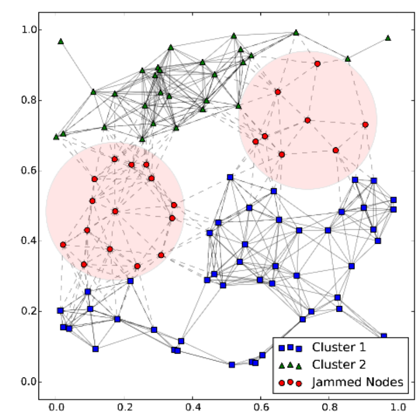
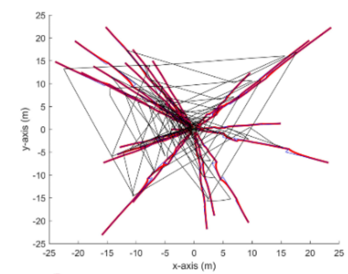
Network Systems

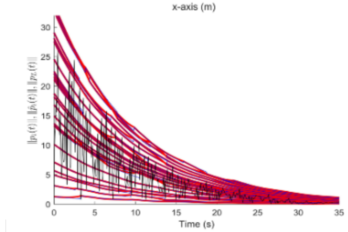
Network Systems

- Design and analysis challenges for both controlling agents within a network (stochastic time-varying and random graph models) and controlling agents over a network
- Determining conditions under which random communication graphs attain required connectivity properties and positioning agents to achieve network objectives (e.g., jamming adversarial networks)
- Develop models where the control system can adapt in real time as service degrades
- Develop control techniques that allow a system to adapt its operation and use of network resources based on QoS that the network is able to provide

RT3 will develop analysis, design, and synthesis methods for agents *within* a network and *over* a network to generalize existing graph theory-based methods and improve the interface and adaptability between controls and communications







UF UNIVERSITY of FLORIDA

Duke UNIVERSITY

TEXAS The University of Texas at Austin

UC SANTA CRUZ

Network Systems

- Optimization of agent placement in wireless networks
- Using network QoS information to optimize control system operation
- Closing the loop between control and network: optimization of network operation and feedback of predicted communication performance

Network Systems

- Control over networks with intermittent connectivity
  - Byzantine adversaries that disrupt the network topology

- For networks with time-varying links, under what conditions will the networks be sufficiently connected for control?

Operation in Complex RF Environments

- Mobile autonomous agents require wireless communication for control and coordination
- RF environment for such agents may suffer from many challenging impairments:
  - Time-varying multipath fading
  - Shadowing from terrain and buildings
  - Interference
  - Hostile jamming

UF UNIVERSITY of FLORIDA Duke UNIVERSITY TEXAS The University of Texas at Austin UC SANTA CRUZ

Network Optimization

- Assured operation of networked systems requires optimization of the network topology and/or placement of agents
  - Topology may be optimized to ensure sufficient connectivity
  - Databases and computation resources may be located to ensure access
  - Jammers may be placed to disrupt adversary communications
  - Mobile communication nodes may be placed to reinforce network structure to make it robust to jamming/interference

- Graphs make useful abstraction for representing such networks
- Vertices represent communicators
- Edges represent wireless links
    - Possibly time-varying!

Example: Jamming and Countermeasures

- Recently developed algorithm to find optimal jammer placements to partition a wireless network
- Objective is to minimize # of jammers required

- Challenges:
  - Connectivity is a global property of graphs
  - Jammers may affect communication in multiple ways (reduction in SINR, input saturation, etc.)
  - Jammers may be placed anywhere is Euclidean space
- Key insights
  - If limit jammers to discrete set of locations, optimal solution can be found via integer linear program and suboptimal, fast solutions can be found via multiresolution graph cut
  - Many possible jammer locations in Euclidean space jam the same set of nodes: can reduce continuous space to discrete

Let $\mathcal{G}(\mathcal{V}, \mathcal{E})$ be the network we aim to partition by placing jammers at positions in $\mathcal{J}$

Let $\mathcal{H} = \mathcal{H}(\mathcal{V}', \mathcal{E}'; \mathcal{G}, \mathcal{J})$ be the residual network after jammer placement

$$\mathcal{V}' = \mathcal{V} \backslash \{ v \in \mathcal{V} \mid d_E(u, v) \leq r_j, u \in \mathcal{J} \}$$
$$\mathcal{E}' = \mathcal{E} \backslash \{ (u, v) \mid u \in \mathcal{V} \backslash \mathcal{V}' \text{ or } v \in \mathcal{V} \backslash \mathcal{V}' \}$$

For the remaining nodes in $\mathcal{V}'$, we aim to find a partition

$$\Gamma_K(\mathcal{H}) = \left\{\mathcal{V}_1, \mathcal{V}_2, \dots, \mathcal{V}_K \subset \mathcal{V}'\right\}$$

$$\begin{aligned} \text{s.t. } & 0 < |\mathcal{V}_i| \leq b_i & i = 1, \dots, K \\ & \mathcal{V}_i \cap \mathcal{V}_j = \emptyset & i \neq j \\ & \{(u, v) \mid u \in \mathcal{V}_i, v \in \mathcal{V}_j, i \neq j\} \cap \mathcal{E}' = \emptyset & (u, v) \in \mathcal{E} \end{aligned}$$

- K is the minimum number of disconnected clusters
- b_i bounds the number of nodes in cluster i
- $b_k = \left\lceil \frac{|\mathcal{V}|}{K} \right\rceil$ to simplify exposition

- Construct Candidate Jammer Locations (CJLs) using sequence of searches:
  - For each node, find neighborhood with radius twice the jamming radius (r_j)
  - Use depth-first search (DFS) to find every possible subset of that neighborhood that has radius < r_j
  - Eliminate duplicate solutions

DFS

Data: $\mathcal{G}(\mathcal{V},\mathcal{E})$, r_j, γ
Result: $\mathcal{D} = \{\mathbf{d_i} = ((x_i, y_i, r_i), \mathbf{v_i})\}$

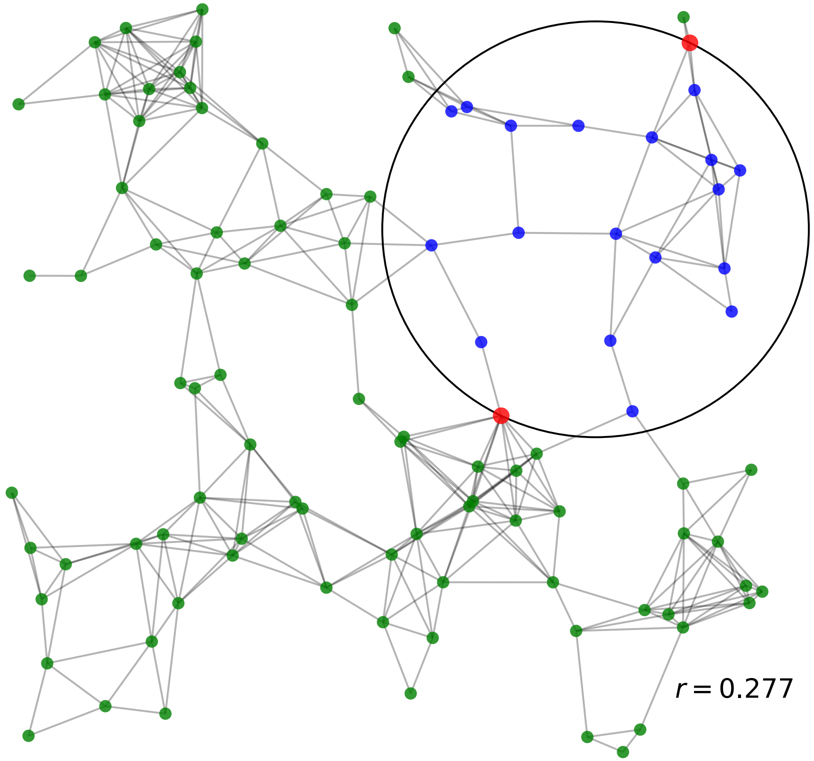
```
create empty stack S and solution set D
foreach v_i ∈ V do
    find N_{2r_j}(v_i) and push to S
    while S is not empty do
        v = S.pop()
        if ∃d_i = ((x_i, y_i, r_i), v_i) ∈ D ∋ v = v_i then
            continue
        end
        calculate (x, y, r) = md(v)
        if r > γ then
            find {b_1, b_2, ...} = B((x, y, r), v)
            foreach b_i ∈ {b_1, b_2, ...} do
                push v\b_i into S
            end
        else
            add ((x, y, r), v) to D
        end
    end
end
```

Remove Redundant Solutions

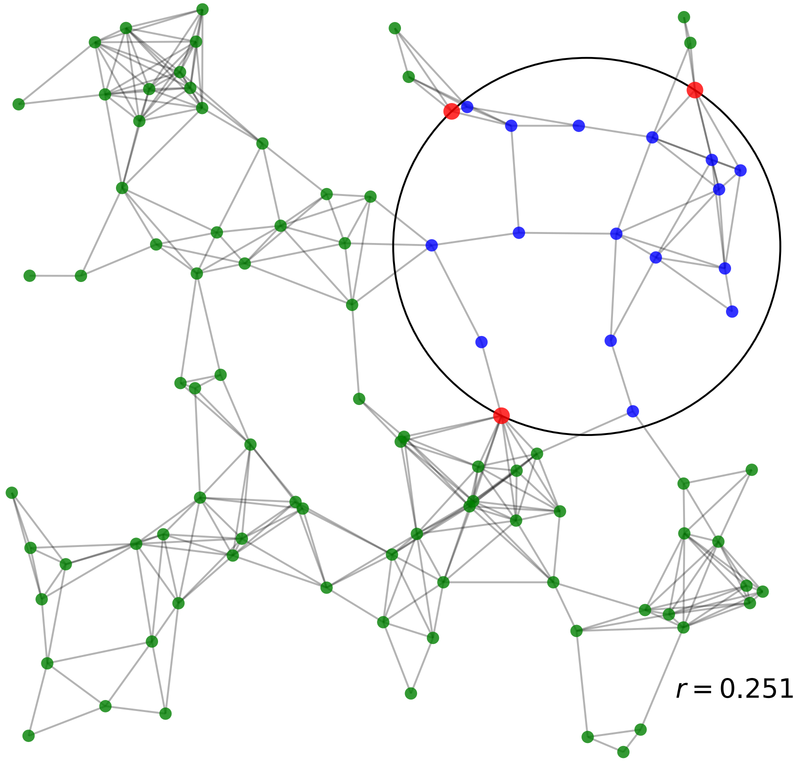
Data: $\mathcal{G}(\mathcal{V},\mathcal{E})$, r_j, γ
Result: $\mathcal{D} = \{\mathbf{d_i} = ((x_i, y_i, r_i), \mathbf{v_i})\}$

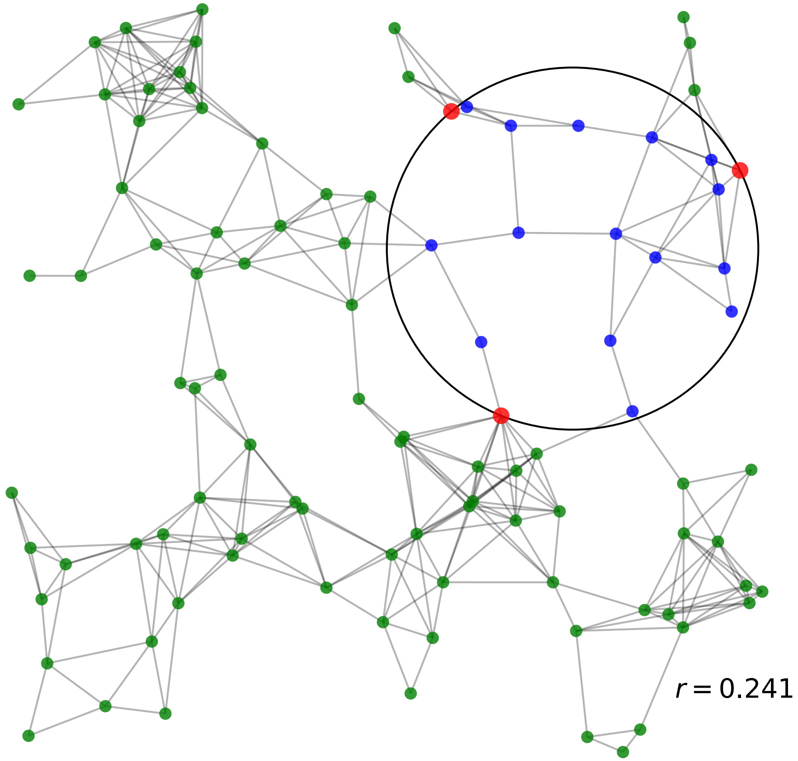
```
create empty stack S and solution set D
foreach d_i ∈ D do
    foreach d_j ∈ D do
        if i ≠ j and d_j ≤ d_i then
            remove d_j from D
        end
    end
end
```

- *Proved that search over CJLs is sufficient to find a minimum cardinality set of jammer locations to partition network*

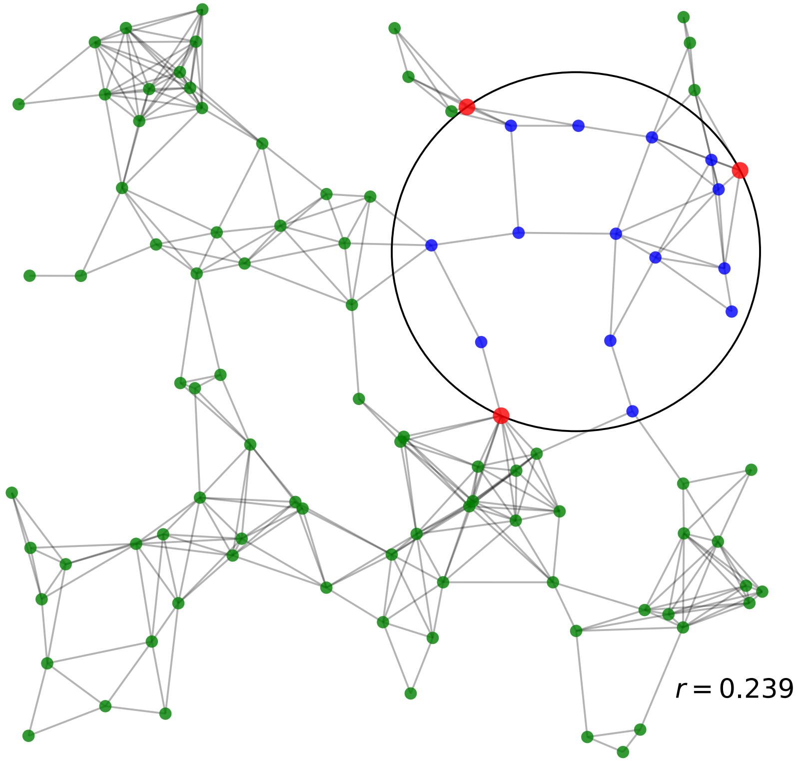


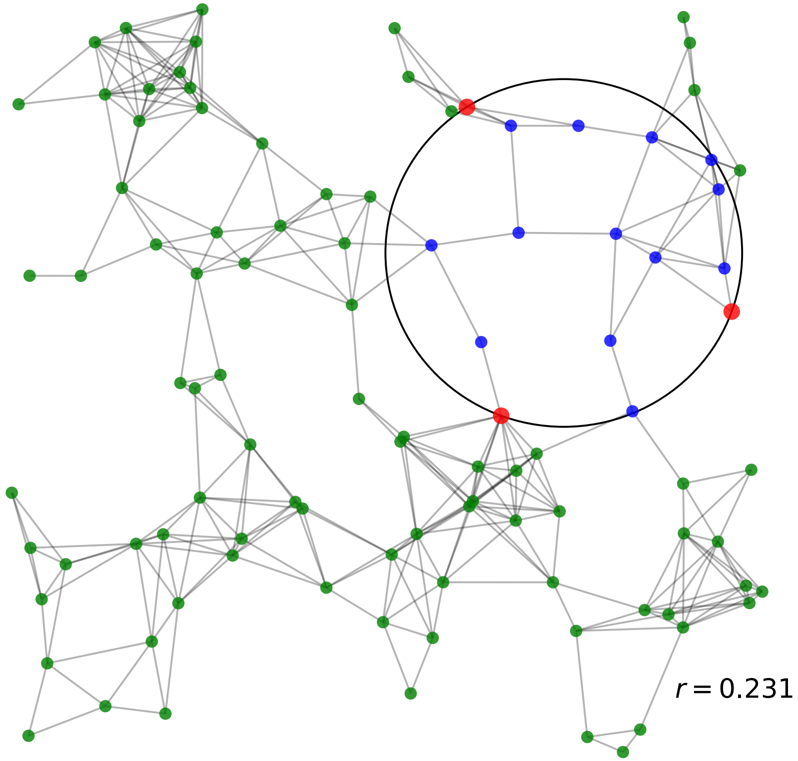
$r = 0.277$

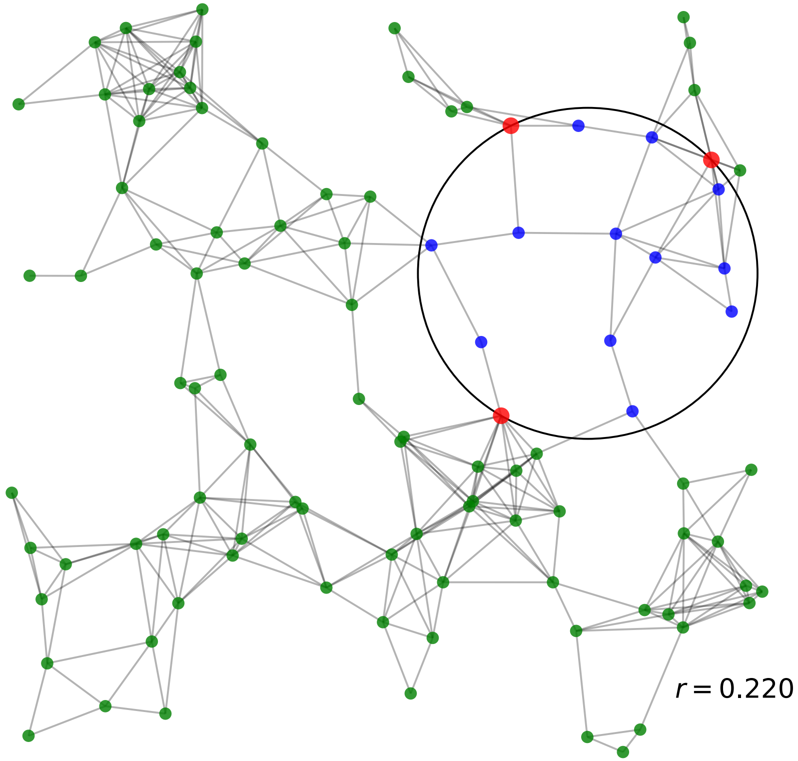

$r = 0.251$

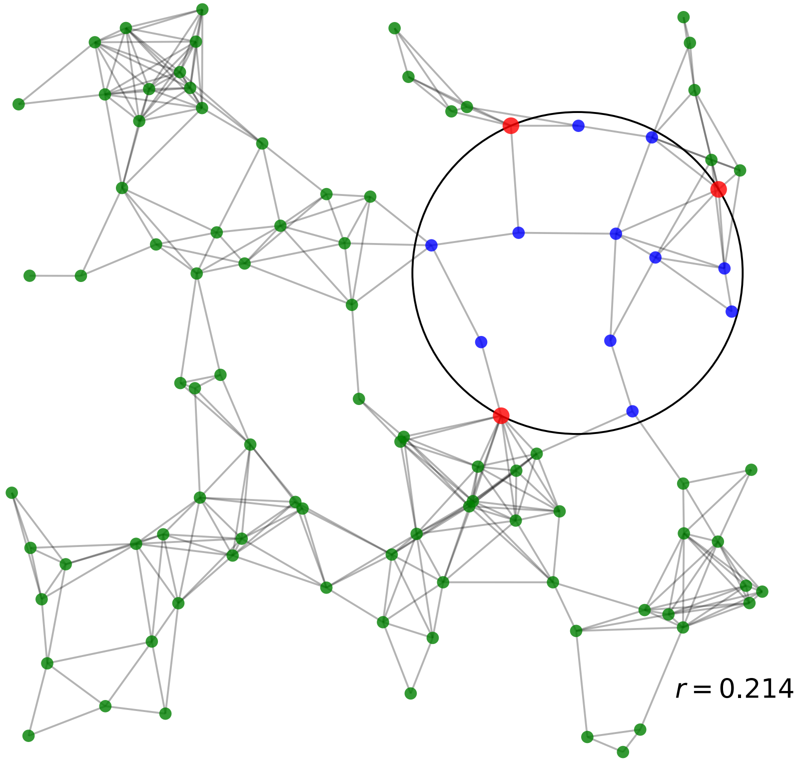


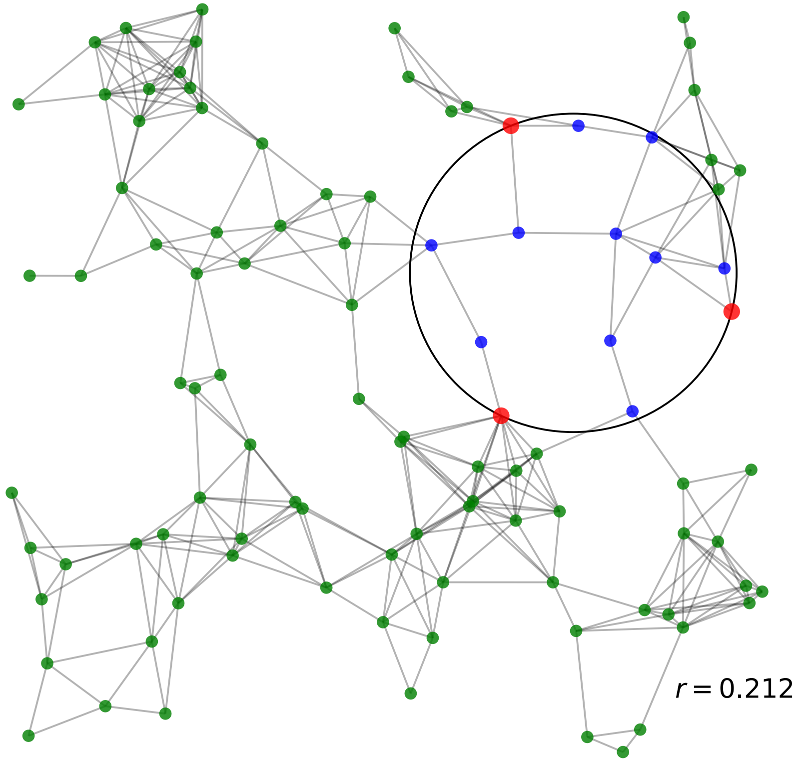
$r = 0.241$

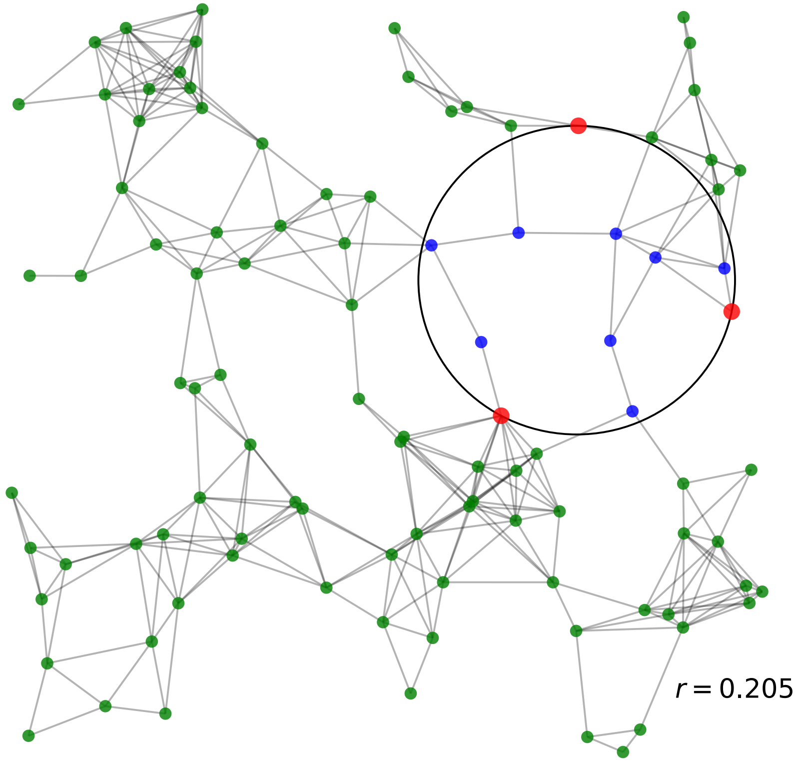

$r = 0.239$

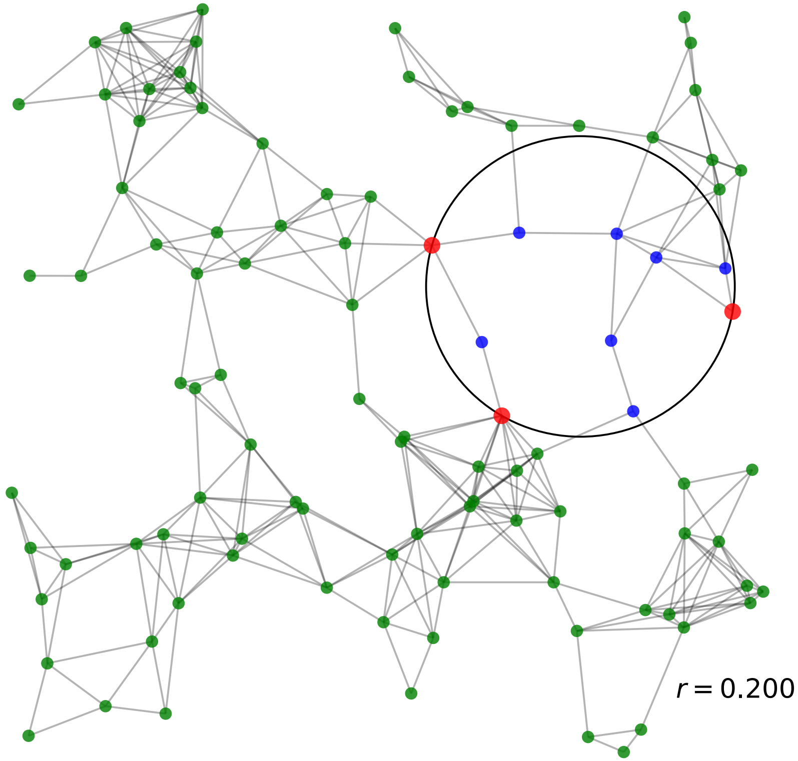

r = 0.231


$r = 0.220$

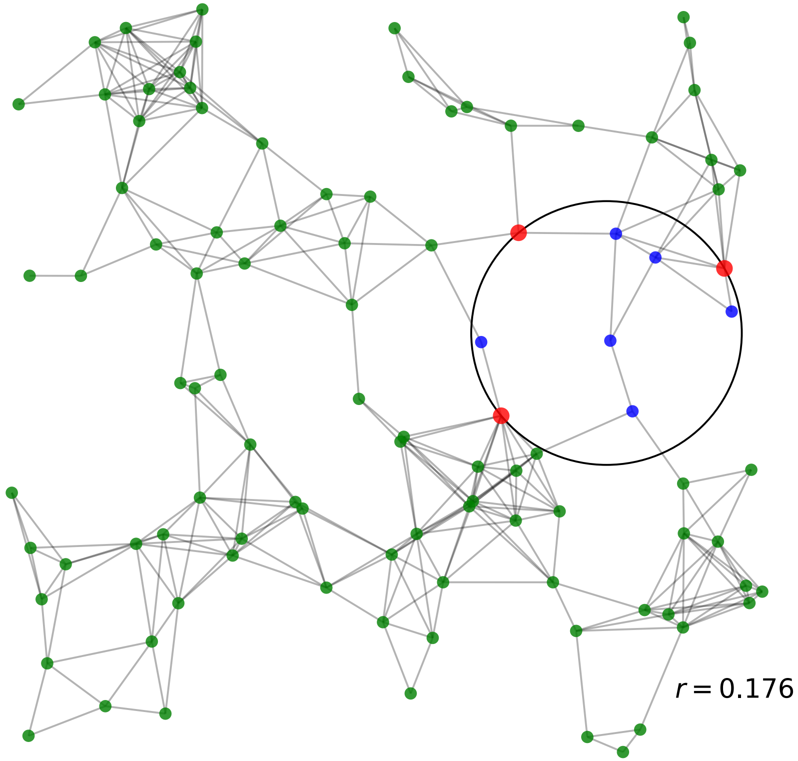

$r = 0.214$

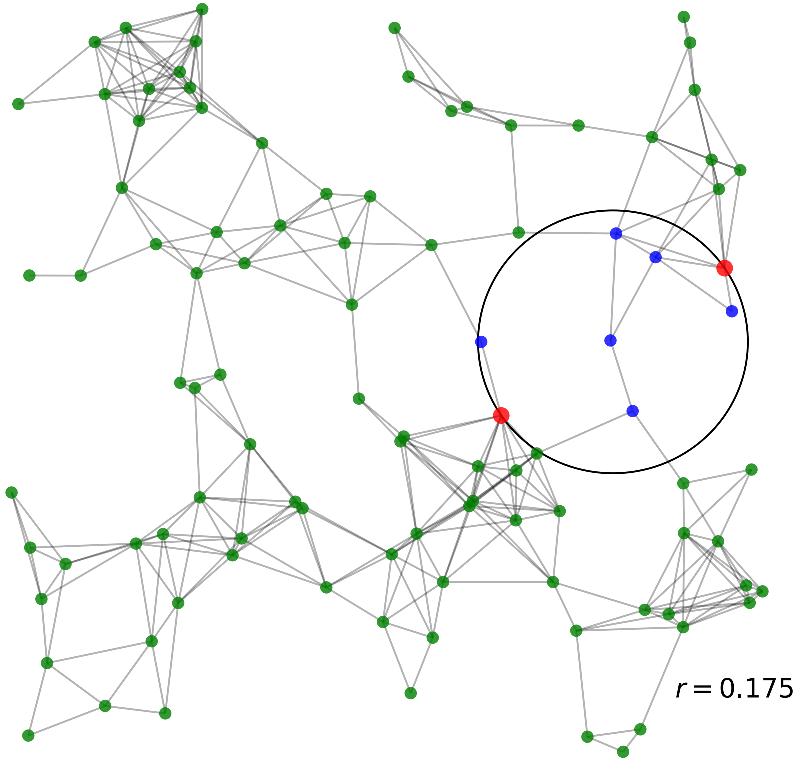

$r = 0.212$

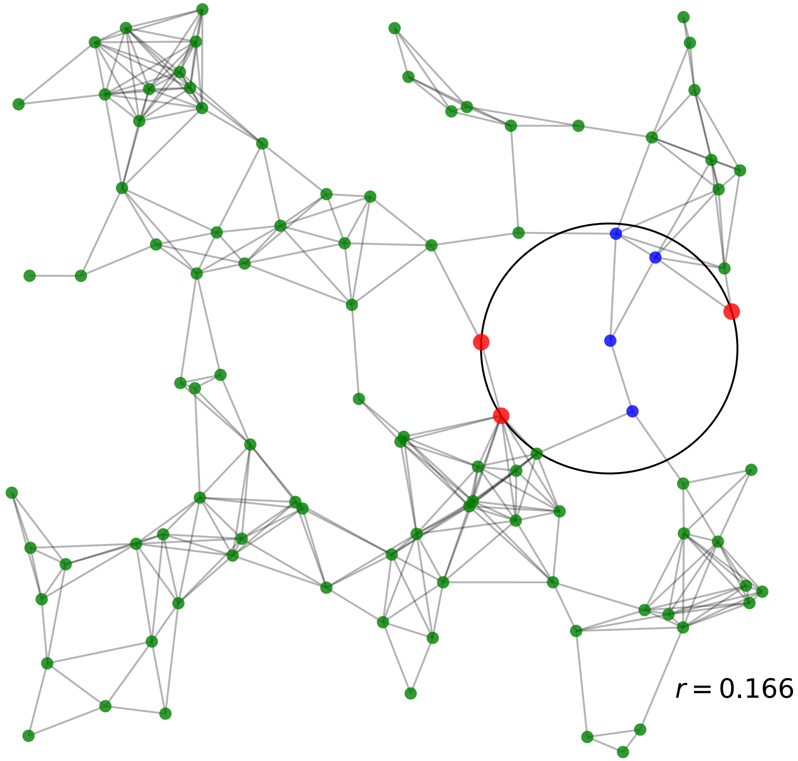

$r = 0.205$

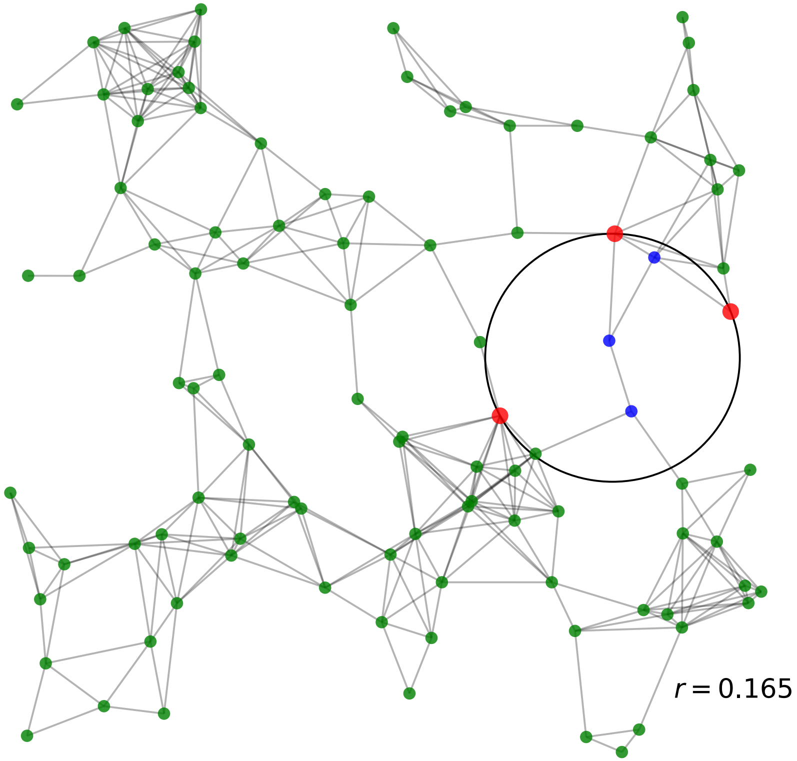


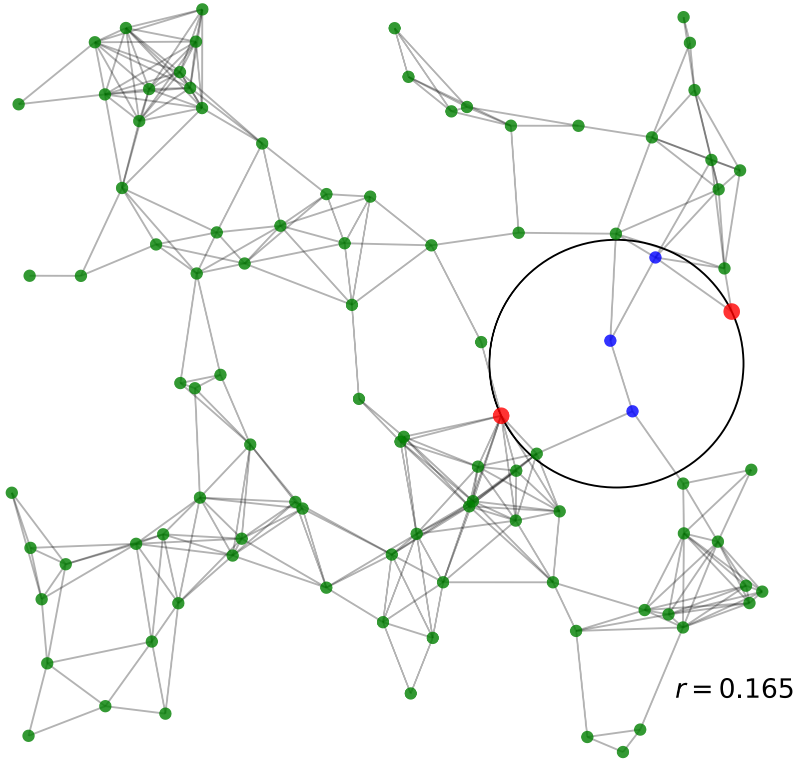
$r = 0.200$


$r = 0.176$

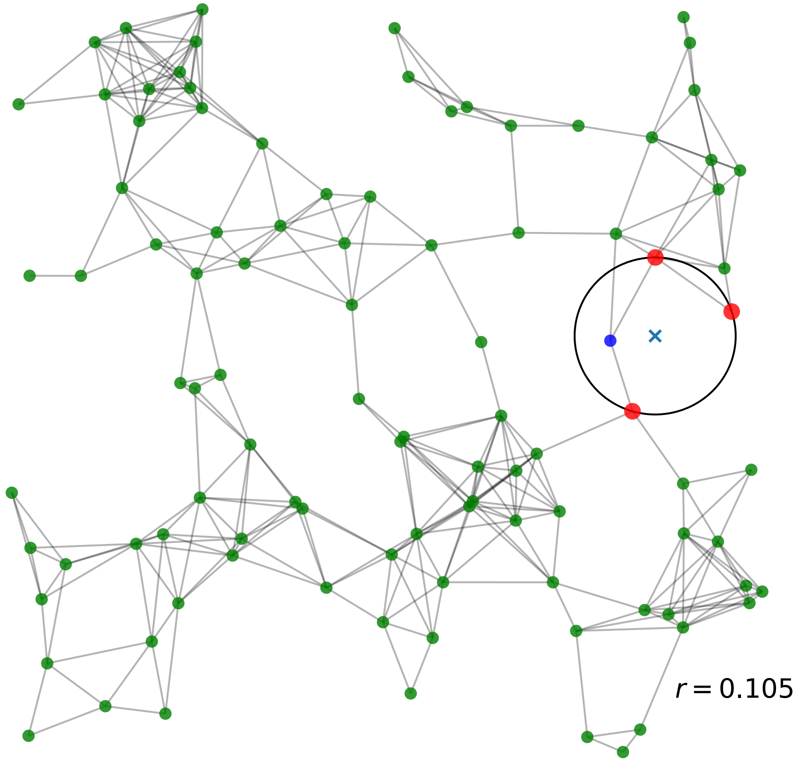

$r = 0.175$

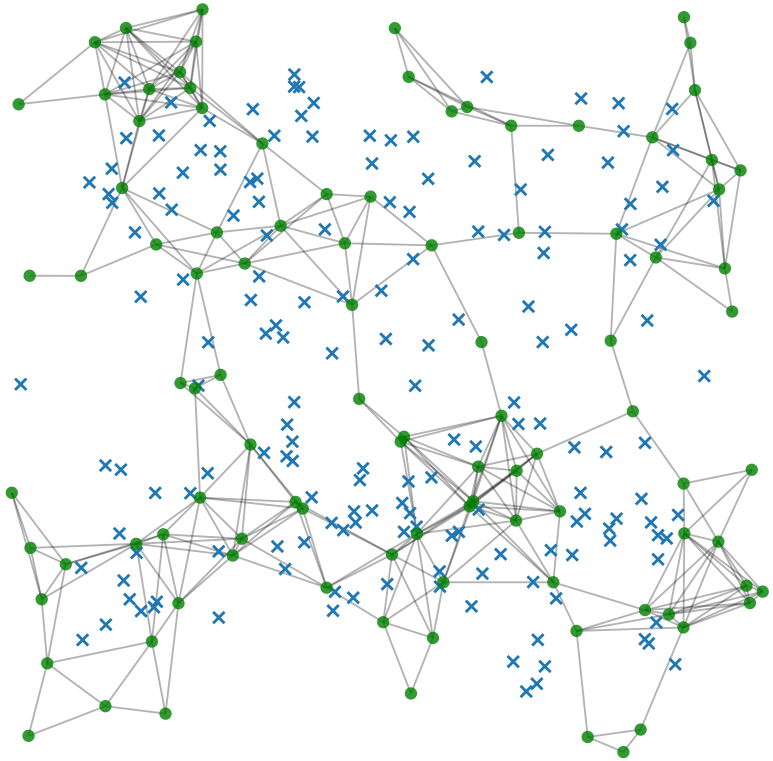

$r = 0.166$


$r = 0.165$



$r = 0.165$


$r = 0.105$



$$\min \quad \sum_{i=1}^{n} j_i$$

Minimize number of jammers

$$\min \quad \sum_{i=1}^{n} j_i$$

$$\text{s. t.} \quad \sum_{k=1}^{K} x_i^{(k)} = 1 \qquad i = 1, \ldots, N$$

$x_i^k = 1$ if node i belongs to cluster k and 0 otherwise. Every node belongs to exactly one cluster

$$\min \quad \sum_{i=1}^{n} j_i$$

$$\text{s. t.} \quad \sum_{k=1}^{K} x_i^{(k)} = 1 \qquad i = 1, \ldots, N$$

$$- \sum_{i \in V} j_i \leq -1$$

At least one jammer is deployed

$$\min \quad \sum_{i=1}^{n} j_i$$

$$\text{s. t.} \quad \sum_{k=1}^{K} x_i^{(k)} = 1 \qquad i = 1, \ldots, N$$

$$- \sum_{i \in V} j_i \leq -1$$

$$- \sum_{i \in V} x_i^{(k)} \leq -1 \qquad k = 1, \ldots, K$$

Every cluster contains at least one node

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{n} j_i \\
\text{s.t.} \quad & \sum_{k=1}^{K} x_i^{(k)} = 1 && i = 1, \ldots, N \\
& -\sum_{i \in V} j_i \leq -1 \\
& -\sum_{i \in V} x_i^{(k)} \leq -1 && k = 1, \ldots, K \\
& \sum_{i \in V} x_i^{(k)} \leq b_k && k = 1
\end{aligned}
$$

Cluster must be smaller than specified size

$$
\begin{aligned}
\text{min} \quad & \sum_{i=1}^{n} j_i \\
\text{s. t.} \quad & \sum_{k=1}^{K} x_i^{(k)} = 1 && i = 1, \ldots, N \\
& -\sum_{i \in V} j_i \leq -1 \\
& -\sum_{i \in V} x_i^{(k)} \leq -1 && k = 1, \ldots, K \\
& \sum_{i \in V} x_i^{(k)} \leq b_k && k = 1 \\
& \sum_{i \in V} x_i^{(k)} - \sum_{i \in V} x_i^{(k-1)} \leq 0 && k = 2, \ldots, K
\end{aligned}
$$

Cluster size is in non-increasing order to avoid duplicated solutions

$$
\begin{array}{lll}
\min & \sum_{i=1}^{n} j_i & \\
\text{s. t.} & \sum_{k=1}^{K} x_i^{(k)} = 1 & i = 1, \ldots, N \\
& -\sum_{i \in V} j_i \leq -1 & \\
& -\sum_{i \in V} x_i^{(k)} \leq -1 & k = 1, \ldots, K \\
& \sum_{i \in V} x_i^{(k)} \leq b_k & k = 1 \\
& \sum_{i \in V} x_i^{(k)} - \sum_{i \in V} x_i^{(k-1)} \leq 0 & k = 2, \ldots, K \\
& x_i^{(0)} - \sum_{m \in V} j_m A_{mi}^{(J)} \leq 0 & i \in V
\end{array}
$$

Nodes within jamming range of a jammer is assigned to cluster 0

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{n} j_i \\
\text{s. t.} \quad & \sum_{k=1}^{K} x_i^{(k)} = 1 && i = 1, \ldots, N \\
& -\sum_{i \in V} j_i \le -1 \\
& -\sum_{i \in V} x_i^{(k)} \le -1 && k = 1, \ldots, K \\
& \sum_{i \in V} x_i^{(k)} \le b_k && k = 1 \\
& \sum_{i \in V} x_i^{(k)} - \sum_{i \in V} x_i^{(k-1)} \le 0 && k = 2, \ldots, K \\
& x_i^{(0)} - \sum_{m \in V} j_m A_{mi}^{(J)} \le 0 && i \in V \\
& x_i^{(k)} - x_j^{(k)} - x_i^{(0)} - x_j^{(0)} \le 0 && (i, j) \in E
\end{aligned}
$$

Nodes that share an edge must belong to same cluster unless one belongs to cluster 0

- Simulated networks that are Random Geometric Graphs
- Equal communication and jamming radii equal to 15% of side of 100-node network
- Network dimensions scale with $\sqrt{n}$, where

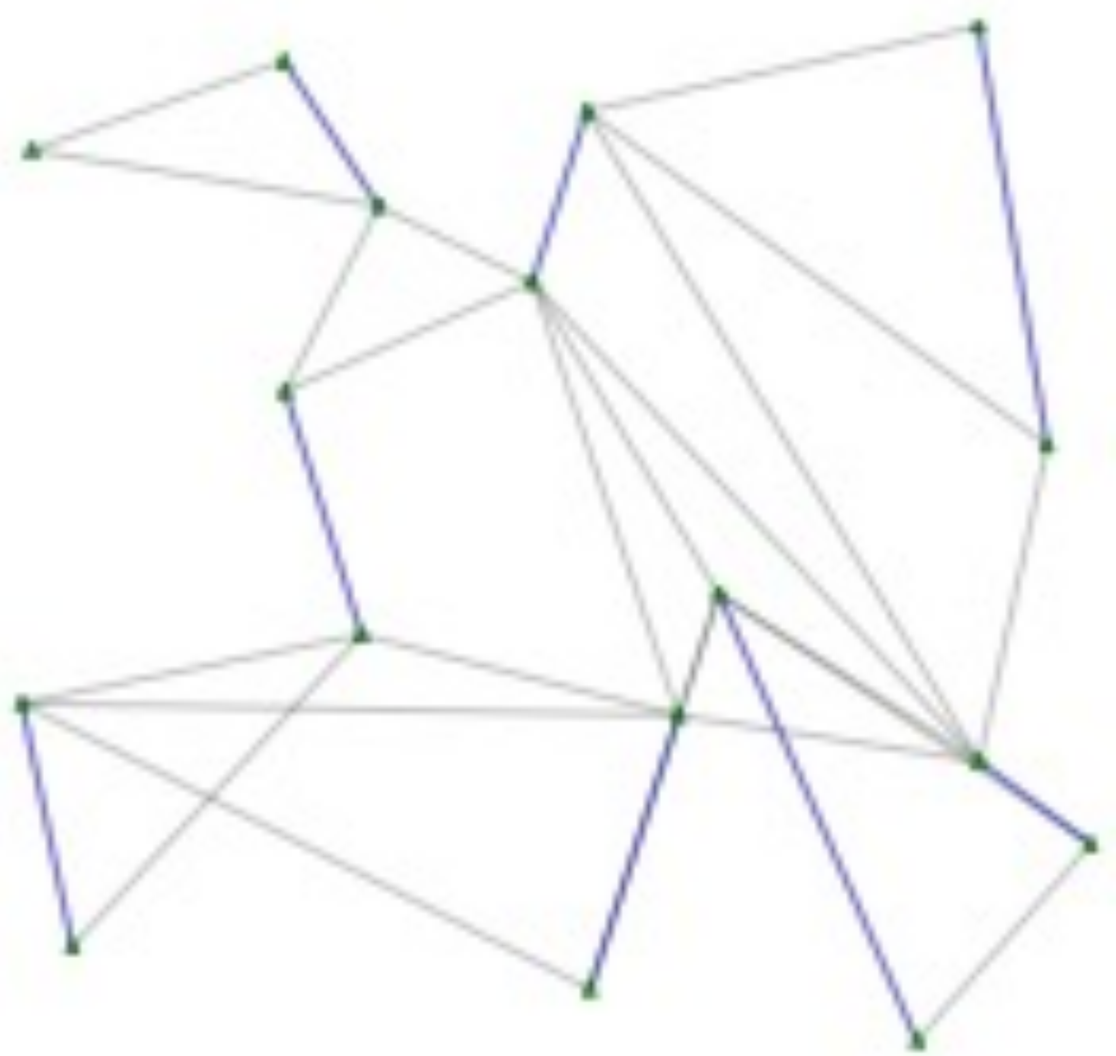
$$n = \text{number of radios}$$

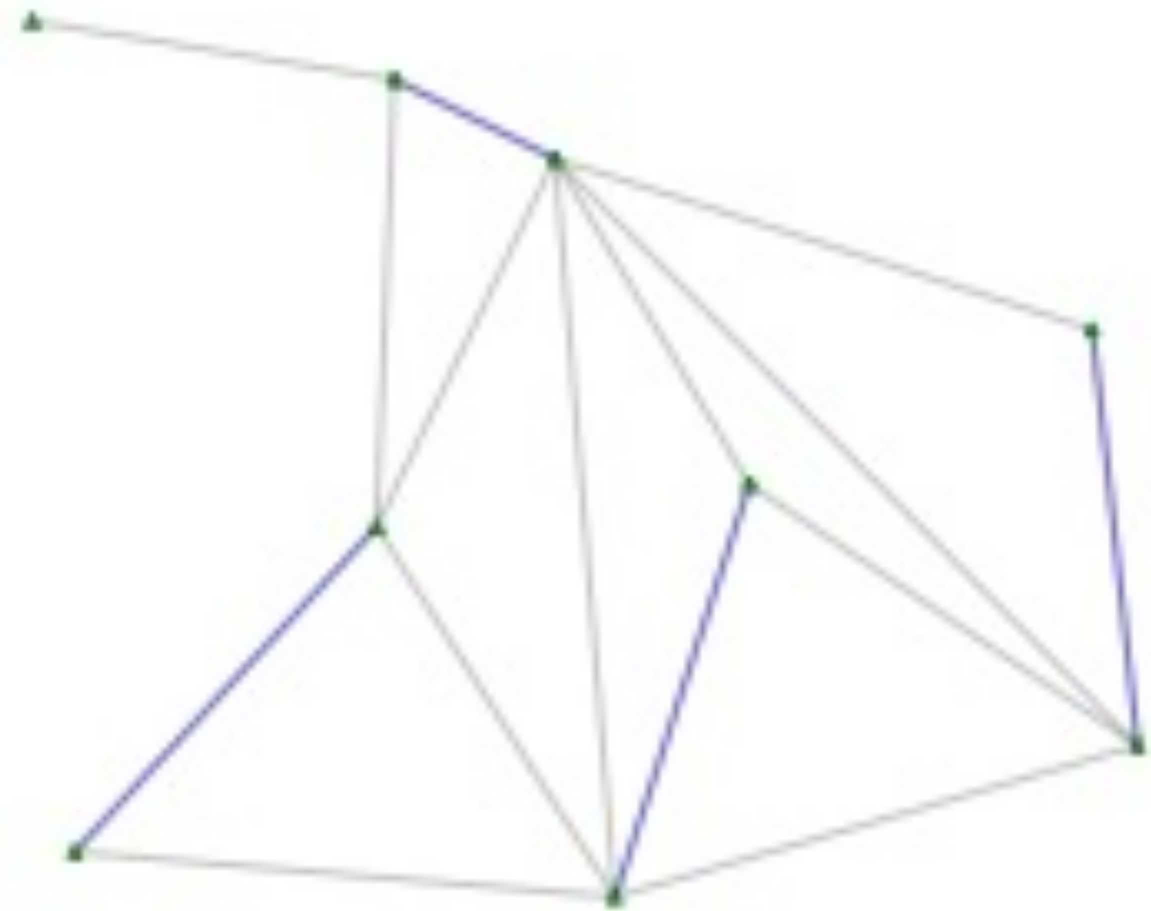
UF UNIVERSITY of FLORIDA · Duke UNIVERSITY · TEXAS The University of Texas at Austin · UC SANTA CRUZ

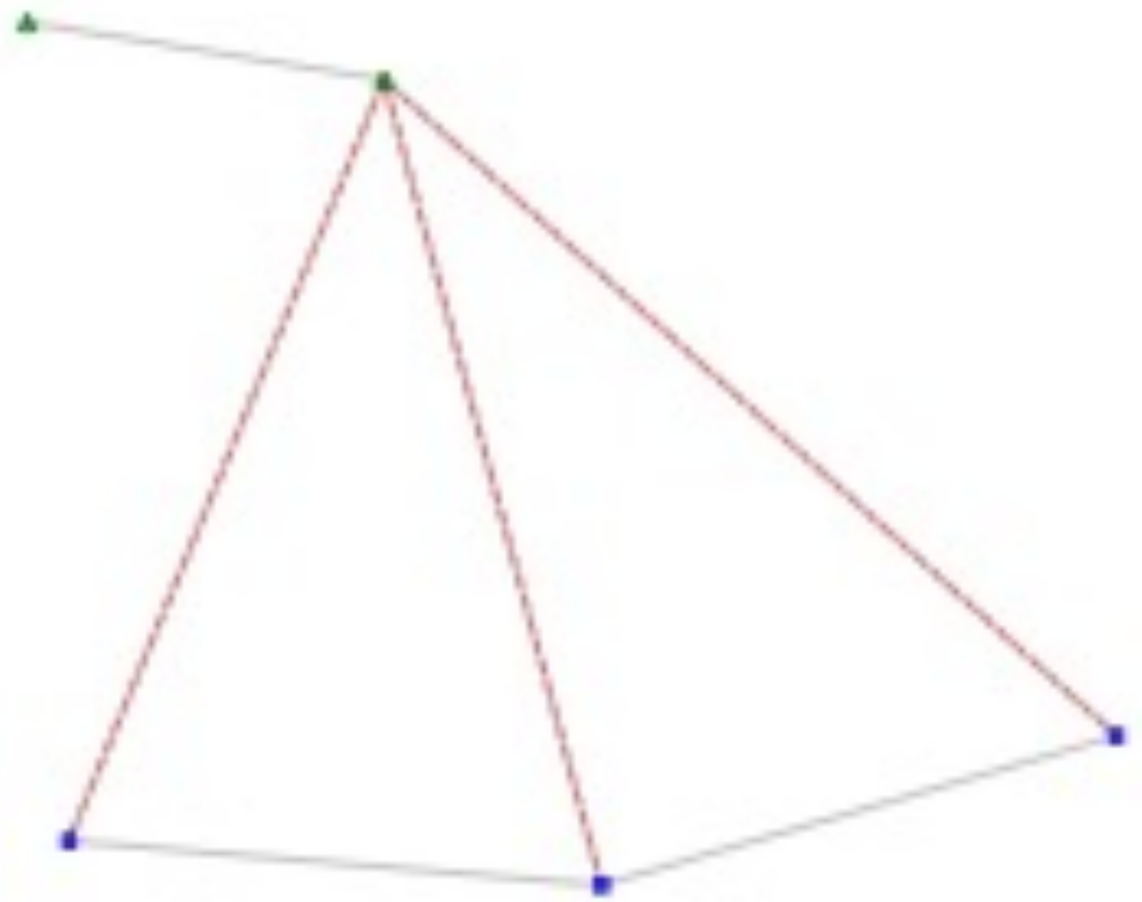
Suboptimal Search

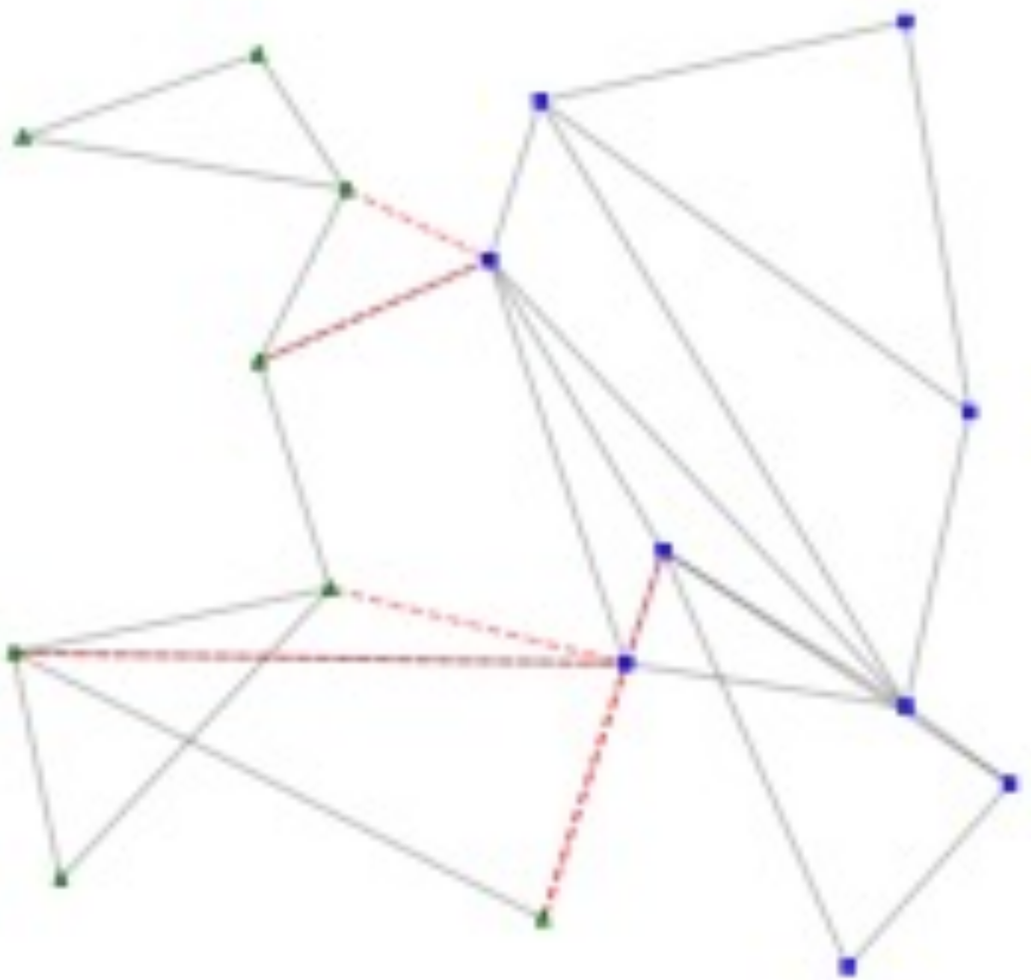
- Suboptimal search using multiresolution graph cuts
- Repeatedly coarsen graph by combing sets of adjacent vertices
- Then find edge cut on small graph
- Then repeatedly uncoarsen and refine edge separator

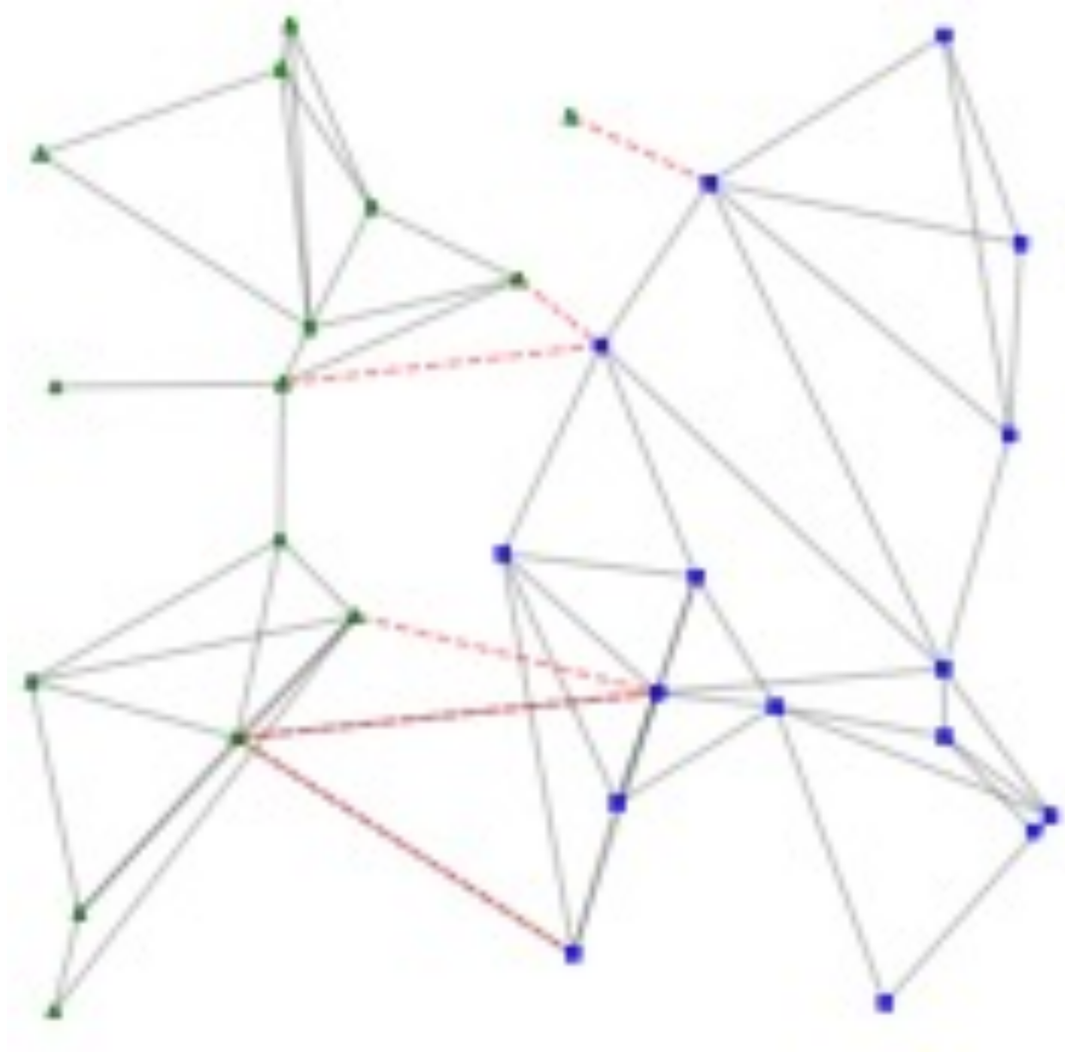
- Finally, use linear program to find minimum jammer placement to jam all links of edge separator

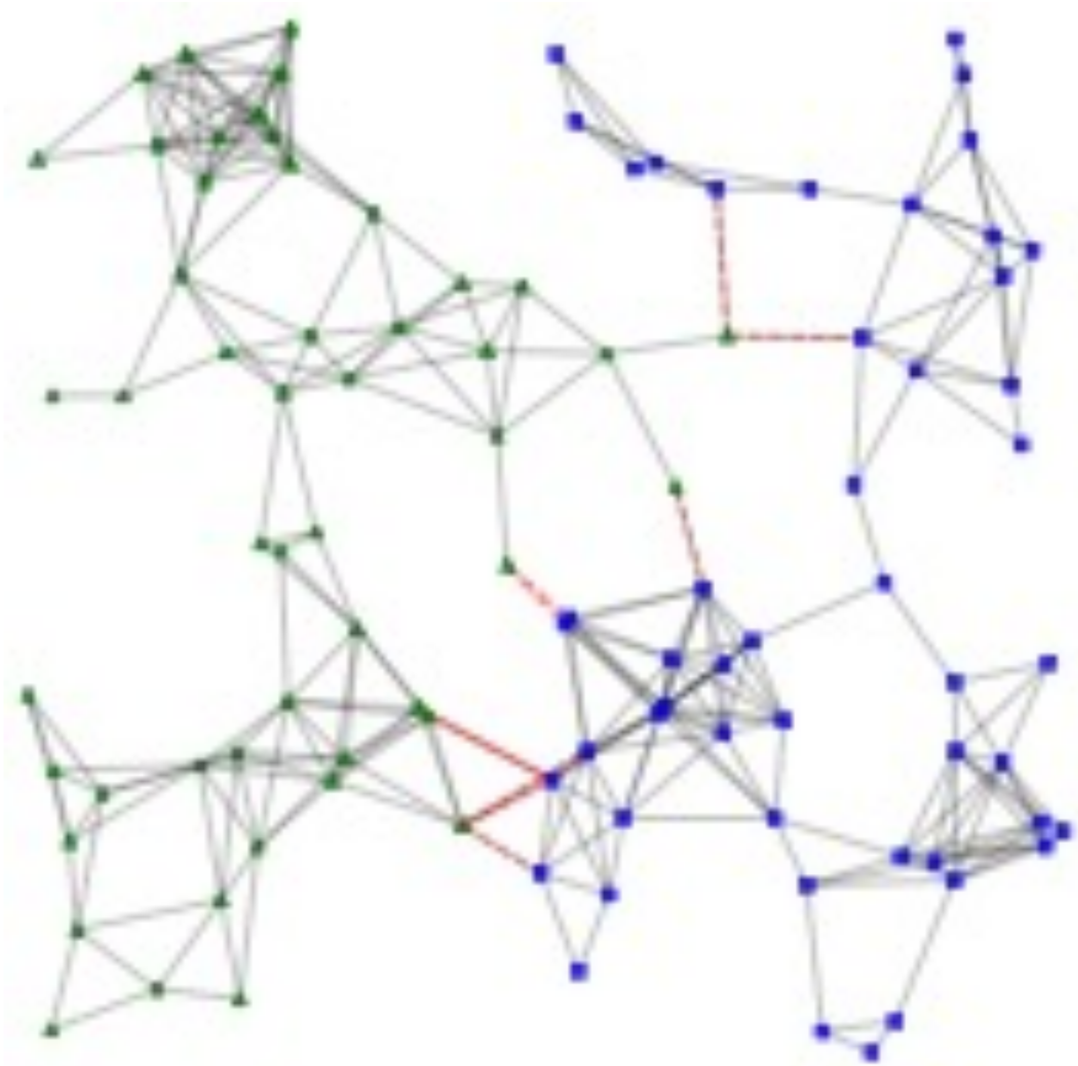












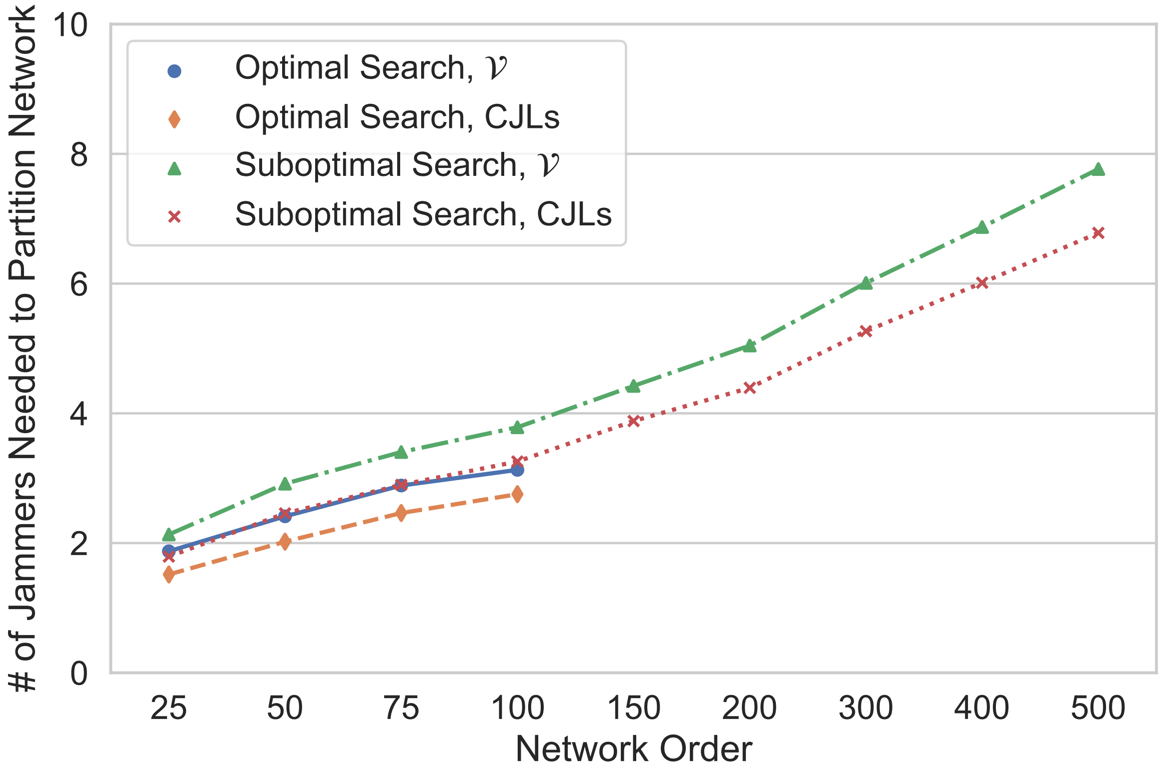


Table: Optimal formulation ILP time complexity and CJL overhead

Order	ILP w/out CJL(s)	Search CJL(s)	Ratio
25	3.619	0.434	11.98%
50	21.864	5.540	25.34%
75	103.573	15.571	15.03%
100	530.792	29.977	5.65%

Time to find suboptimal solution is < 10 ms for networks up to 500 nodes

Enhancing Network Robustness

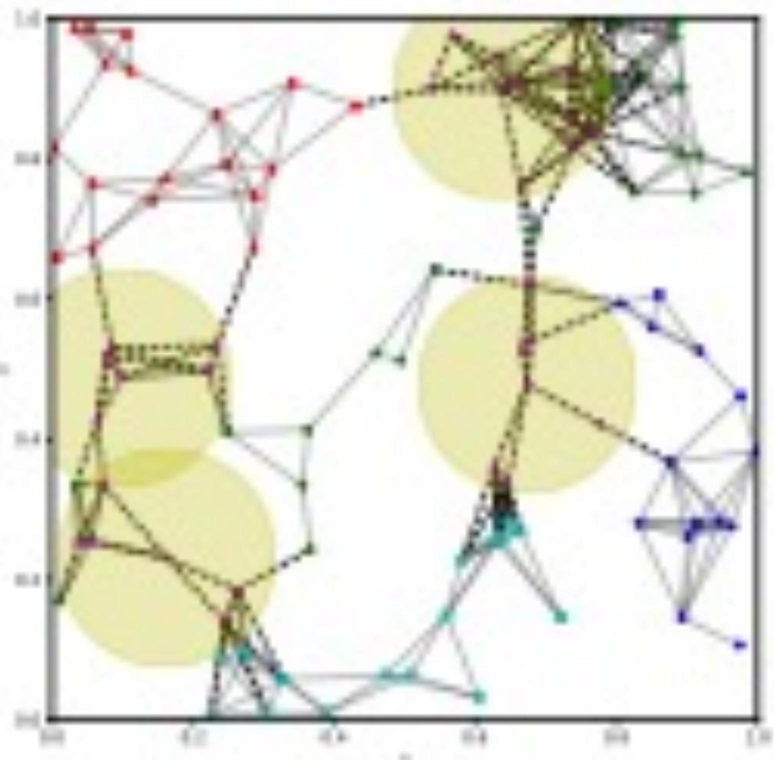
- Can use proactive or reactive measures to avoid disruption for jamming
  - **Proactive:** we increase the network robustness by adjusting the network topology or adding some additional communicators to make the network harder to partition
  - **Reactive:** we adjust the network topology by repositioning nodes or adjusting communication parameters based on jammed links

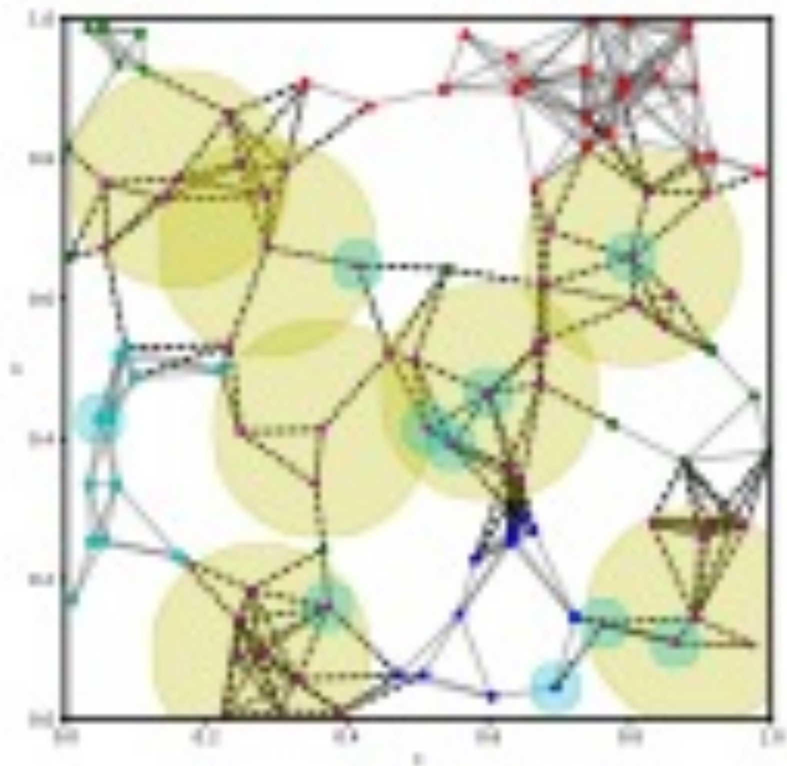
Let:

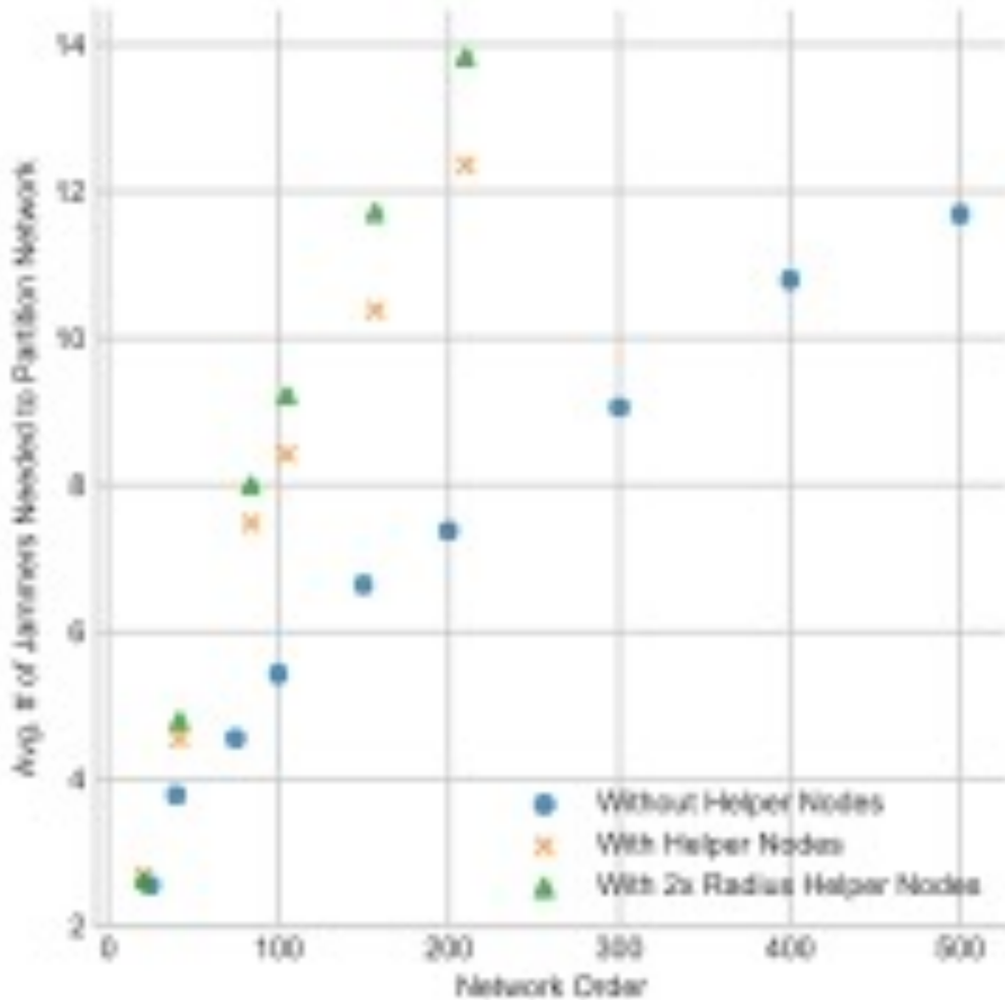
- $\mathcal{G}(\mathcal{V}, \mathcal{E})$ be the network that is to be made more robust
- $\mathcal{P} = \{P_1, P_2, \ldots, P_{N_h} \mid \text{loc}(P_i) \in \mathbb{R}^2,\ i = 1, 2, \ldots, N_h\}$ are the locations of N_h helpers
- $\mathcal{H}(\mathcal{V}', \mathcal{E}', \mathcal{G}, \mathcal{P})$ be the reinforced network induced by placing helper nodes with
  $\mathcal{V}' = \mathcal{V} \cup \mathcal{P}$
  $\mathcal{E}' = \{(u, v) \in \mathcal{V}' \times \mathcal{V}' \mid \text{dist}(u, v) \leq R_c\}$

Then the helper node placement problem is:

$$\hat{\mathcal{P}} = \arg \max_{\mathcal{P} \in (\mathbb{R}^2)^{N_h}} \eta(\mathcal{H}(\mathcal{V}', \mathcal{E}', \mathcal{G}, \mathcal{P}))$$






Avg. # of Jammers Needed to Partition Network
Network Order
Without Helper Nodes
With Helper Nodes
With 2x Radius Helper Nodes

Future Work

- Found optimal solutions for fixed networks, simple channels
- Extend to:
  - Mobile radios and jammers (time-varying networks)
  - Fading channels (probabilistic links)
  - Uncertainty in channel knowledge
  - Looser definitions of connectivity: more next
  - Adapt topology using communication parameters: adjust transmission power, modulation, coding
  - Adapt topology to achieve other objectives: maximize throughput, minimize latency, prioritize links or flows
  - Adapt placement of resources: caches, databases, computation centers