Inferring and Mitigating a Link's Hindering Transmissions in Managed 802.11 Wireless Networks

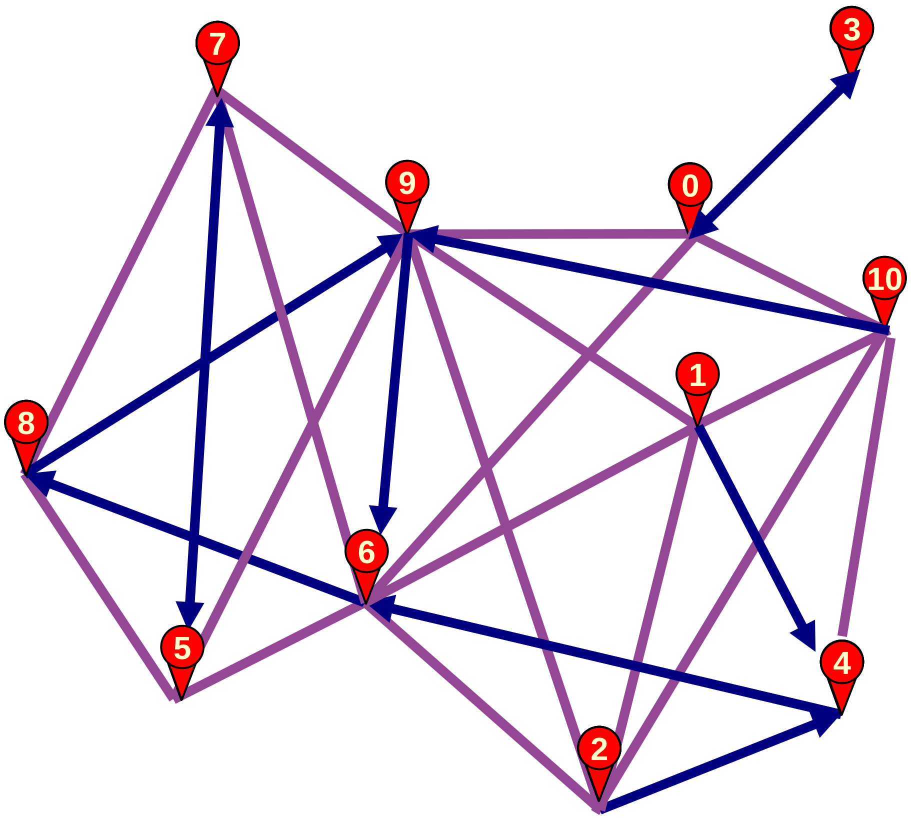
Eugenio Magistretti, Omer Gurewitz*,
and Edward Knightly

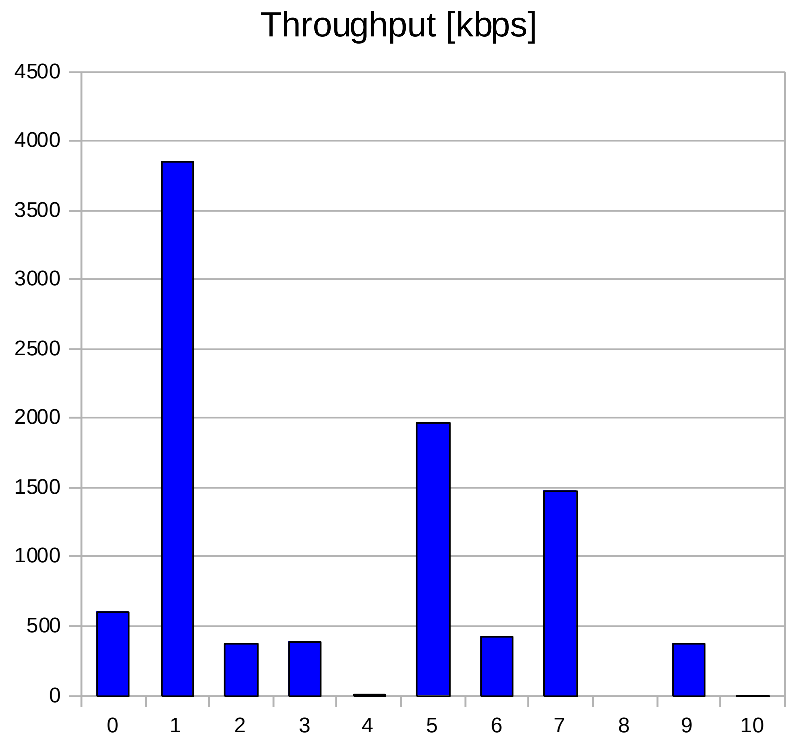
Rice University
*Ben Gurion University

http://networks.rice.edu

The Management Problem

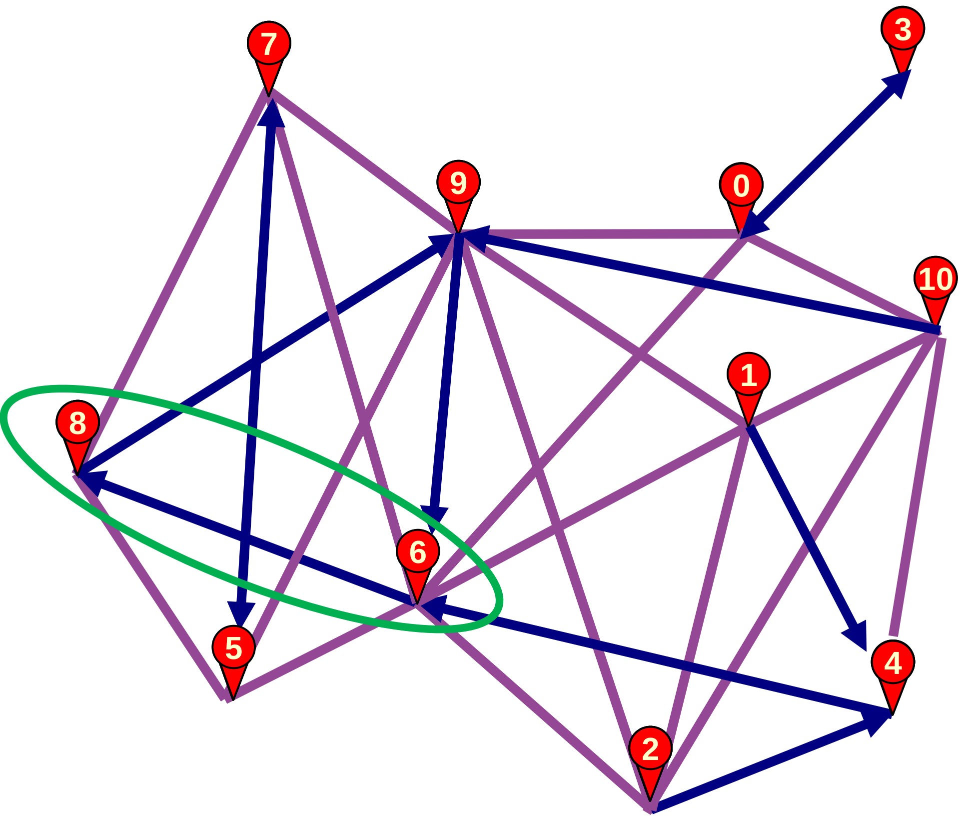
Given a dense wireless network (e.g., WLAN, Wireless Mesh Network…)

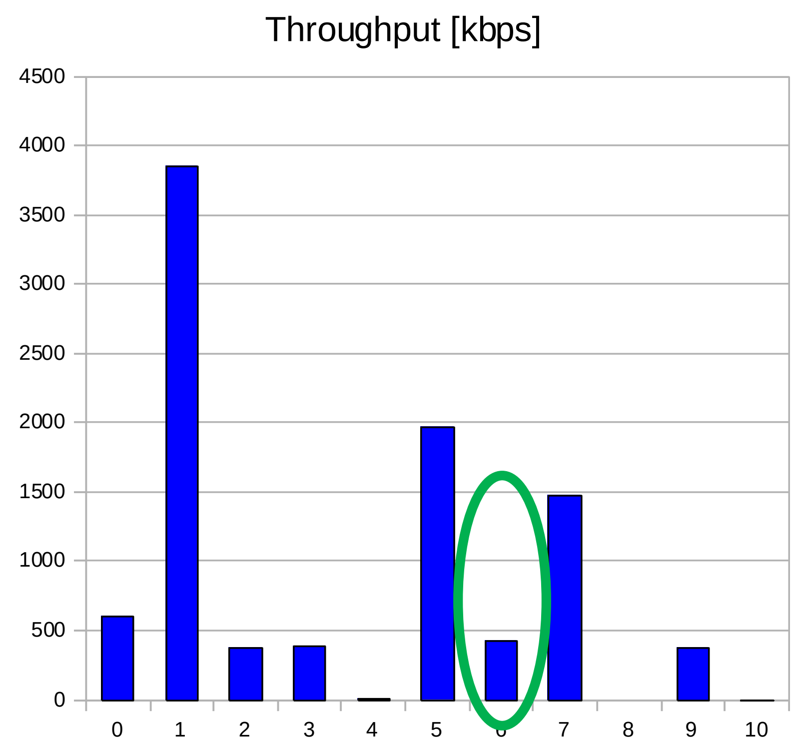




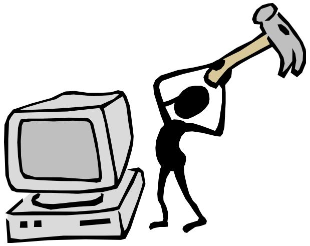
The Management Problem

Given a dense wireless network (e.g., WLAN, Wireless Mesh Network…)





- The throughput of the specific flow is lower than the manager expects
  - Why? How to fix it?

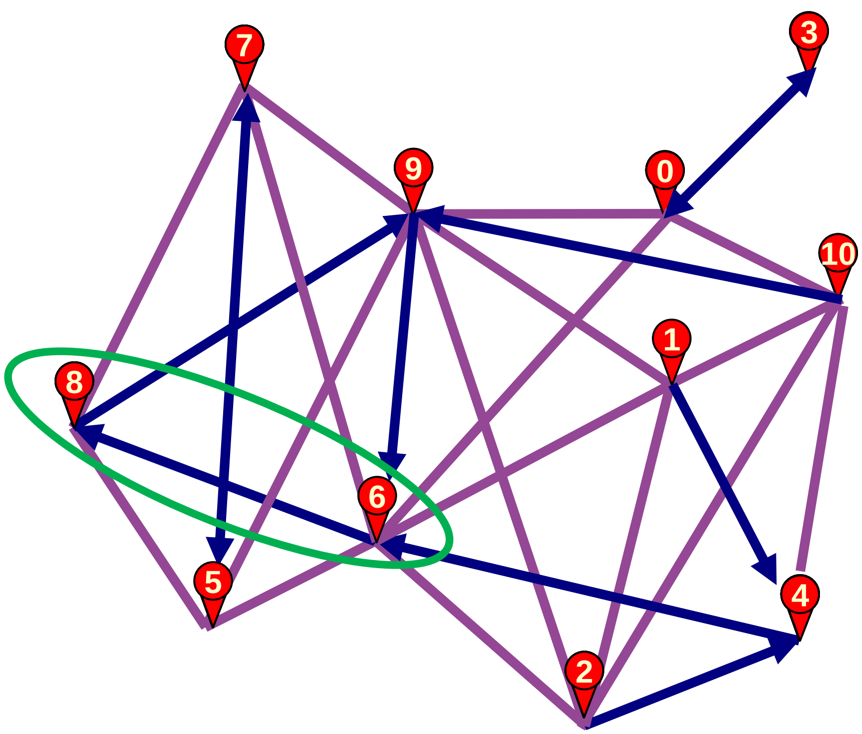


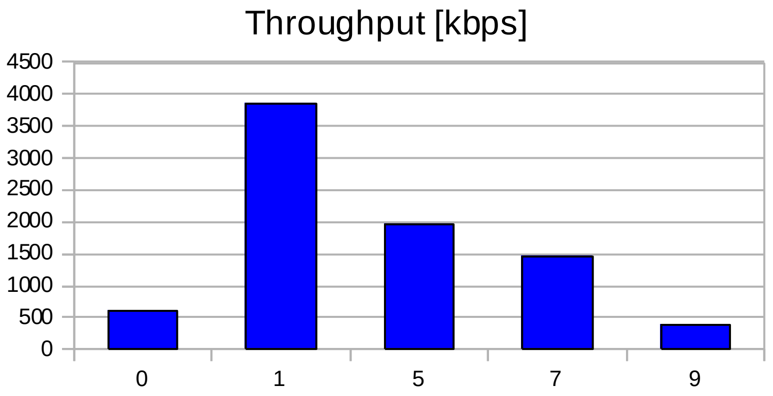
Objective

- Improve the throughput of a specific flow using a small set of passively collected, time-aggregate local channel measurements reported by the nodes.
  - Determine which flow should be throttled / moved to another channel
  - Predict the throughput gain

The Management Problem

- Does it really matter which link we throttle?
  - Example: Given a topology and the flow throughput...

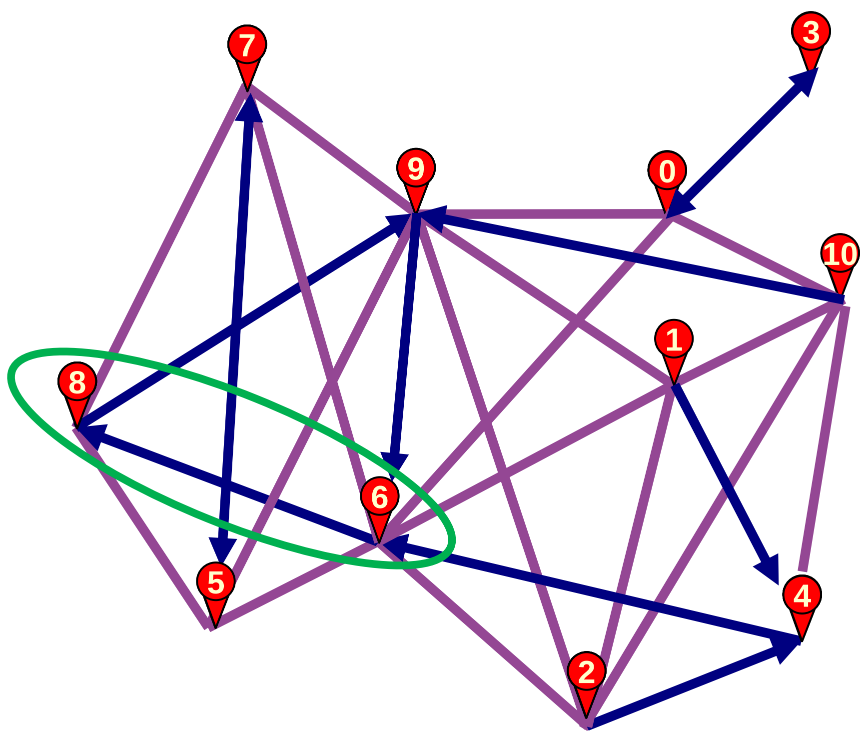


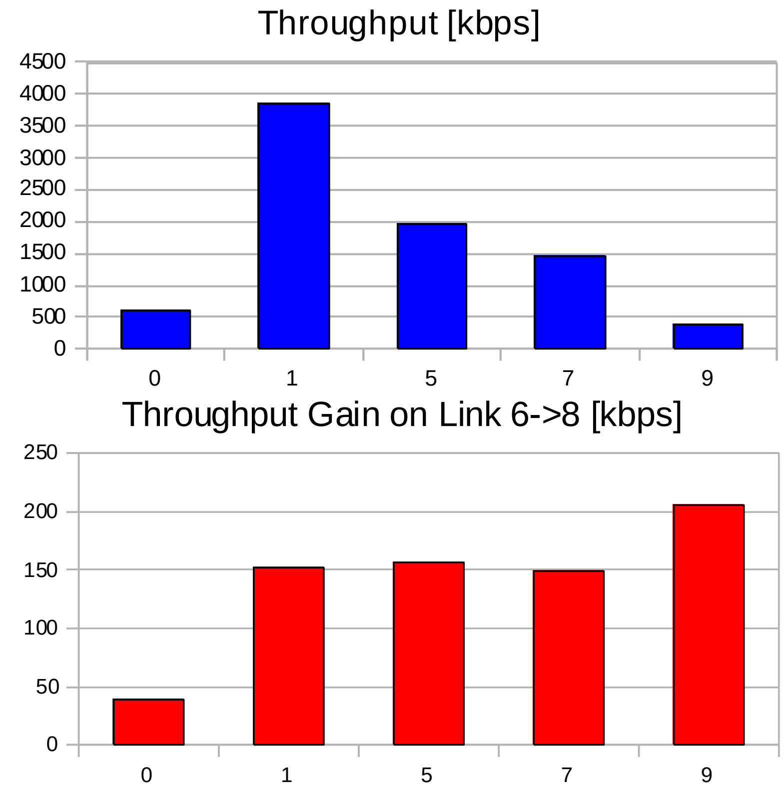


The Management Problem

- Does it really matter which link we throttle?
  - Example: Given a topology and the flow throughput...

  Limit the transmission rate of different “neighbor” links for 400 kbps

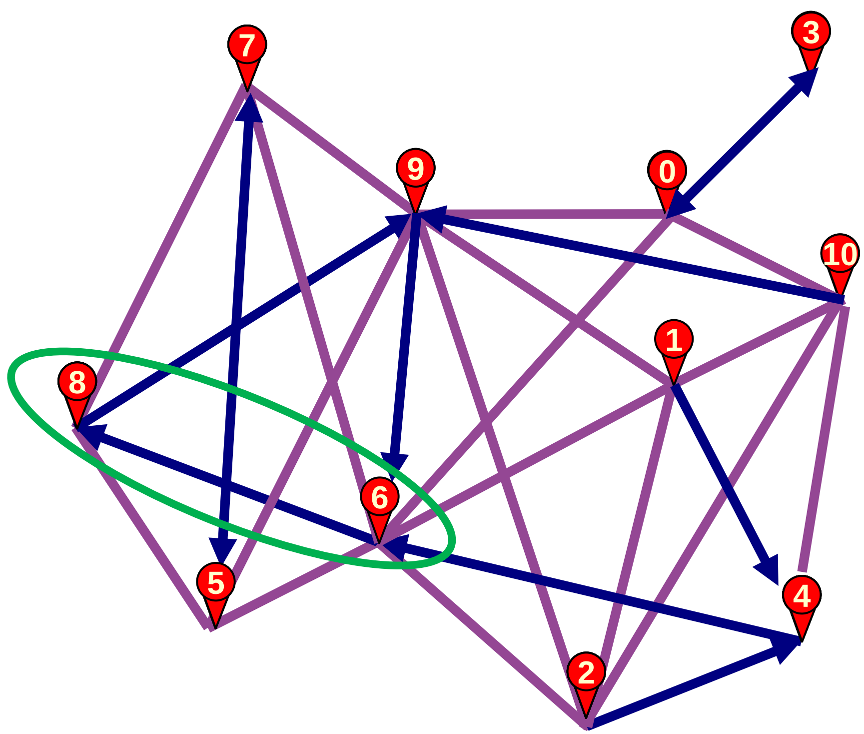


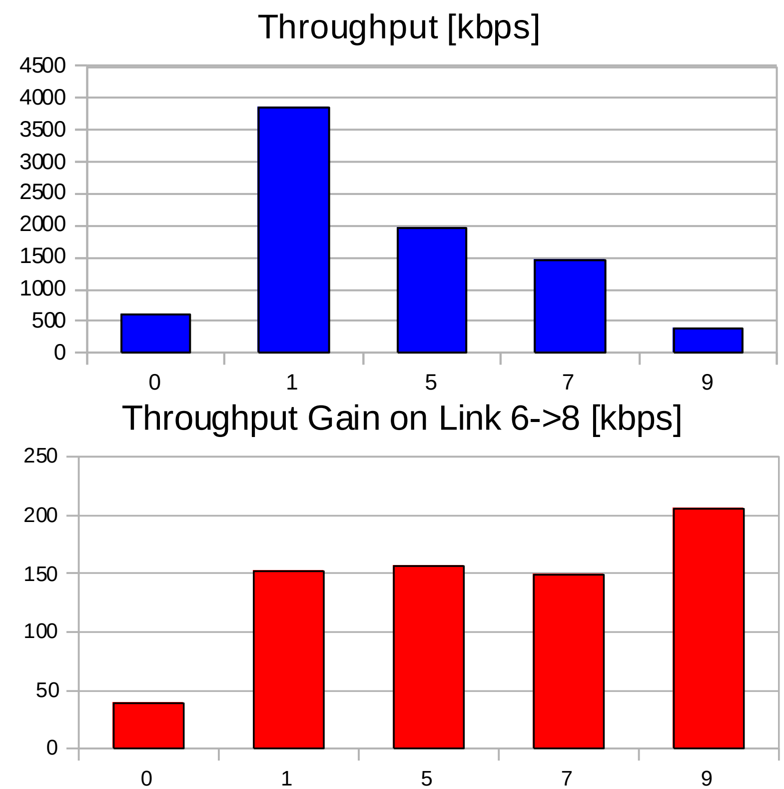


The Management Problem

- Does it really matter which link we throttle?
  - Example: Given a topology and the flow throughput...

    Limit the transmission rate of different “neighbor” links for 400 kbps



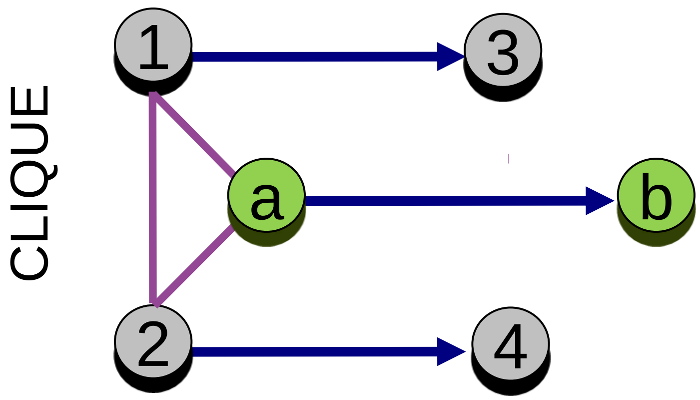


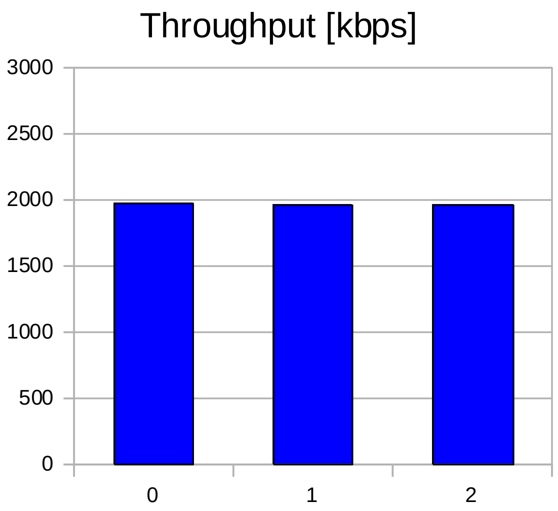
- In our experiments, a flow can gain from **7%** to **172%** of the rate-limited quantity

The Management Problem

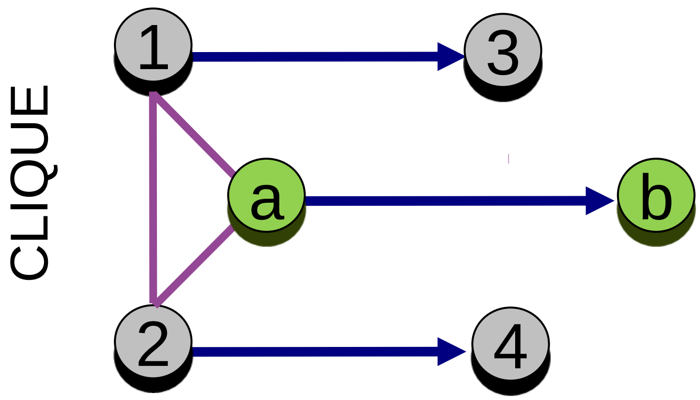
Throttling different flows produces different throughput gains

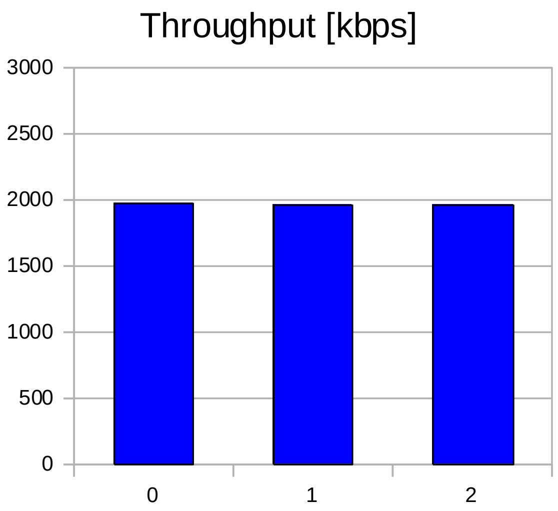
Coordination

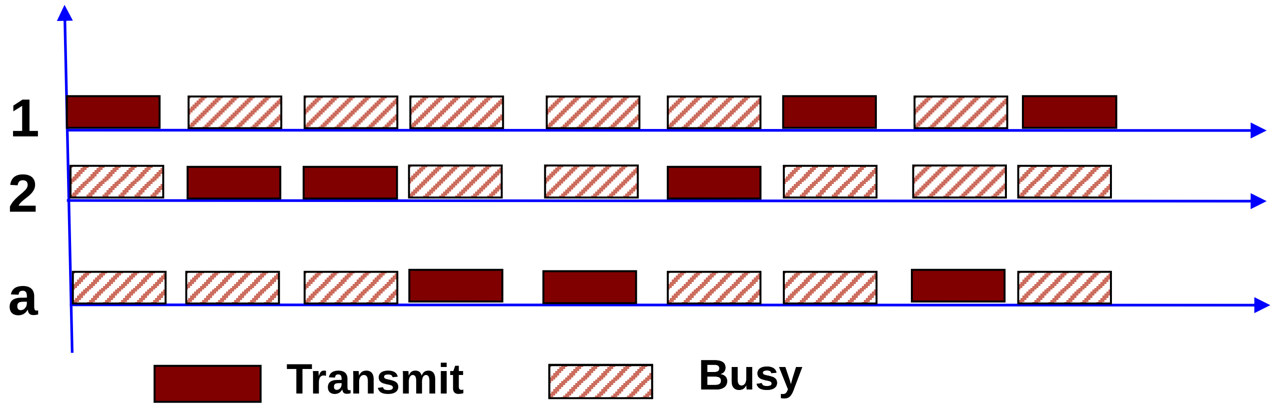




Coordination

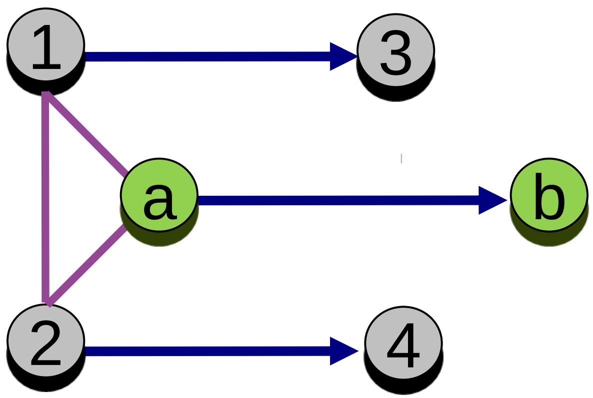


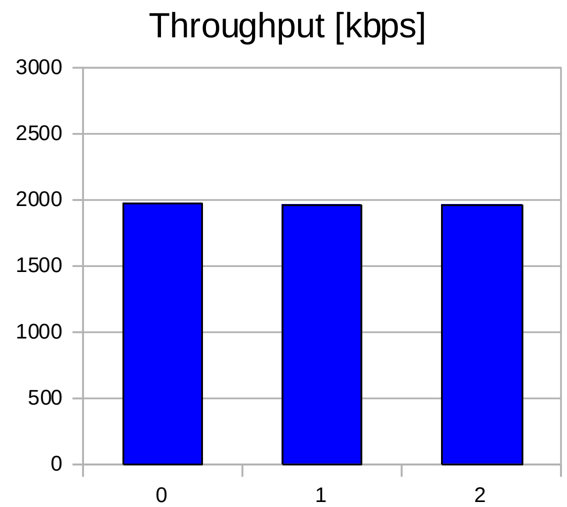




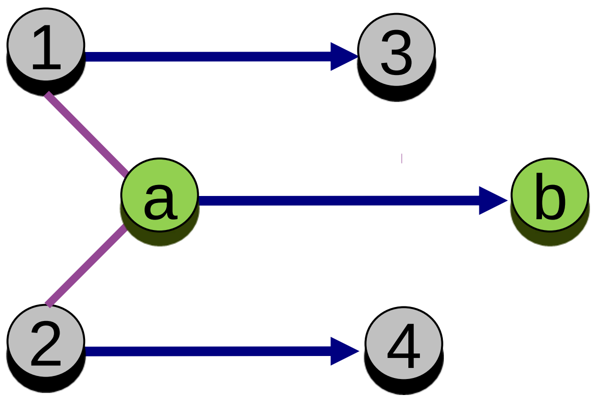
Coordination

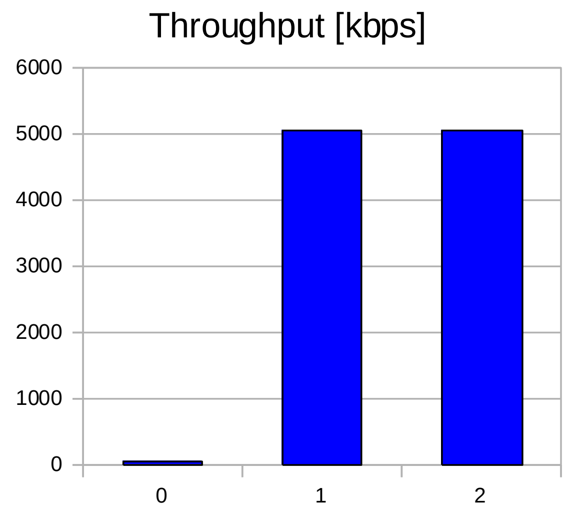
CLIQUE



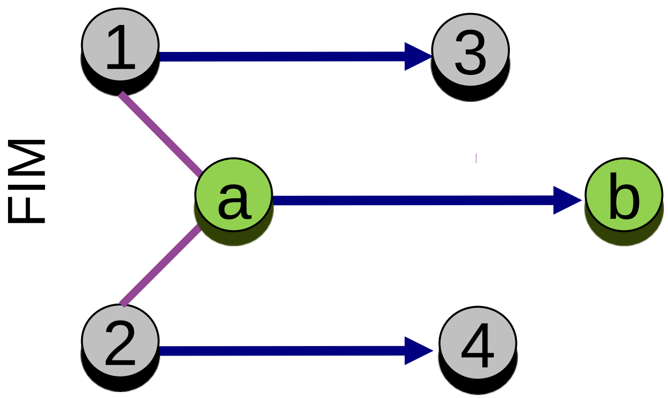


FIM

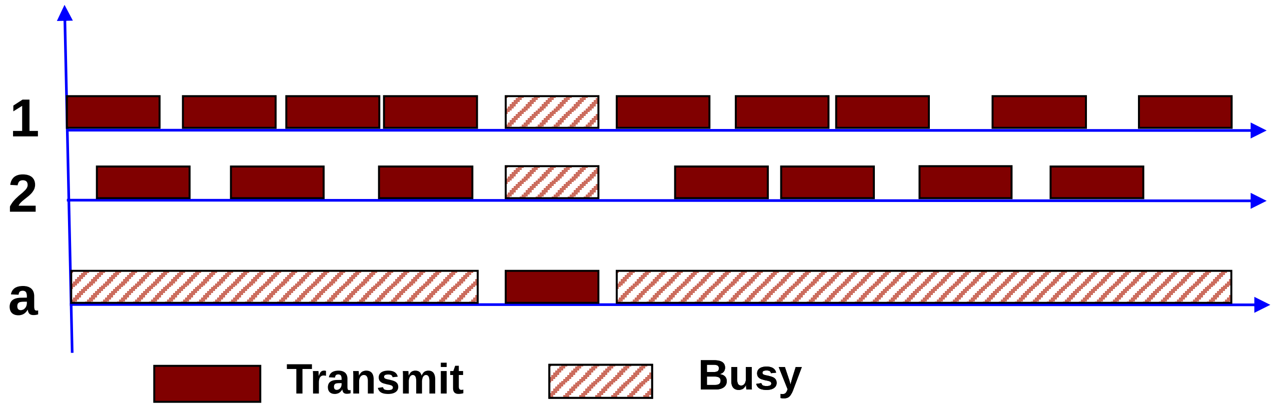




Coordination

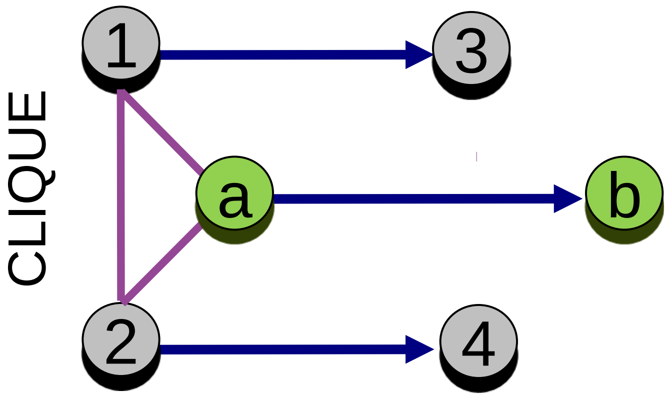


- Flow in the Middle
  - Transmissions of flows 1 and 2 are not **coordinated**
  - Flow a senses the medium busy most of the time

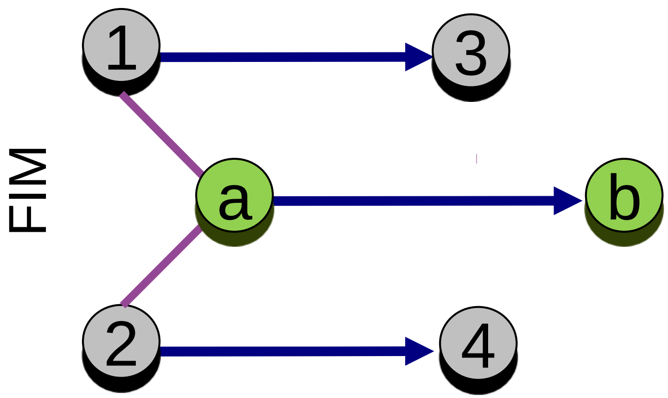


Coordination

- Trying to improve flow a, do the different throughputs/ topology affect the gain?
- Example: rate-limit flow 1 for 400 kbps

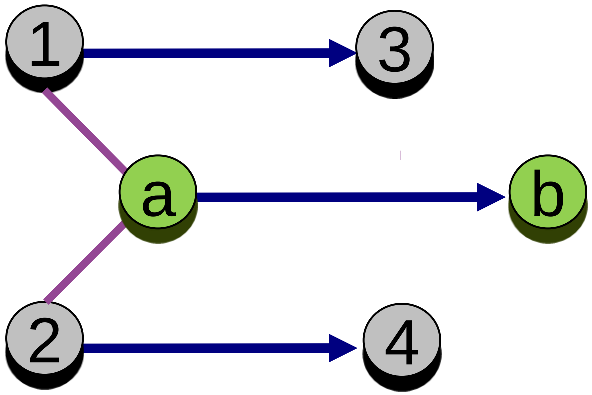


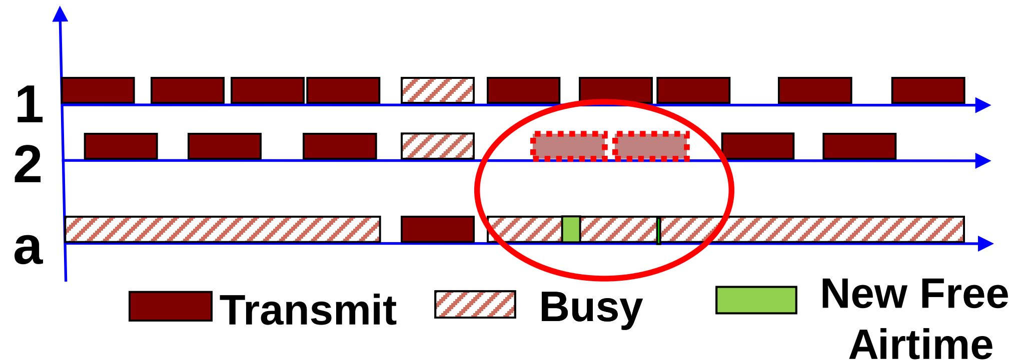
- Flow a gains about **300 kbps**



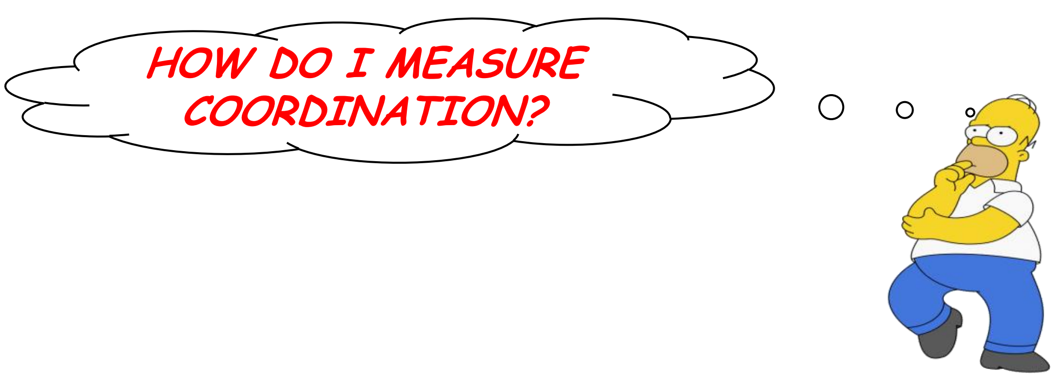
- Flow a gains only **120 kbps**

Coordination



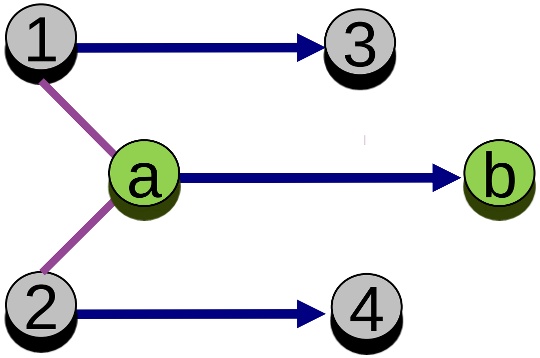


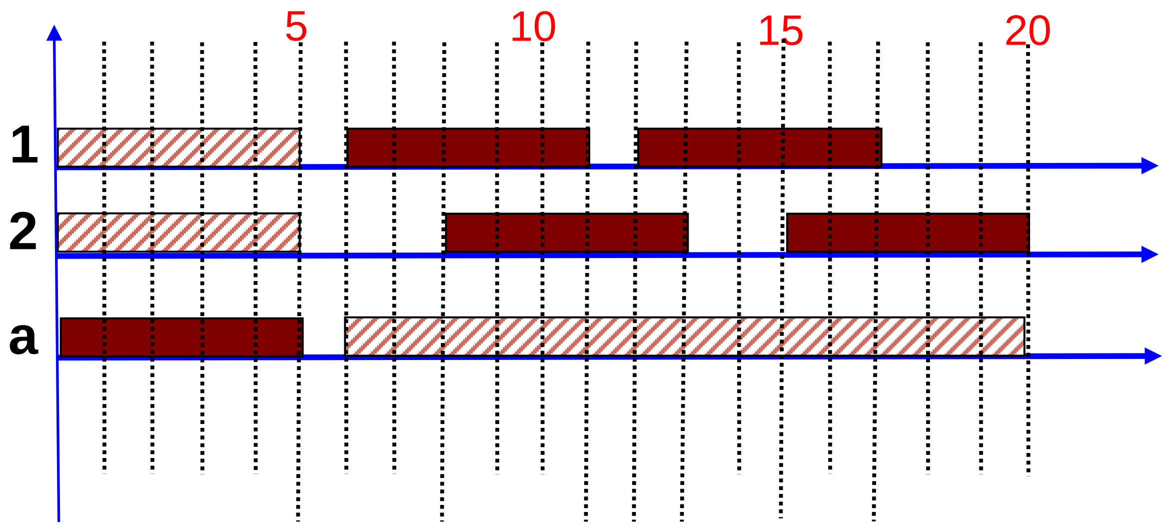
- The throughput gain of flow a depends on the **coordination** between the transmissions of neighbors 1 and 2



Activity Share: Measure of Node Coordination

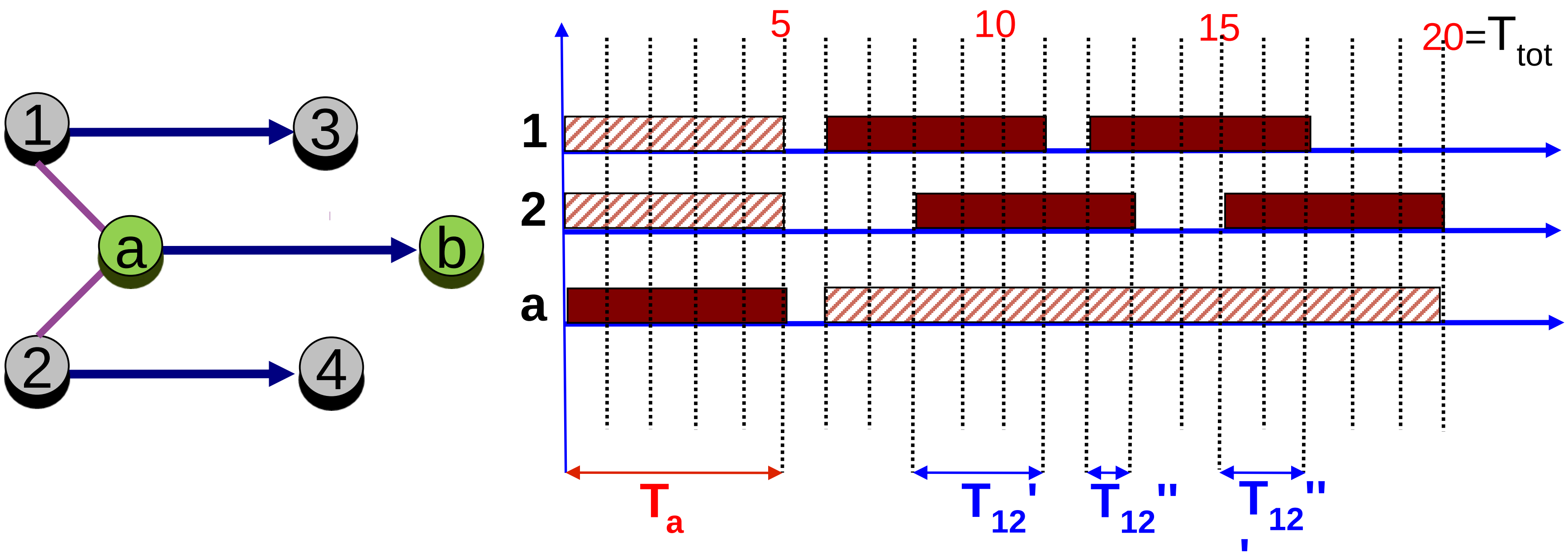
- Activity Share
  - Fraction of time that different sets of nodes spend transmitting simultaneously





Activity Share: Measure of Node Coordination

- Activity Share
  - Fraction of time that different sets of nodes spend transmitting simultaneously



$AS(\{a\}) = |T_a|/T_{tot} = 5/20$

$AS(\{1,2\}) = |T_{12}' \cup T_{12}'' \cup T_{12}'''|/T_{tot} = 6/20$

Activity Share: Measure of Node Coordination

The Activity Share captures the mutual relationship and coordination among nodes

Activity Share Inference

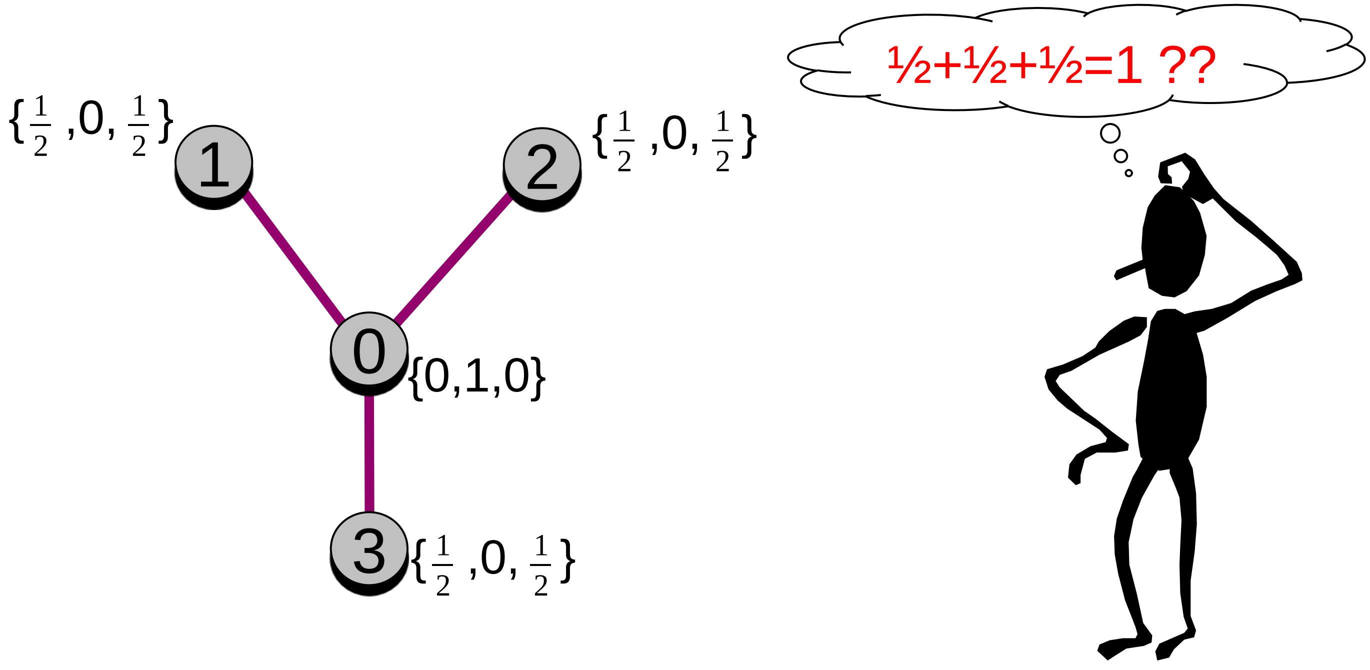
- The Activity Share
  - cannot be locally measured, **hence** nodes need to exchange information
  - can be computed *exactly* by exchanging traces, **but** trace exchange is airtime consuming

accuracy ⟺ overhead

- How to **infer the Activity Share with limited overhead**?

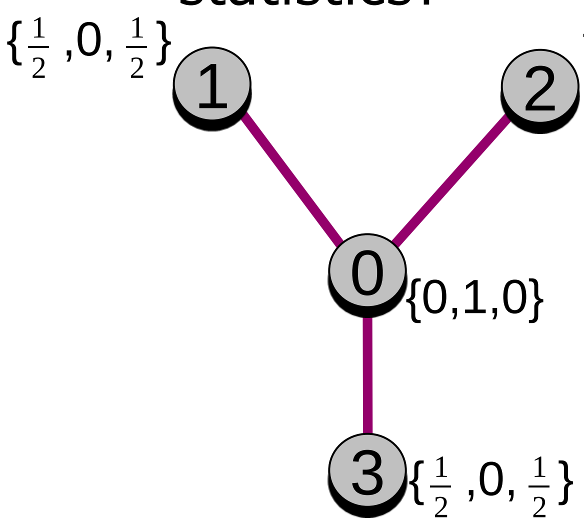
Activity Share Inference (cont'd)

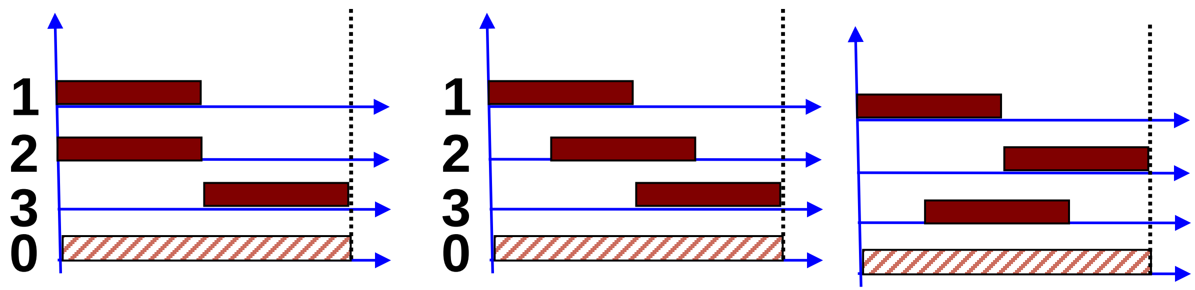
- Each node collects and reports time averages for {transmitting, busy, idle}
- Q. Which Activity Share distributions yield these node statistics?



Activity Share Inference (cont'd)

- Each node collects and reports time averages for {transmitting, busy, idle}
- Q. Which Activity Share distributions yield these node statistics?

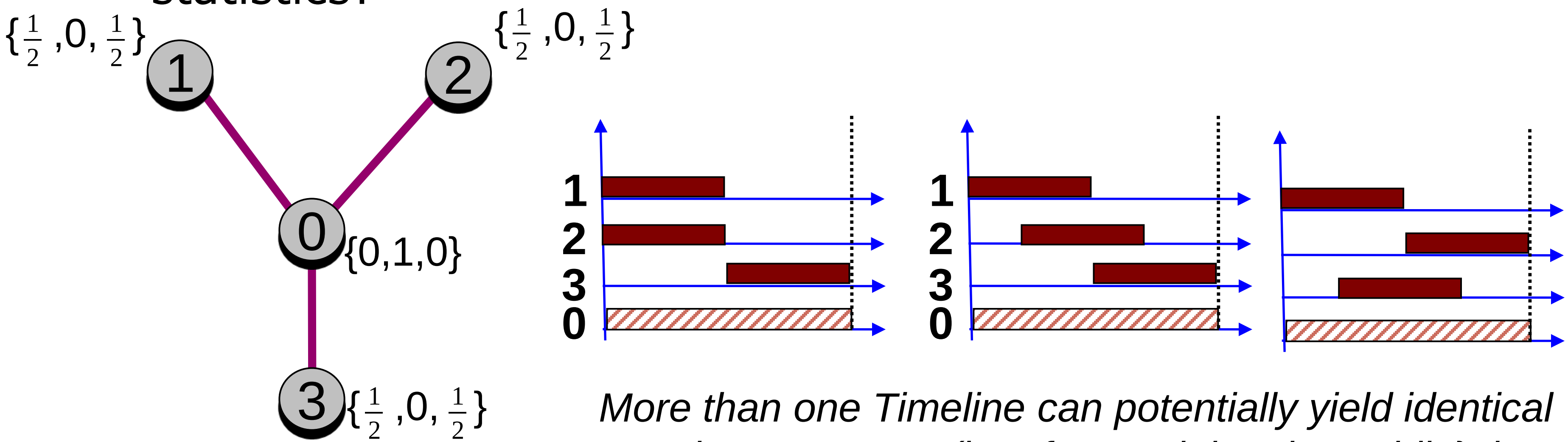




More than one Timeline can potentially yield identical report time averages (i.e., {transmitting, busy, idle} times)

Activity Share Inference (cont'd)

- Each node collects and reports time averages for {transmitting, busy, idle}
- Q. Which Activity Share distributions yield these node statistics?



More than one Timeline can potentially yield identical report time averages (i.e., {transmitting, busy, idle} times)

The reports define a **solution domain** for the Activity Share

Activity Share Inference Secret Sauce

- **Physics**: eliminate distributions that are "impossible"
  - Ex. My busy time coincides with neighbors' transmitting time

- **Protocols**: penalize distributions that defy 802.11 rules
  - Ex. Neighbors transmitting simultaneously violates carrier sense. Should be rare.
- **Unbiased**: minimize relative entropy
  - Find the distribution with the least bias from the prior knowledge

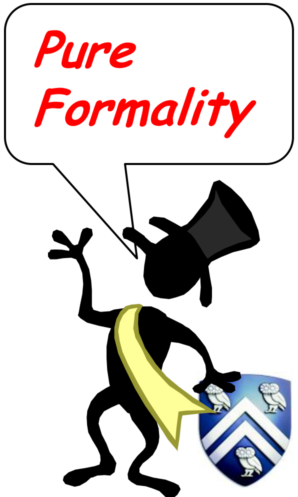
Optimization problem

- Variables: the Activity Share distribution, $\bar{x} = \{x_0, x_1, \ldots, x_\gamma\}$
- Data: time-aggregate measurements reported by the nodes {transmitting, busy, idle} for all nodes
- Objective function :

$$\underset{\bar{x}}{Min}\left[\sum_{j=0}^{\gamma-1} x_j \log \frac{x_j}{\omega_j}\right]$$

** ω is the prior distribution of the network states*

- Constraints: *AS* distribution must satisfy the constraints imposed by all local observations

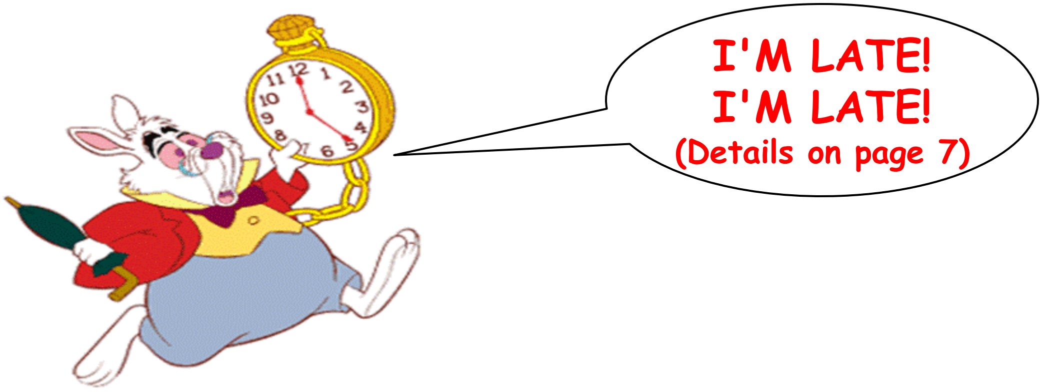


Throughput Prediction

- Given the Activity Share
can we estimate who to throttle?

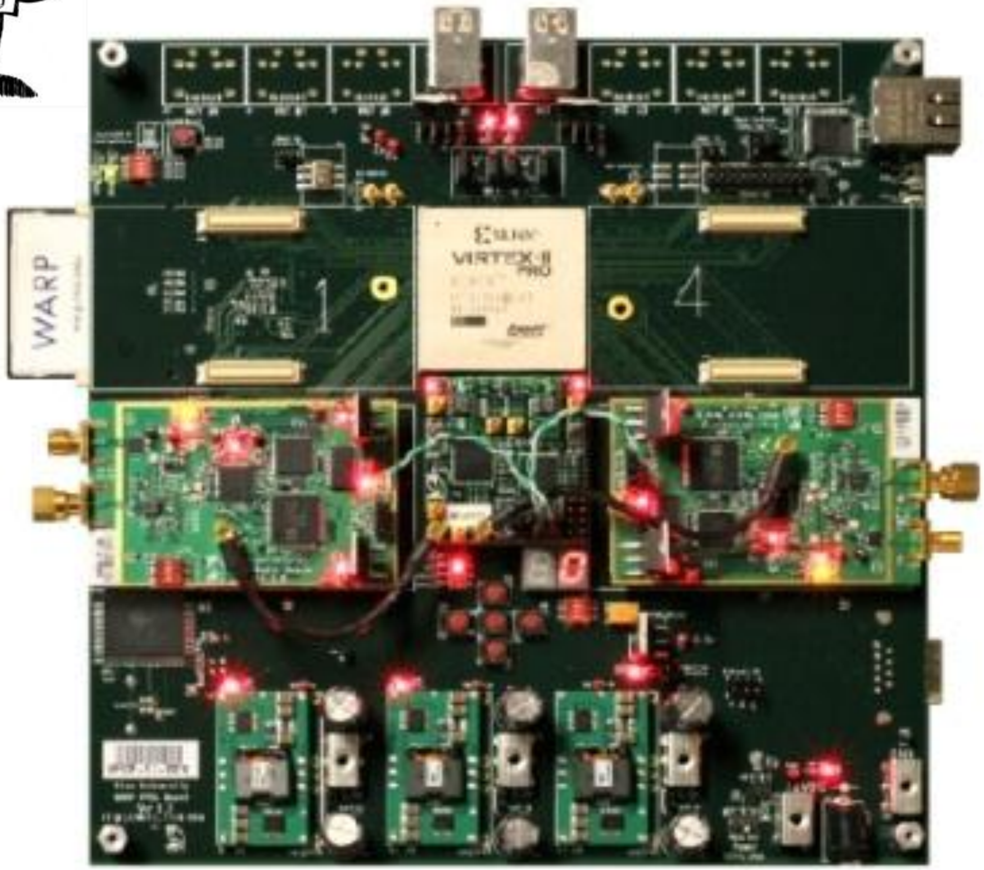
- Predict how alternative rate-limiting actions will benefit the throughput of the target flow

1. Estimate the Activity Share after a rate-limiting action
2. Compute the relationship between throughput and Activity Share



Too Many Equations… Does It Work??

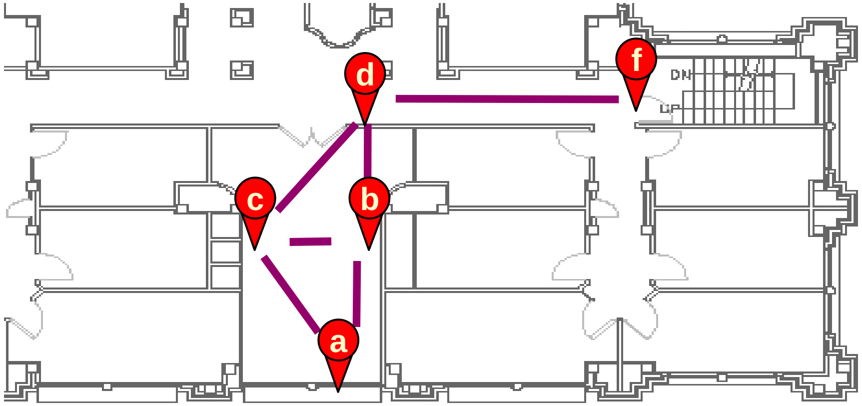
WARP FPGA BOARD

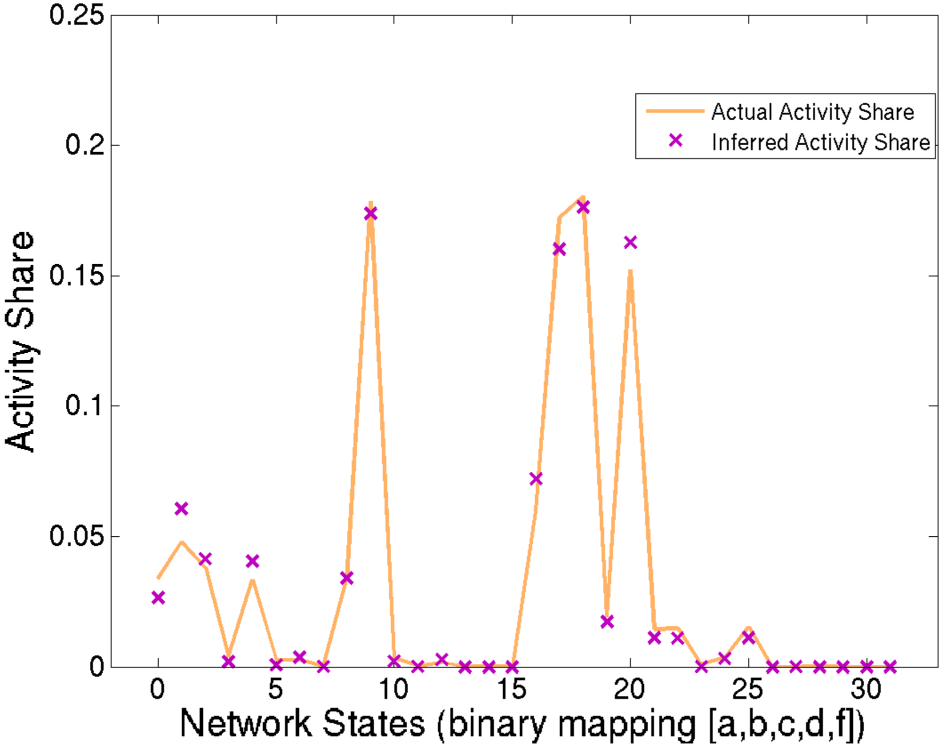


- Composed of timing and up to 4 radio daughtercards
- Xilinx Virtex Pro-II
  - FPGA ➔ customize the operations of the radio device (without performance penalty)
  - PPC405 ➔ support higher communications layers (MAC) design with C-like programming
- Interfacing via USB, Ethernet, Serial (RS-232), MGT ports (and pins)

Activity Share Inference

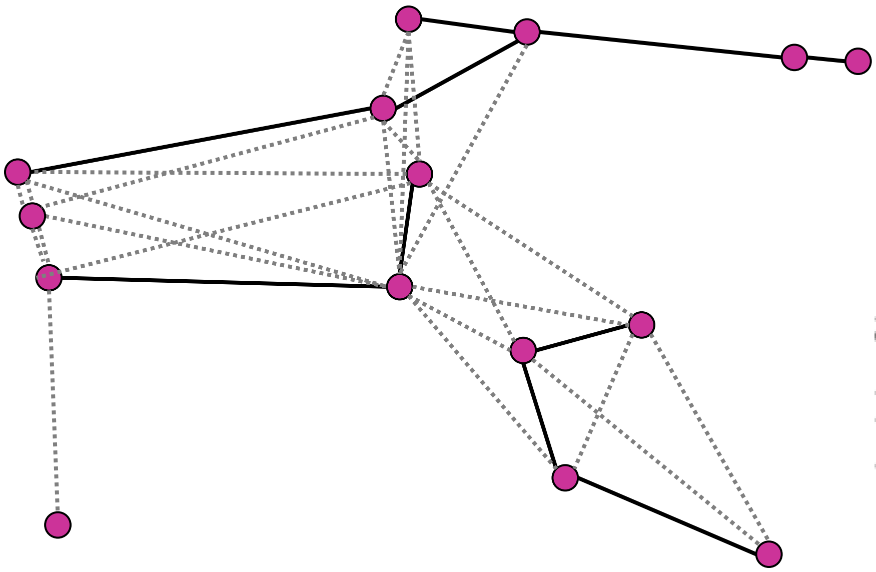
Predicted vs. Actual Activity Share (testbed results)

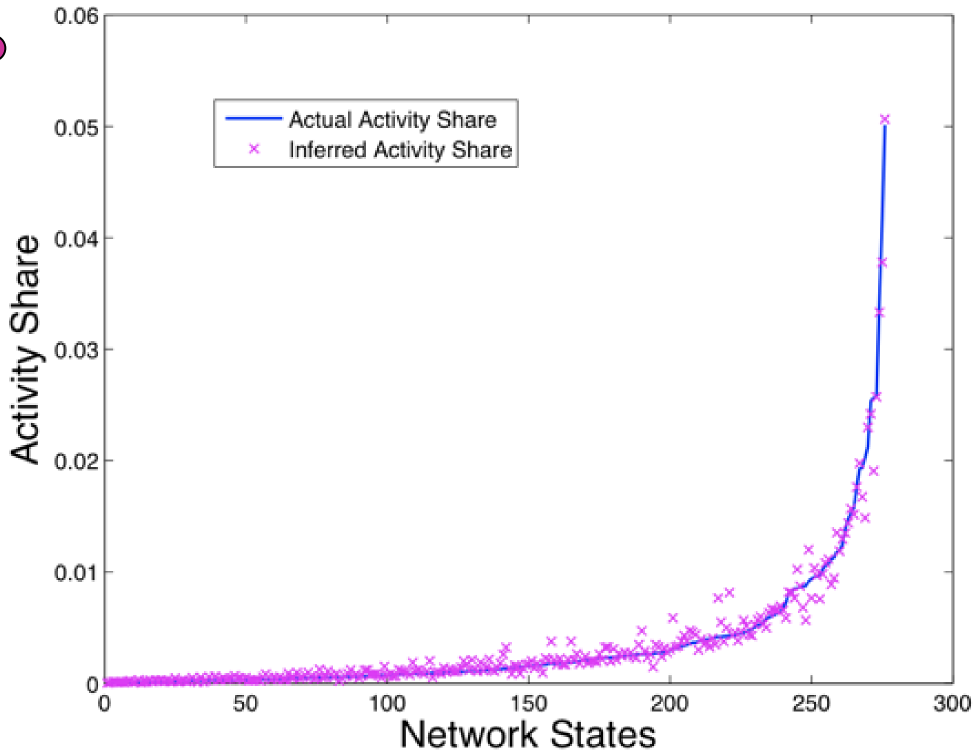




Activity Share Inference

Predicted vs. Actual Activity Share (simulations results)

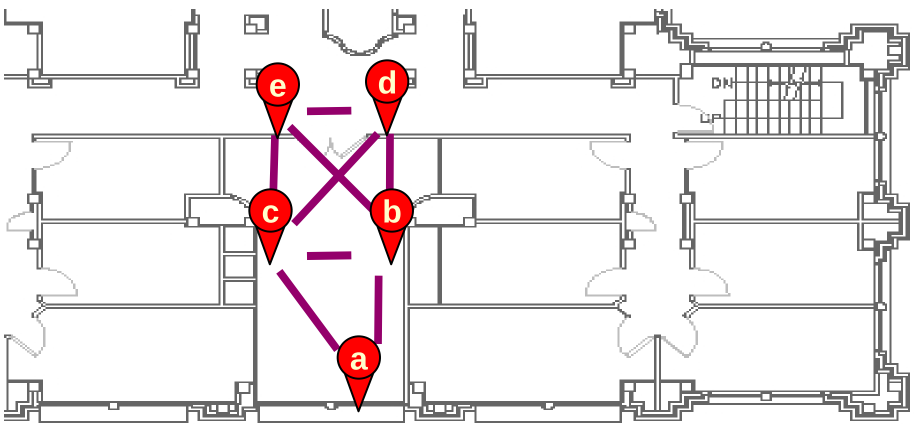


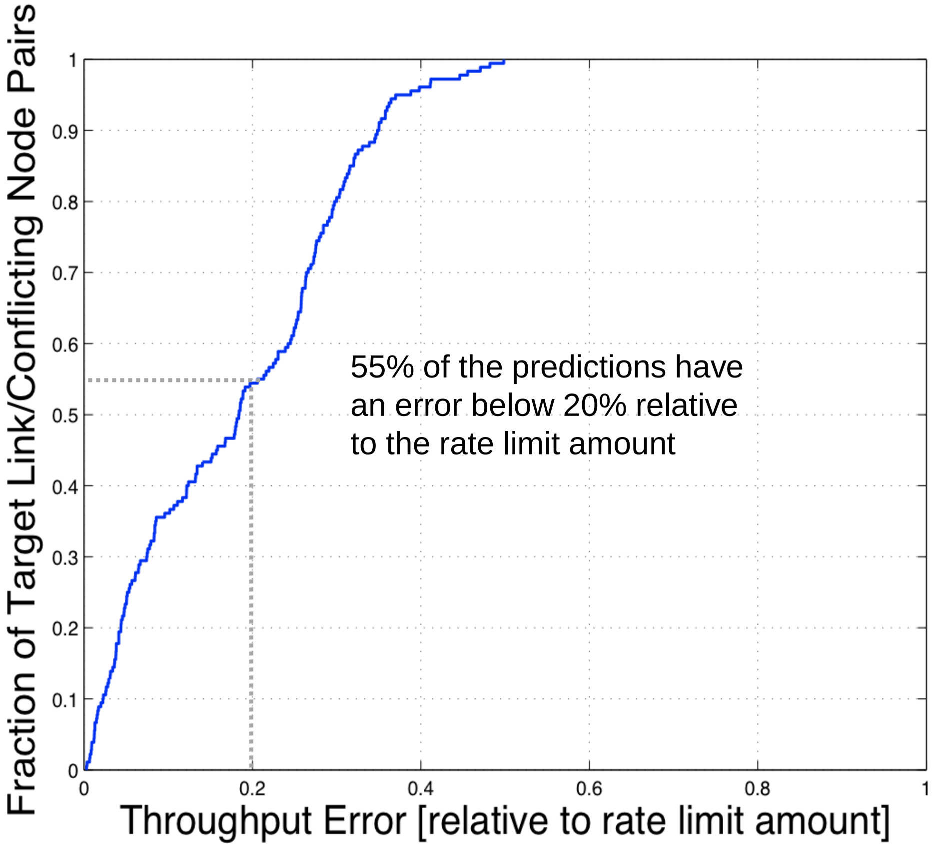


Accurate Inference results both for testbed and simulations

Throughput Prediction

Rate-limiting different conflicting nodes for the same kbps quantity (testbed results)



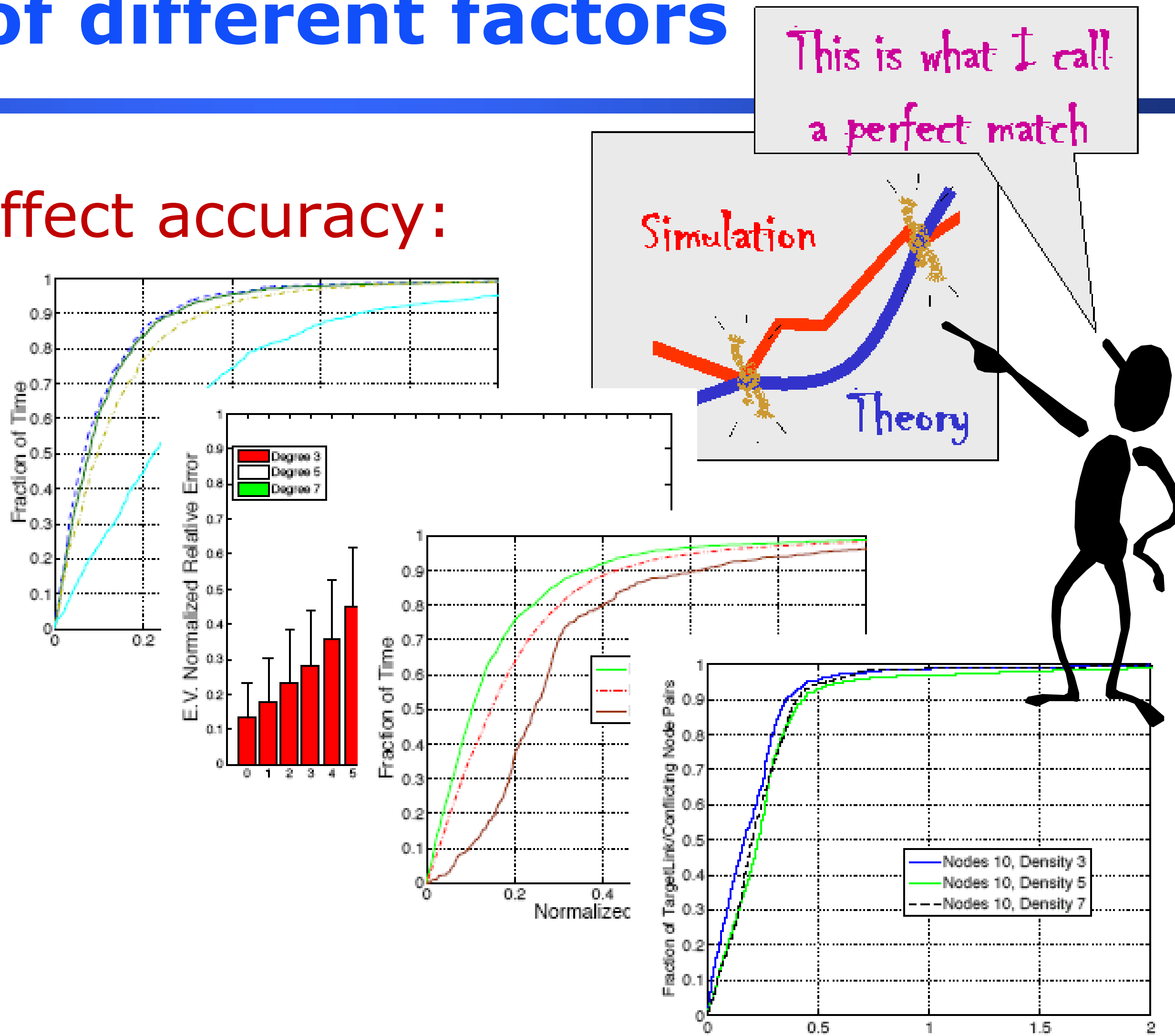


High Accuracy in predicting the candidates to be rate-limited
Low error in predicting the gain

Effect of different factors

Many factors can affect accuracy:

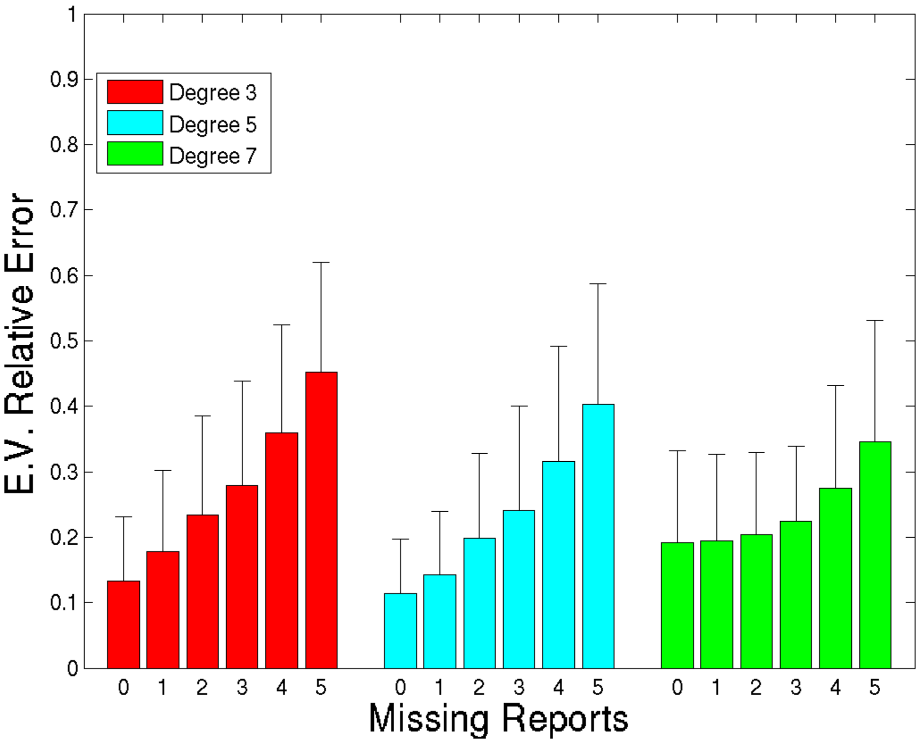
- Density
- Traffic
- Report Intervals
- Report Losses



Thorough factor evaluation can be found in the paper

Robustness to Report Losses

- Under congestion, reports can be lost and not reach the manager
- How much accuracy do we lose?
- *High density:*
  - *reports of neighboring nodes are related* $\Rightarrow$ more robust to report losses

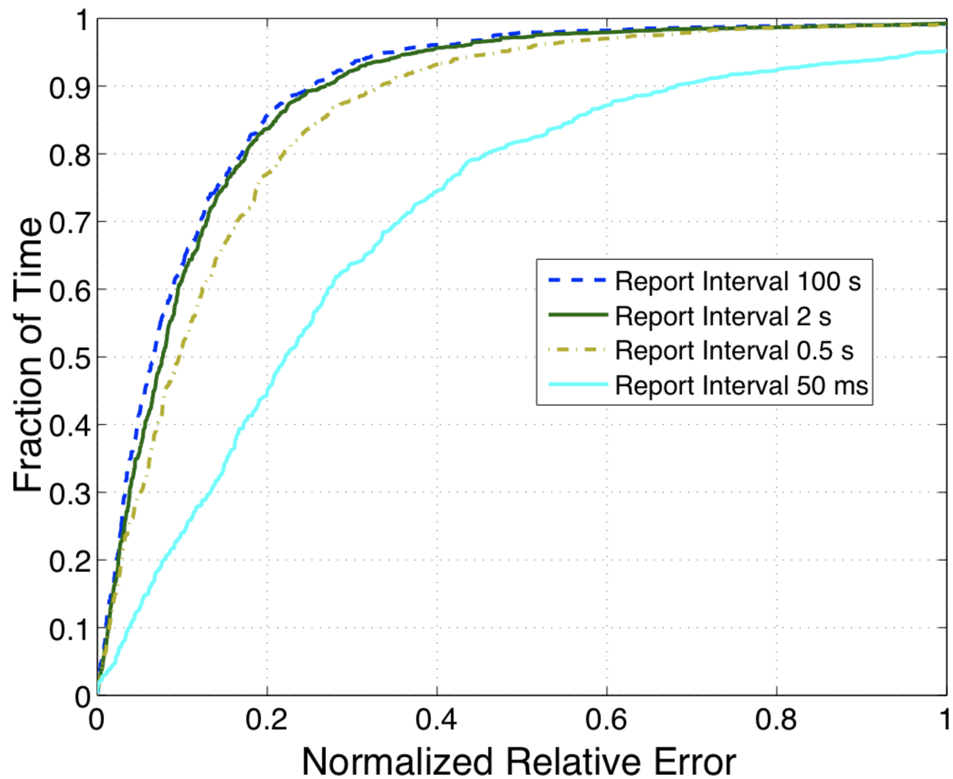


- ns2 simulations
- 10 nodes
- various densities (3 to 7 neighbors)
- all possible combinations of 1 to 5 lost reports

Few losses have a mild effect on inference accuracy

Impact of Report Interval

- Simulations
- Report Interval
  - large: favors statistical significance, low overhead
  - small: favors responsiveness to network changes



- Avg. relative errors 4.1% (20 s), 7.6% (2 s), 10.2% (500 ms), 29% (100 ms)

The manager can adapt the report interval to the network dynamics with small penalty on accuracy

Summary of Inference and Management

- Understanding coordination is key to identifying:
  - causes of under-served links
  - potential throughput gains of rate-limiting conflicting nodes
- Activity Share captures coordination
- We showed:
  - How to infer the Activity Share
  - How to use the Activity Share for throughput predictions

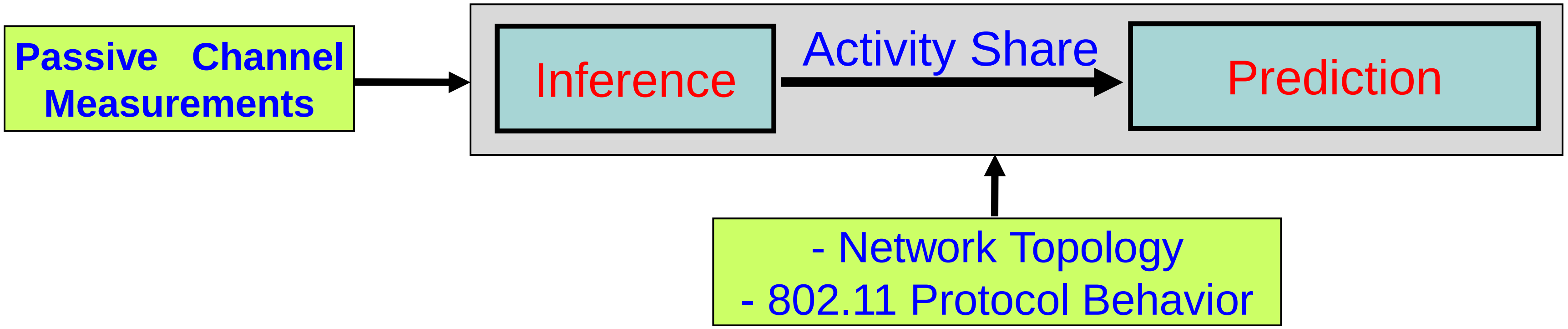
Questions

MIDAS

Management, Inference, and Diagnostics using Activity Share

1. Inference - Infer link coordination
   - Input: statistics from the nodes
   - Output: measure of Coordination

2. Prediction- Determine link interactions and identify corrective actions
   - Input: measure of Coordination
   - Output: Management actions to achieve a target objective



RICE