Characterizing Tradeoffs in Detection and Localization of Network Performance Faults

Leilani Gilpin
University of California San Diego
Swara Kopparty
Harvard University

Under the guidance of Dr. Aaron Jaggard, Dr. Vijay Ramachandran, and Dr. Rebecca Wright

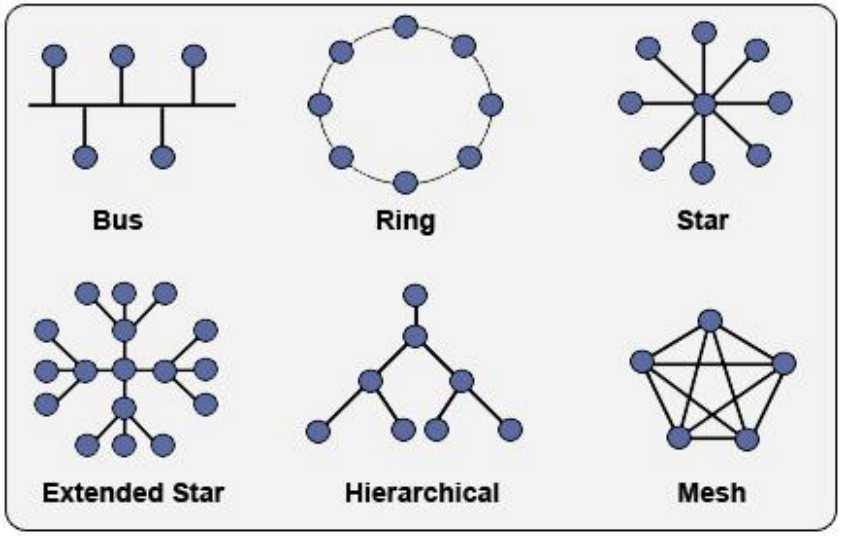
Introduction

- Network operators need to make sure that they satisfy their Service Level Agreements
  - Networks are operating smoothly, no faults
- Standard approach is to monitor network behavior for any faults in performance
- How do we monitor the network behavior?
- Probe-based measures
- Two phases: Detection and Localization

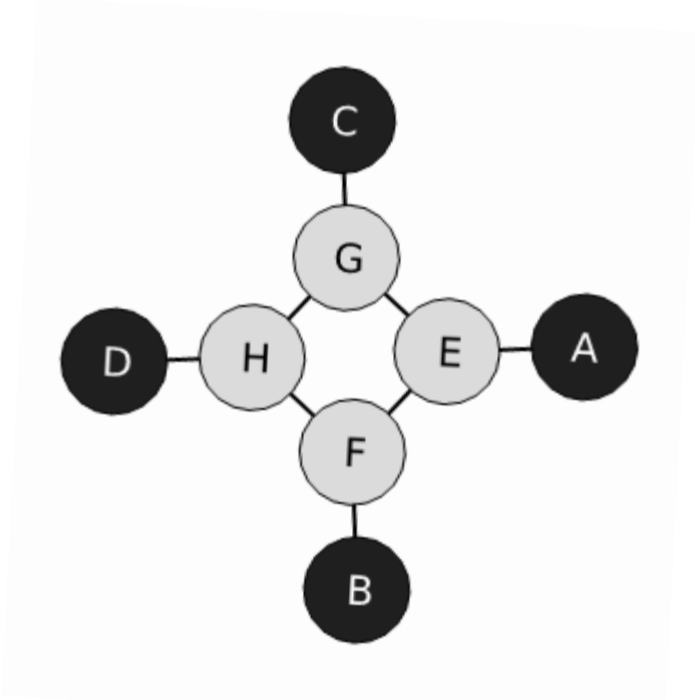
Tradeoffs

- In doing the measurement, there are bound to be tradeoffs that need to be characterized
  - Number of probes to place
  - Where to place the probes
  - Number of messages to send from computer to computer
  - Runtime of the actual algorithm
  - Characterizing necessary overhead vs. optimal overhead

Examples of Network Graphs



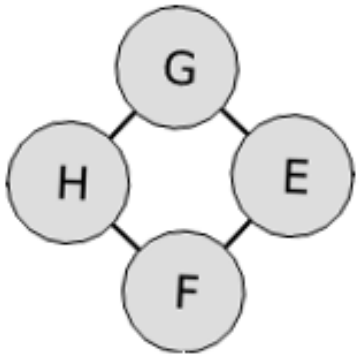
Examples of network graphs



A simple network topology

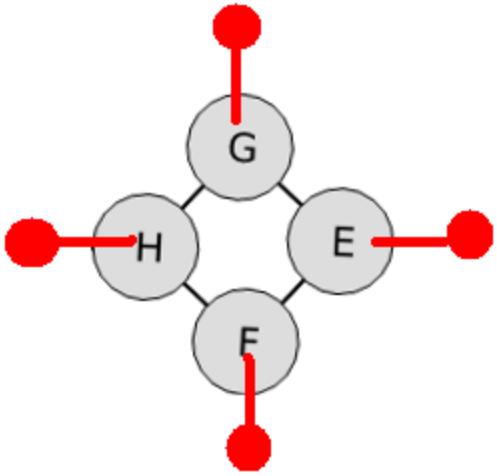
- Each node represents a computer in the network
- We look at directed graphs, because direction and quantity of traffic flow is also important

Detection



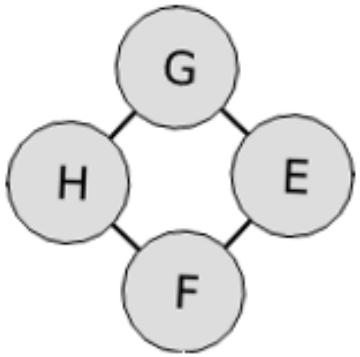
- Need some way of detecting that there is a fault in performance
- Place probing beacons at strategic places in the network
  - Placed at measurement nodes
- Example: Four node ring network

Detection – Trivial Strategy



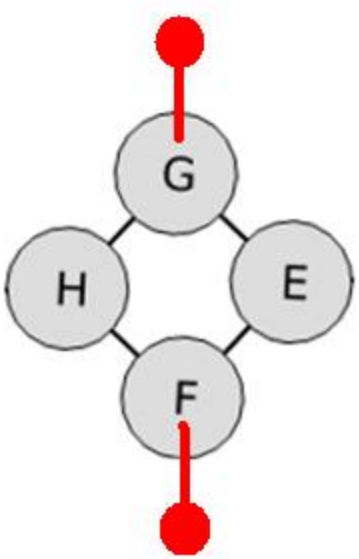
- Every node has a probing beacon
- Follow every path once
  - Pro: covers all cases
  - All faults will be detected, and immediately
  - Con: Generates lot of computational and network overhead

Detection – Other Extreme ☺



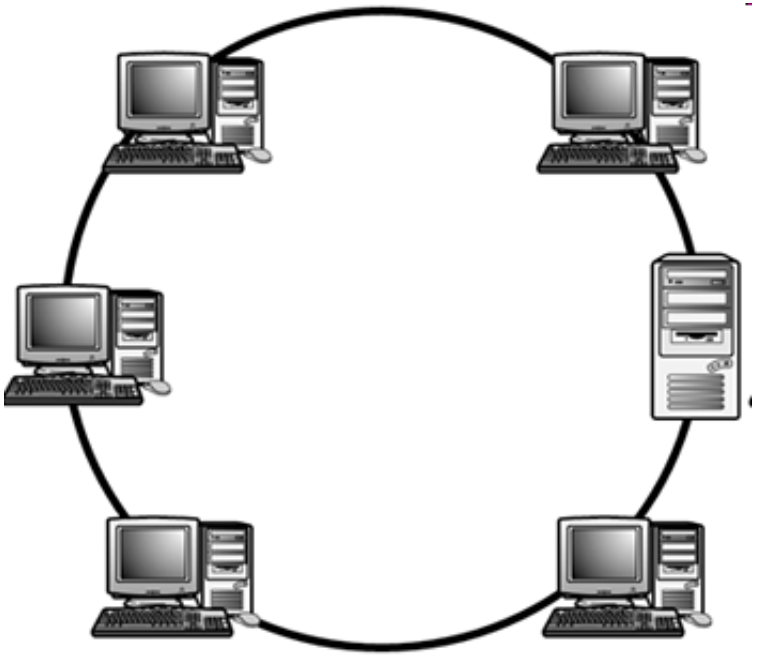
- No probing beacons
  - Pros: no overhead at all!
  - Cons: we forgot to solve the problem

Detection – Less Extreme Strategy



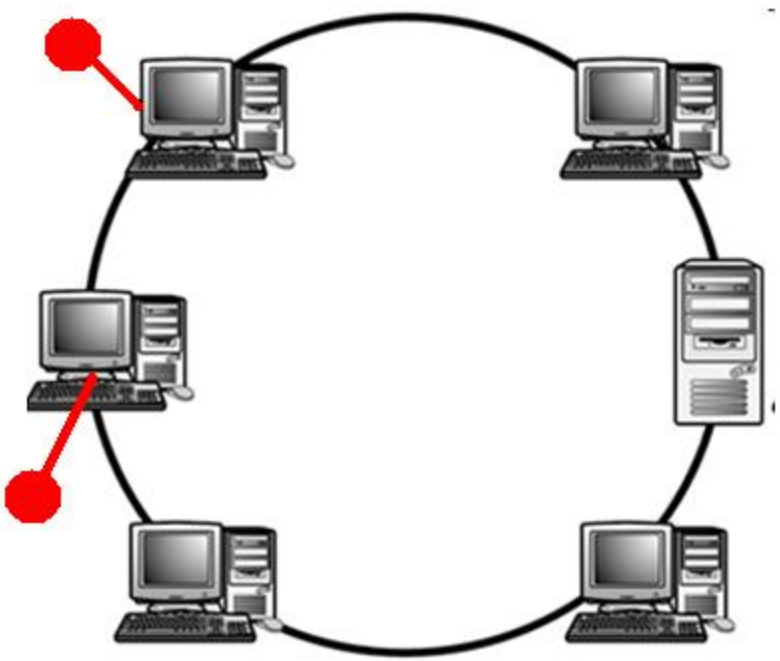
- Two nodes have probing beacons
  - Pro: Much less network and computational overhead
  - Con: May not check every node for faults, will not be able to find the problem immediately

Localization



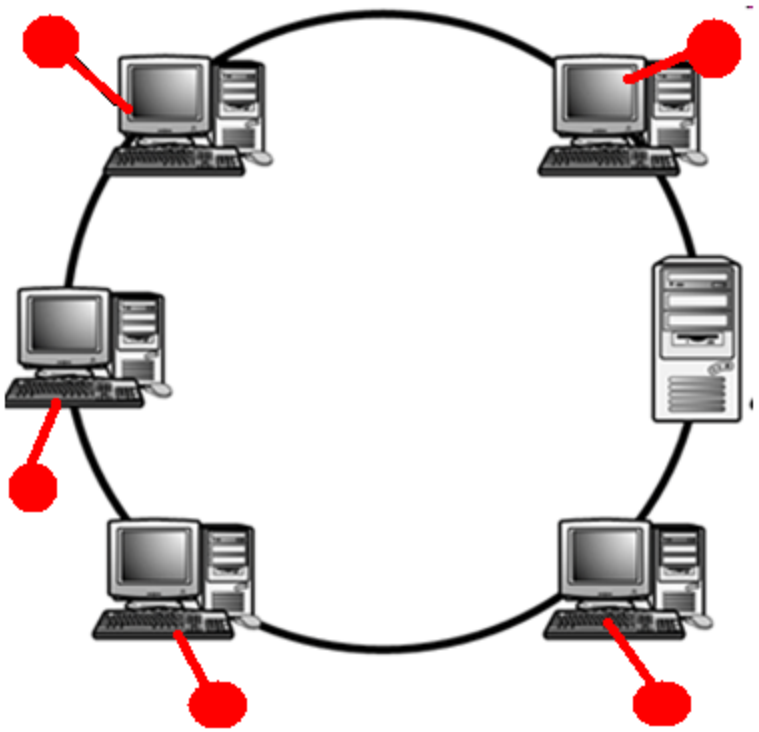
- Once we know there is an anomaly, we need to find which node is causing the trouble
- Example: 5-node ring network

Localization – Example Strategy



- Two probing beacons
  - Pro: Detection will occur with little network overhead
  - Con: Cannot localize the anomalous node

Localization – Another Example



- Multiple probing beacons (e.g., 5)
  - Pro: Can detect and localize the anomalous node immediately
  - Con: Again – more overhead

Problem

- Catch network faults
  - Bound to be tradeoffs in minimizing computational overhead, network overhead, and speed of catching faults
- Given a network topology
  - Where should we place probing beacons
  - Which paths should we probe
- Give an algorithm that does this for a family of topologies

Plan of Action

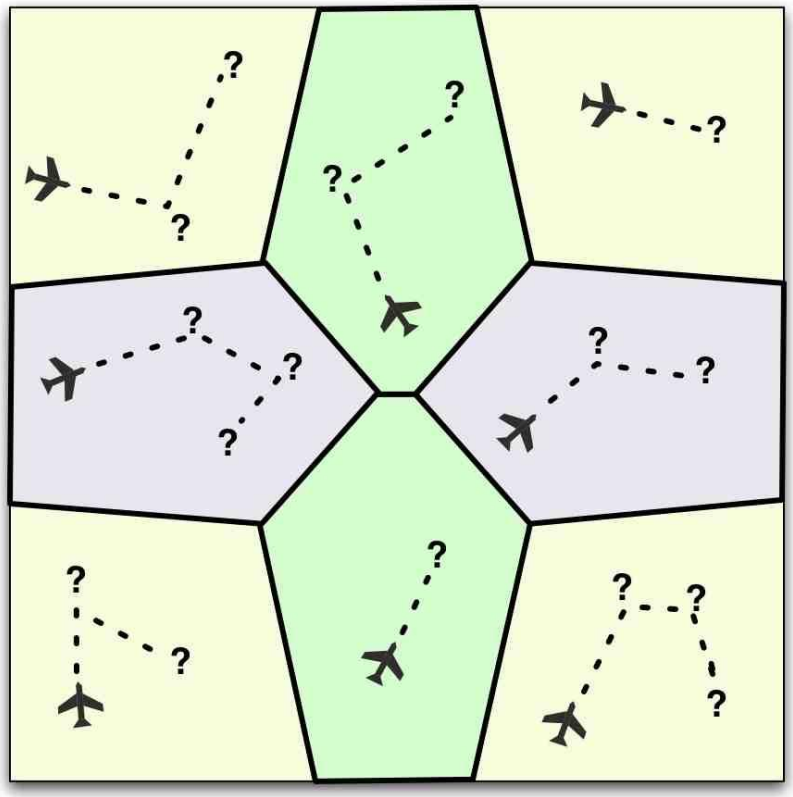
- Look at more general strategies
- Perhaps try to characterize algorithms for families of graphs without worrying about tradeoffs
  - Cycle graphs, path graphs
- Ultimately: Look for generalizations and adaptations to incorporate minimizing the tradeoffs for families of network topologies

First steps

- Fundamental tradeoffs for classes of algorithms
- Listing desirable properties and formalize them
  - How can we tackle these properties in isolation
- Create a network modeling to test them
- Can we create a code a better algorithm?

Different Directions

- Background in robotic networks
- Applying aspects of control theory and communication complexity
- Can distributed networks and robotic controls apply to this problem?



Theoretical Applications

- Classes or families of algorithms
- Looking at networks as metrics
  - Coverage vs. time
  - Set cover approaches
  - NP Hard problems
- Modeling and average case complexity

Thank you for listening!

References

1. P. Barford, N. Duffield, A. Ron, J. Sommers, "Network Performance Anomaly Detection and Localization." INFOCOM 2008
2. P. Barford, J. Kline, D. Plonka, and A. Ron, “A Signal Analysisof Network Traffic Anomalies,” in *Proceedings of ACM SIGCOMM Internet Measurement Workshop*, November 2002.
3. Y. Bejerano and R. Rastogi, “Robust Monitoring of Link Delays and Faults in IP Networks,” in *Proceedings of IEEE INFOCOM’03*, April 2003.