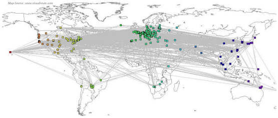
Improving Network Connectivity Using Trusted Nodes and Edges

Waseem Abbas, Aron Laszka,
Yevgeniy Vorobeychik, Xenofon Koutsoukos

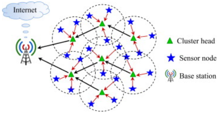
VANDERBILT UNIVERSITY

May 24, 2017

Motivation



internet topology

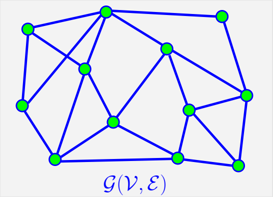


sensor network

infrastructure

social network

- Networks are failure-prone and are vulnerable to attacks that can result in node (edge) removals.
- As a result, connectivity between nodes (required for network operations), might be severely affected.
- We desire networks to be structurally robust, e.g., to remain connected under node (edge) removals.



How can we **improve structural robustness** (e.g., connectivity) of networks?

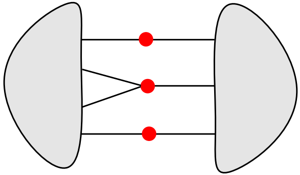
Contributions

- We study network connectivity with trusted nodes and edge.
  - **Main idea:** Connectivity can be significantly improved by selecting a small subsets of nodes (edges) as trusted (insusceptible to failures).
- Definitions and generalization of Menger's theorem.
- Compute connectivity with trusted nodes (edges).
- Compute optimal set of trusted nodes to achieve desired connectivity.
  - problem complexity
  - heuristics
- Numerical evaluation.

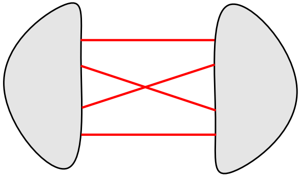
Vertex and Edge Connectivity

k-vertex connected:
Graph remains connected if any set of $(k-1)$ *vertices* are removed.

k-edge connected:
Graph remains connected if any set of $(k-1)$ *edges* are removed.



$k = 3$



$k = 4$

In general, higher connectivity is desired, for instance to improve

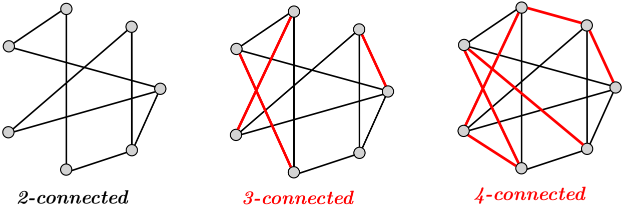
- reliability,
- resilience against failures,
- network routing, etc.

Improving Connectivity through Augmentation

Problem

How can we efficiently *improve* the vertex (edge) connectivity of networks represented as graphs?

Connectivity augmentation: Add minimum number of extra edges strategically to achieve desired network connectivity.



Connectivity augmentation could be prohibitively expensive, or not suitable from security perspective.

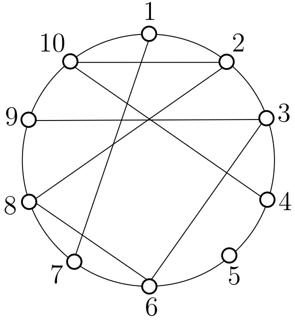
Improving Connectivity through Trusted Nodes and Edges

- **A different approach:**
  - Make a small subset of nodes (edges) **trusted**.
- **Trusted nodes and edges:**
  - They are hardened and are insusceptible to failures.
  - Remain operational at all times.

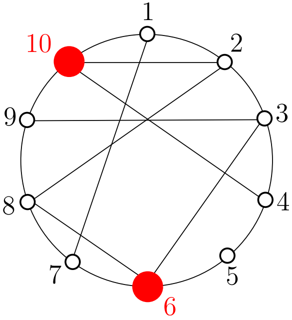
Consequently, the network connectivity can be measured by the number of *non-trusted* nodes (edges) that need to be removed.

Instead of **redundancy** to improve connectivity, we exploit the notion of **trustedness of a small sub-network** to improve connectivity.

Connectivity through Trusted Nodes (Example)

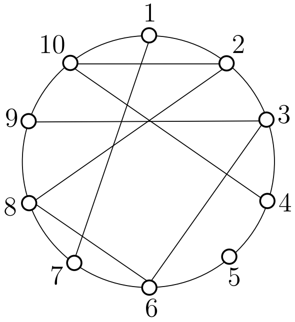


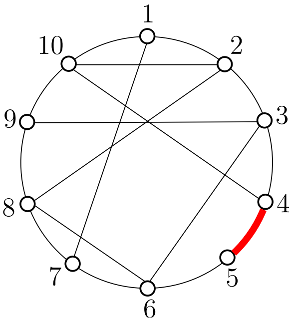
- The graph is 2-vertex connected.



- Nodes 6 and 10 are trusted.
- At least four of the non-trusted nodes need to be removed to disconnect the graph. (4-vertex connected).

Connectivity through Trusted Edges (Example)





- The graph is 2-edge connected.

- Edge $4 \sim 5$ is trusted.
- At least three of the non-trusted edges need to be removed to disconnect the graph. (3-edge connected).

Node and Edge Connectivity with Trusted Nodes

A graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ is **k-vertex connected with trusted nodes** $\mathcal{T}_v \subset \mathcal{V}$ if there does not exist a set of fewer than k vertices in $\mathcal{V} \setminus \mathcal{T}_v$ whose removal disconnects the graph.

A graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ is **k-edge connected with trusted nodes** $\mathcal{T}_e \subset \mathcal{E}$ if there does not exist a set of fewer than k edges in $\mathcal{V} \setminus \mathcal{T}_e$ whose removal disconnects the graph.

Related issues:

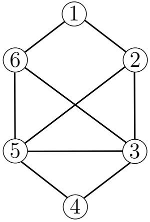
- **theoretical** basis (Menger's type result),
- **computing connectivity** with trusted nodes and edges,
- computing an **optimal set of trusted** nodes and edges.

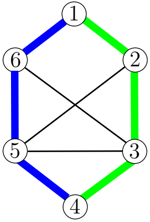
Menger's Theorem (Fundamental Theorem of Connectivity)

The minimum number of nodes (edges) whose removal disconnects two nodes, say u and v, is equal to the maximum number of pairwise node-independent (edge-independent) paths from u to v.

Node (edge) connectivity ⟷ **Number of node (edge) independent paths between any two nodes**

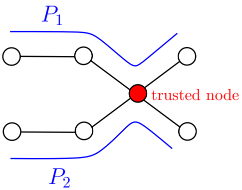
Example:





Menger's Theorem and Connectivity with Trusted Nodes

- **Node-independent paths with trusted nodes**



The only common node in paths P_1 and P_2 is the trusted node.

- **Node trusted path**
  A path with all trusted nodes is a node trusted path.

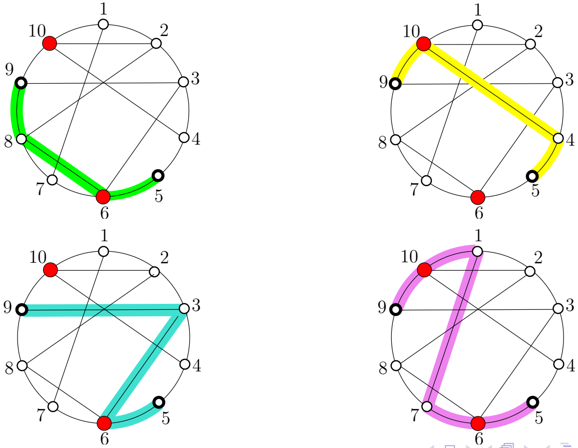
Theorem

Following statements are equivalent:

- $\mathcal{G}(\mathcal{V}, \mathcal{E})$ is k-vertex connected with trusted nodes $\mathcal{T}_v$.
- For any two non-adjacent vertices u and v, either there exists a node-trusted path between them, or there exists at least k paths between them that are node-independent with $\mathcal{T}_v$.

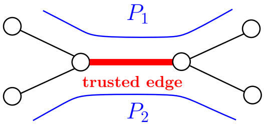
Example:

- The graph is **4-vertex connected** with $\mathcal{T}_v = \{6, 10\}$.
- Four node-independent paths between nodes 5 and 9 are shown.



Menger's Theorem and Connectivity with Trusted Edges

- **Edge-independent paths with trusted edges**



The only common edge in paths P_1 and P_2 is the trusted edge.

- **Edge trusted path**
  A path consisting of only trusted edges.

Theorem

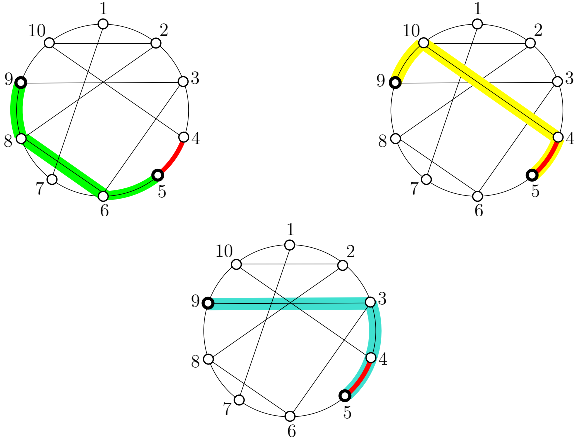
Following statements are equivalent:

- $\mathcal{G}(\mathcal{V}, \mathcal{E})$ is k-edge connected with trusted edges $\mathcal{T}_e$.
- For any two vertices u and v, either there exists an edge-trusted path between them, or there exists at least k paths between them that are edge-independent with $\mathcal{T}_e$.

Menger's Theorem and Connectivity with Trusted Edges

Example:

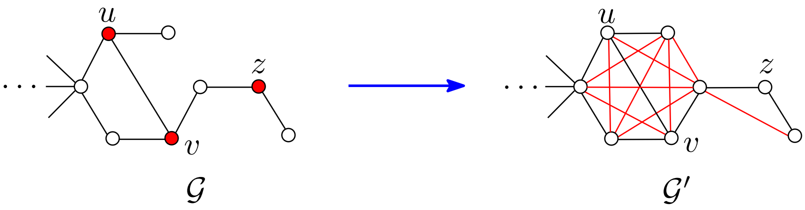
- The graph is **3-edge connected** with $\mathcal{T}_e = \{4 \sim 5\}$.
- Three edge-independent paths between nodes 5 and 9 are shown.



Computing Node Connectivity with Trusted Nodes

From a given $\mathcal{G}(\mathcal{V}, \mathcal{E})$ with trusted nodes $\mathcal{T}_v \subset \mathcal{V}$, obtain a new graph $\mathcal{G}'(\mathcal{V}, \mathcal{E}')$ as follows:

- If two nodes in $\mathcal{G}$ are connected by a trusted node, or by a node trusted path; then these nodes are adjacent in $\mathcal{G}'$.



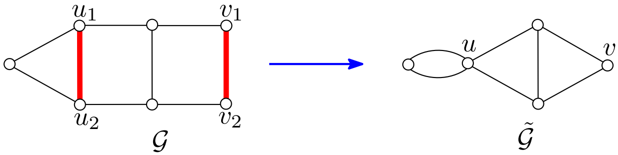
Proposition

$$\text{Vertex connectivity of } \mathcal{G} \text{ with } \mathcal{T}_v \quad = \quad \text{Vertex connectivity of } \mathcal{G}'$$

Computing Edge Connectivity with Trusted Edges

From a given $\mathcal{G}(\mathcal{V}, \mathcal{E})$ with trusted edges $\mathcal{T}_e \subset \mathcal{E}$, obtain a new graph $\tilde{\mathcal{G}}(\tilde{\mathcal{V}}, \tilde{\mathcal{E}})$ as follows:

- If two nodes in $\mathcal{G}$ are connected by a trusted edge, or by an edge trusted path; then identify these two nodes in $\tilde{\mathcal{G}}$.



Proposition

$$\text{Edge connectivity of } \mathcal{G} \text{ with } \mathcal{T}_e \quad = \quad \text{Edge connectivity of } \tilde{\mathcal{G}}$$

Computing Trusted Nodes – Problem Complexity

Finding a minimum set of trusted nodes is computationally hard.

Theorem

Given

- a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$,
- desired connectivity k', and
- the number of trusted nodes T;

then determining if there exists a $\mathcal{T}_v \subset \mathcal{V}$ such that $|\mathcal{T}_v| \leq T$ and $\mathcal{G}$ is k'-connected with $\mathcal{T}_v$ is **NP-hard**.

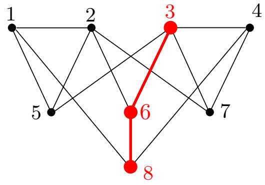
Computing Trusted Nodes – Heuristics

Basic idea:

- Begin with a "sufficient" set of trusted nodes.
- Iteratively remove nodes from the set until no further node can be removed.

Connected dominating set (Γ):

- Every node is either in Γ, or is adjacent to some node in Γ.
- Nodes in Γ induce a connected subgraph.



Computing Trusted Nodes – Heuristics

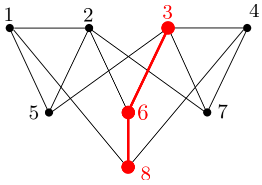
For any desired k-vertex connectivity with trusted nodes $\mathcal{T}_v$,

$$|\mathcal{T}_\mathbf{v}| \leq \gamma_{\mathcal{G}},$$

where $\gamma_\mathcal{G}$ is the connected domination number of $\mathcal{G}$.

Trusted nodes for vertex connectivity

Input: $\mathcal{G}(\mathcal{V}, \mathcal{E})$, k'
Output: $\mathcal{T}_v \subseteq \mathcal{V}$
$\Gamma \leftarrow \texttt{Conn\_Dom\_Set}(\mathcal{G})$
$\mathcal{T}_v \leftarrow \Gamma$
for $i = 1$ to $|\Gamma|$ **do**
  $v \leftarrow \texttt{V\_Conn\_Trust}(\mathcal{G}, \mathcal{T}_v \setminus \{\Gamma(i)\})$
  if $v \geq k'$ **do**
    $\mathcal{T}_v \leftarrow \mathcal{T}_v \setminus \{\Gamma(i)\}$
  end if
end for



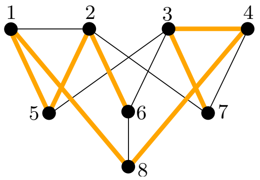
Computing Trusted Edges – Heuristics

- If $\mathcal{T}_e$ is a set of edges in a spanning tree, then edge-trusted path exists between any two nodes.
- Consequently, for any desired k-edge connectivity with trusted edges $\mathcal{T}_e$,

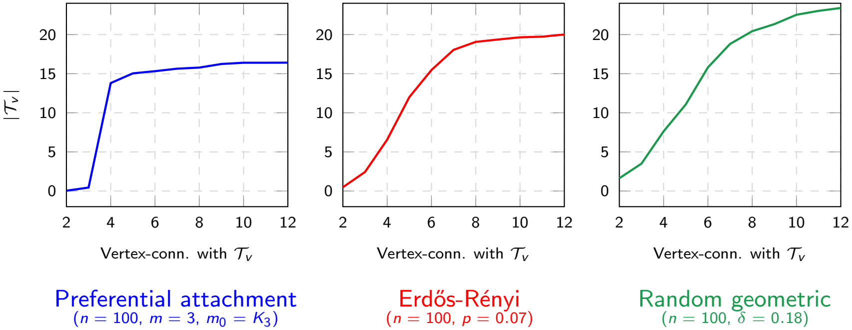
$$|\mathcal{T}_\mathbf{e}| \leq \boldsymbol{n} - \mathbf{1}.$$

Trusted edges for edge connectivity

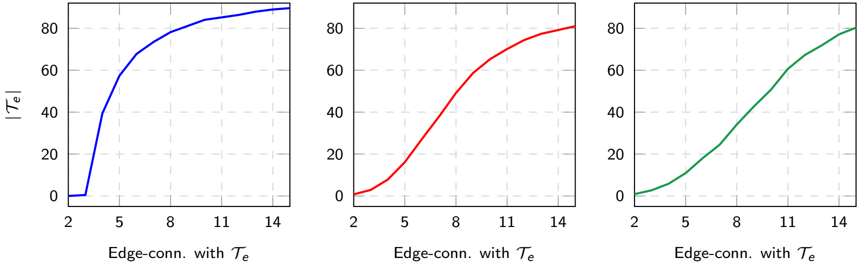
Input: $\mathcal{G}(\mathcal{V}, \mathcal{E})$, k'
Output: $\mathcal{T}_e \subseteq \mathcal{E}$
$\mathcal{E}' \leftarrow$ `Min_Span_Tree`$(\mathcal{G})$
$\mathcal{T}_e \leftarrow \mathcal{E}'$
for $i = 1$ to $|\mathcal{E}'|$ **do**
    $e \leftarrow$ `E_Conn_Trust`$(\mathcal{G}, \mathcal{T}_e \setminus \{\mathcal{E}'(i)\})$
    if $e \geq k'$ **do**
        $\mathcal{T}_e \leftarrow \mathcal{T}_e \setminus \{\mathcal{E}'(i)\}$
    end if
end for



Vertex connectivity with trusted nodes



Edge connectivity with trusted edges



Preferential attachment
($n = 100,\ m = 3,\ m_0 = K_3$)

Erdős-Rényi
($n = 100,\ p = 0.07$)

Random geometric
($n = 100,\ \delta = 0.18$)

Conclusion

- A subset of nodes (edges) can be hardened and can be made trusted.
- Network connectivity can be improved through these trusted components, even in sparse networks, without adding extra links.
- By controlling the number and location of trusted nodes and edges, any desired network connectivity can be obtained.

Future directions:

- Improving other structural robustness measures using trusted components.
- Efficient algorithms to compute trusted nodes and edges.
- An integrated strategy that combines both redundancy and trustedness to improve structural robustness in networks.

Acknowledgments

- National Science Foundation (CNS-1238959)
- Air Force Research Laboratory (FA 8750-14-2-0180)

Thank You