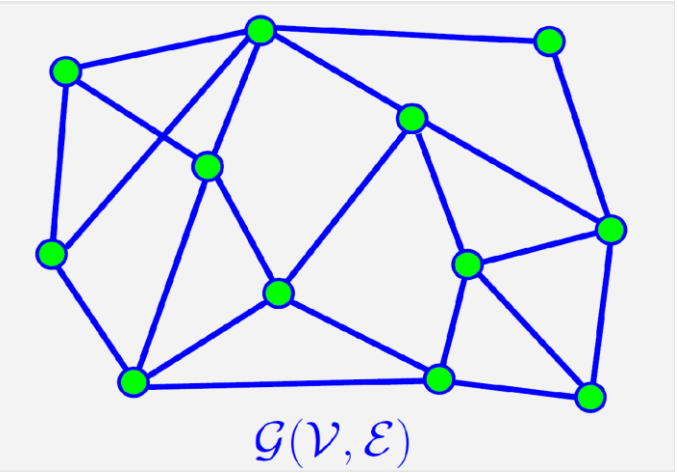
How can we improve structural robustness in networks, as measured by **pairwise connectivity**, without adding any further links?

Approach:

By exploiting **trustiness**, and **diversity** of nodes, we can improve pairwise connectivity even in sparse networks.

Results:

- How trustiness and diversity can be useful?
- Problem complexity
- Heuristics to distribute trusted and diverse nodes
- Numerical evaluation



Measuring Robustness – Pairwise Connectivity

Pairwise Connectivity:

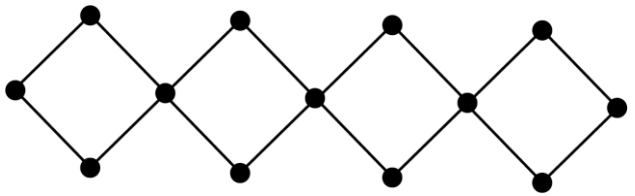
Fraction of **node-pairs** that are connected with each other through a path.

Applications:

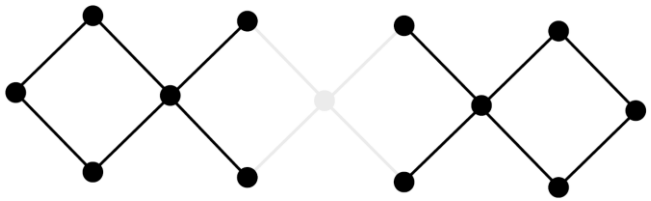
- robustness of communication networks,
- key players in anti-terrorism networks,
- targeted vaccination for pandemic prevention, and
- biological networks etc.

Why Pairwise Connectivity:

Measures both the number and sizes of components



Pair-wise connectivity = 1

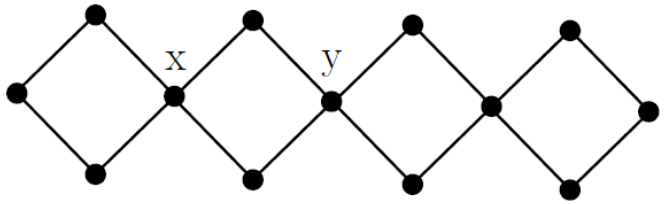


After removing middle node,
Pair-wise connectivity = 0.4545

Measuring Robustness – Pairwise Connectivity

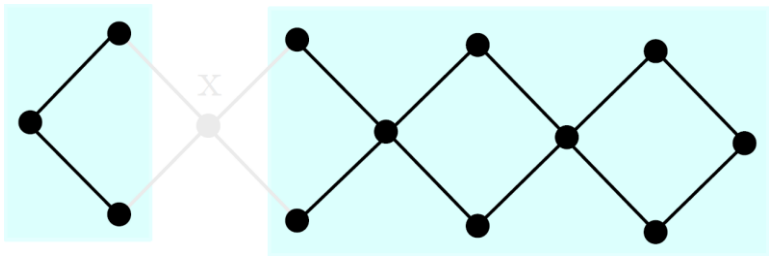
Pair-wise connectivity gives more information about the structural robustness of network as compared to vertex-connectivity.

Example: The graph is 1-connected (can be disconnected by removing either of the nodes x or y).



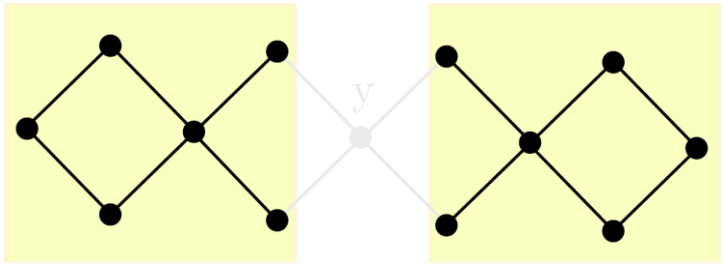
However, pair-wise connectivity is different in both cases.

1) Removing x



Pair-wise connectivity = 0.59

2) Removing y

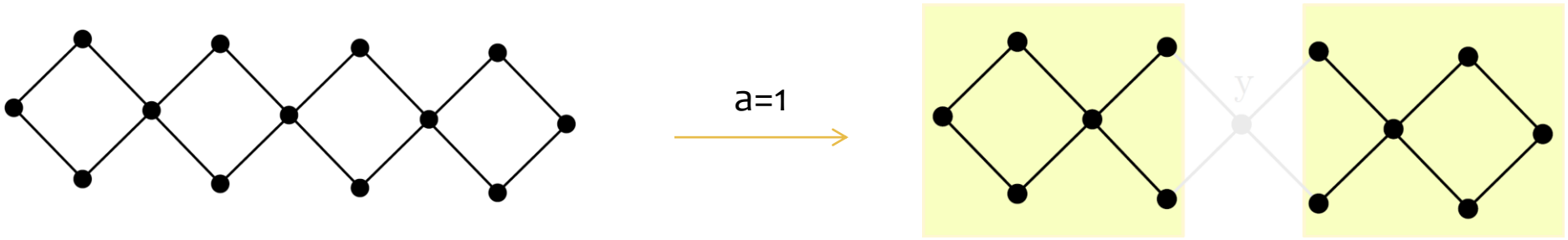


Pair-wise connectivity = 0.454

Attackers Objective

Attacker's objective (Critical node detection problem):

Given an undirected graph **G** and an integer ***a***, delete a subset of at most *a* nodes such that the pair-wise connectivity of the remaining graph is minimized.



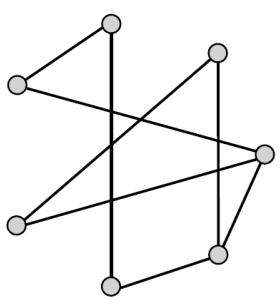
Problem Complexity: Critical node detection problem is known to be **NP-complete** (Arulselvan et al. 2009)

Improving Pairwise Connectivity

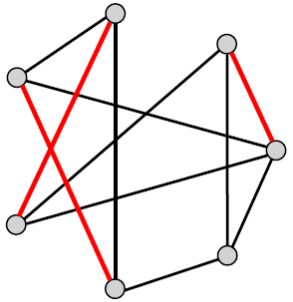
How can we minimize the impact of an attack, that is, maximize the pairwise connectivity of the residual network?

A Typical Approach (Redundancy):

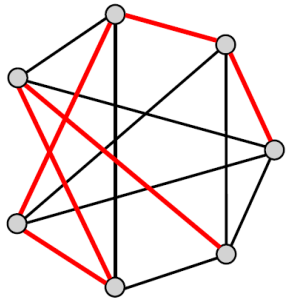
Strategically **add edges**, also known as *Connectivity Augmentation*.



2-connected



3-connected



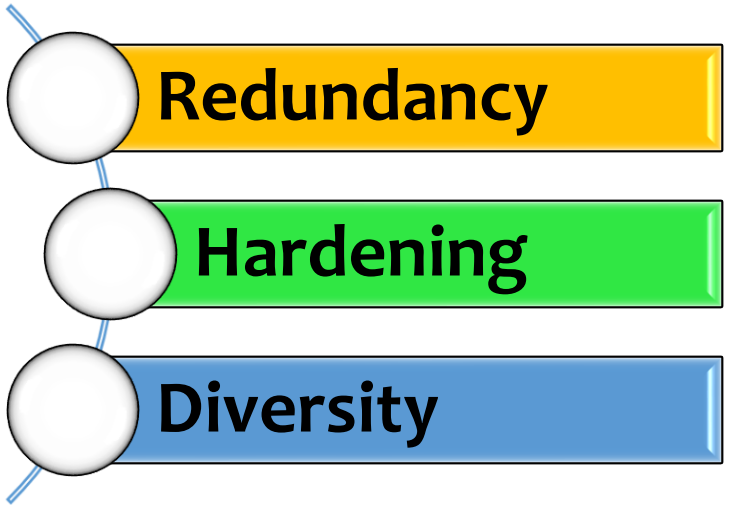
4-connected

Issues:

- Could be prohibitively expensive
- Not suitable for sparse graphs
- Security concerns (attack surface is increased)

Hardening and Diversity

Can there be other ways to improving pairwise connectivity of the residual graph?



Redundancy:
- Adding more components (e.g., devices, links)
- Adversary has to compromise more components

Hardening:
- making individual components or types more resilient (e.g., penetration testing, vulnerability discovery for platforms and tamper resistant hardware for devices)
- Devices are much harder to compromise

Diversity:
- Using multiple types of components (e.g., different software/hardware platforms)
- Disjoint set of vulnerabilities.

Hardening and Diversity

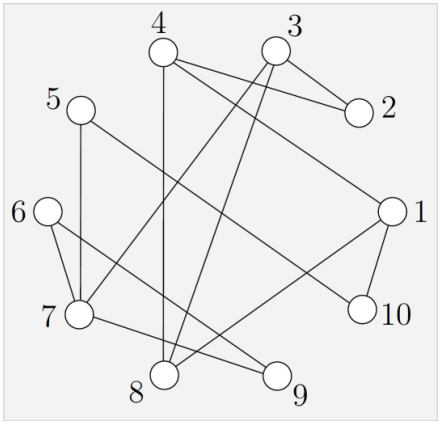
How can we minimize the impact of an attack, that is, maximize the pairwise connectivity of the residual network?

By imposing extra constraints on the attacker, which can be done by employing hardening and diversity.

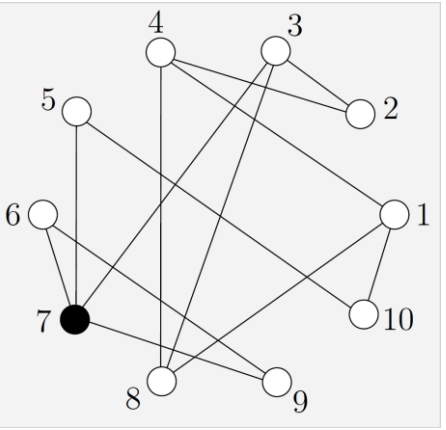
Hardening of nodes:

- A small subset of nodes, say T, is hardened such that these nodes cannot be removed from the network.
- Consequently, attack can be launched only at the nodes that are not hardened.

- Optimal attack of removing two nodes = {1,7}
- Pair-wise connectivity after attack = 0.286



- **Node 7 is hardened**
- Optimal attack = {3,10}
- Pair-wise connectivity after attack = 0.429

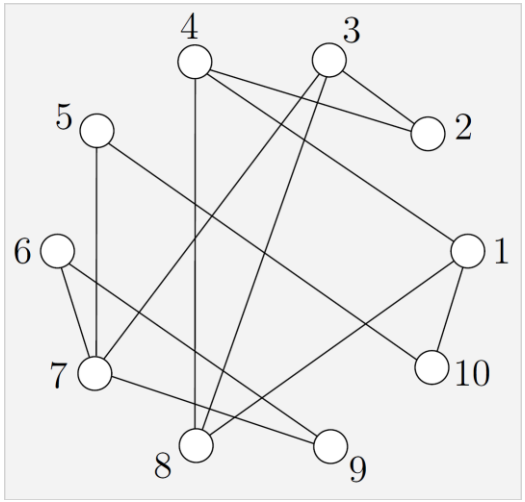


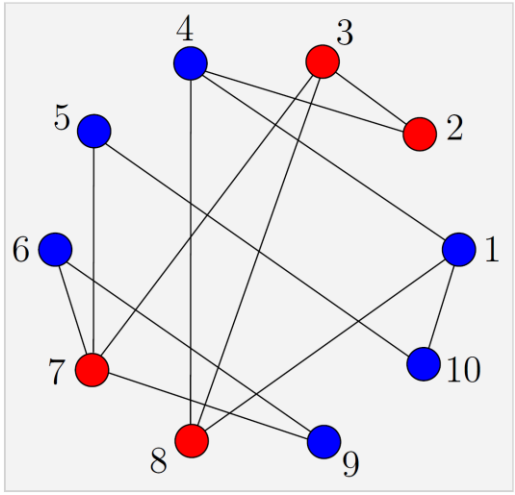
Hardening and Diversity

Diversifying nodes:

- Consider that nodes are heterogeneous and are of multiple types.
- Set of node types: $D = \{D_1, D_2, \ldots, D_d\}$.
- Each node belongs to one of the types in D.
- An attacker can only attack nodes that belong to the same type.

- Optimal attack of removing two nodes = {1,7}
- Pair-wise connectivity after attack = 0.286



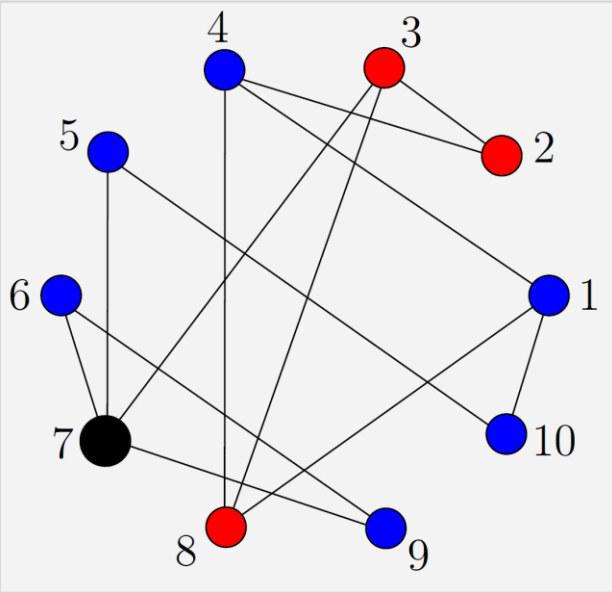


Two types of nodes, red and blue.

- Optimal attack = {2,7}
- Pair-wise connectivity after attack = **0.571**

Hardening and Diversity

- By combining hardening and diversity, pair-wise connectivity resulting after an optimal attack can be further improved.
- Consider **two node types, one hardened node**, and an attack consisting of removing two nodes.

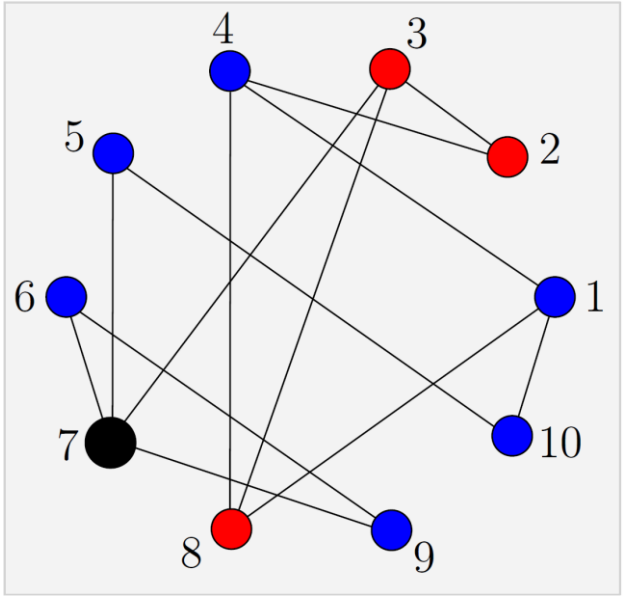


- Two types of nodes, red and blue.
- Node 7 is **hardened**.
- Optimal attack consists of removing blue nodes {1,5}
- Resulting pair-wise connectivity is **0.75**
- Without hardening and diversity, pair-wise connectivity would be **0.286.**

Defender's Objective

1. How to assign **colors (types)** to nodes (diversifying nodes)?
2. Which nodes should be **hardened** (trustiness)?

such that the pairwise connectivity of the residual graph after optimal attack is maximized.



Problem Complexity

Network Robustness Maximization Problem (Decision Version):

Given a network graph $G(V, E)$,

- number of nodes that can be trusted t,
- a set of node types D,
- an attacker budget a, and
- a threshold pairwise connectivity P^* ,

find a set of trusted nodes $T \subseteq V$ such that $|T| \leq t$ and a node type assignment Γ such that the pairwise connectivity of residual graph after optimal attack is at least P^*.

Theorem: Network Robustness Maximization Problem is **NP-hard.**

(Reduction from the Set Cover Problem)

Simulated Annealing Heuristics for the Attacker Problem

1. **Given:** A colored graph G with few trusted nodes.
2. **Initialize:**
3. Randomly select a set of '*a*' nodes with the same color *i*, say A_i
4. Compute Pairwise connectivity of G after removing nodes in A_i , say P_i

5. **while** c < iterations ; **do**
6. "*Perturb*" A_i to get a new A^*
7. Compute Pairwise connectivity of G after removing nodes in A^* , say P^*
8. **if** $P^* < P_i$; **then**
9. with "*some probability*", A^* becomes A_i
10. **end if**
11. update "*temperature*" parameter (to compute probability in line **9**)
12. c = c+1
13. **end while**

14. **Return** A_i (attack)

Simulated Annealing Heuristics for the Defender Problem

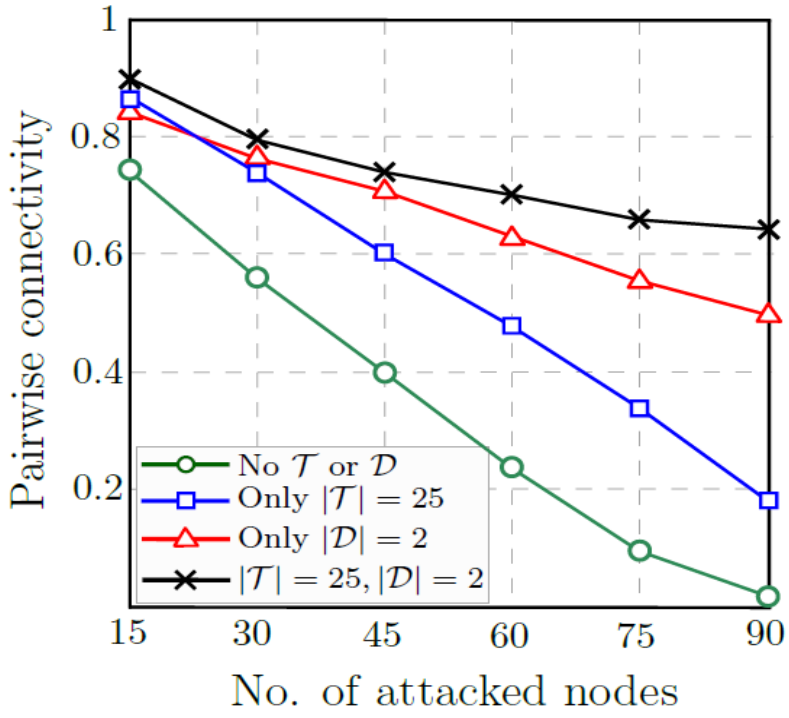
We also present a similar **Simulated Annealing based Heuristic** for the defender

- to **assign colors** to nodes from a given coloring set, and
- to make given number of nodes **trusted**.

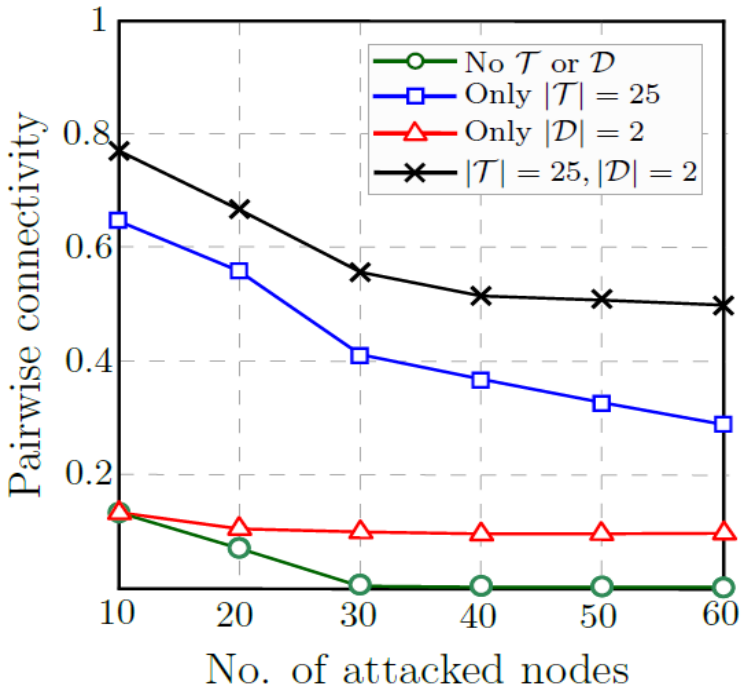
Numerical Results

Two benchmark networks from [1]

- **[ER – 465]** Erdos – Renyi graph with 465 nodes and 699 edges.
- **[BA – 500]** Barabasi – Albert graph with 500 nodes and 499 edges.



[ER – 465]



[BA – 500]

T = Trusted nodes

D = Types of nodes

[1] M. Ventresca, *"Global search algorithms using a combinatorial unranking-based problem representation for the critical node detection problem,"* Computers & Operations Research, 2012

Conclusion

- Instead of adding edges **(redundancy)**, we can improve structural robustness in networks through other approaches.
- By having multiple variants of nodes **(diversity)** and a small number of hardened nodes that are insusceptible to failures **(trustiness)**, we can significantly improve structural robustness.
- Using diverse and trusted nodes, we can **limit attacker's** scope of action by posing extra constraints on it.

Future directions:

- Improving **other robustness and network utility** measures using diversity and trust notions.
- **Efficient algorithms** to assign types and select trusted nodes.

Thank You