Greening Backbone Networks

Shutting Off Cables in Bundled Links

Will Fisher, Martin Suchara,
and Jennifer Rexford

Princeton University

Why Focus on Networks?

- Saving energy in servers vs. networks

- Potential for savings in wired networks
  - $2 billion annual power bill in USA alone
  - Over-engineering and diurnal patterns
  - Power draw does not scale with load

Power Consumption of Network Hardware

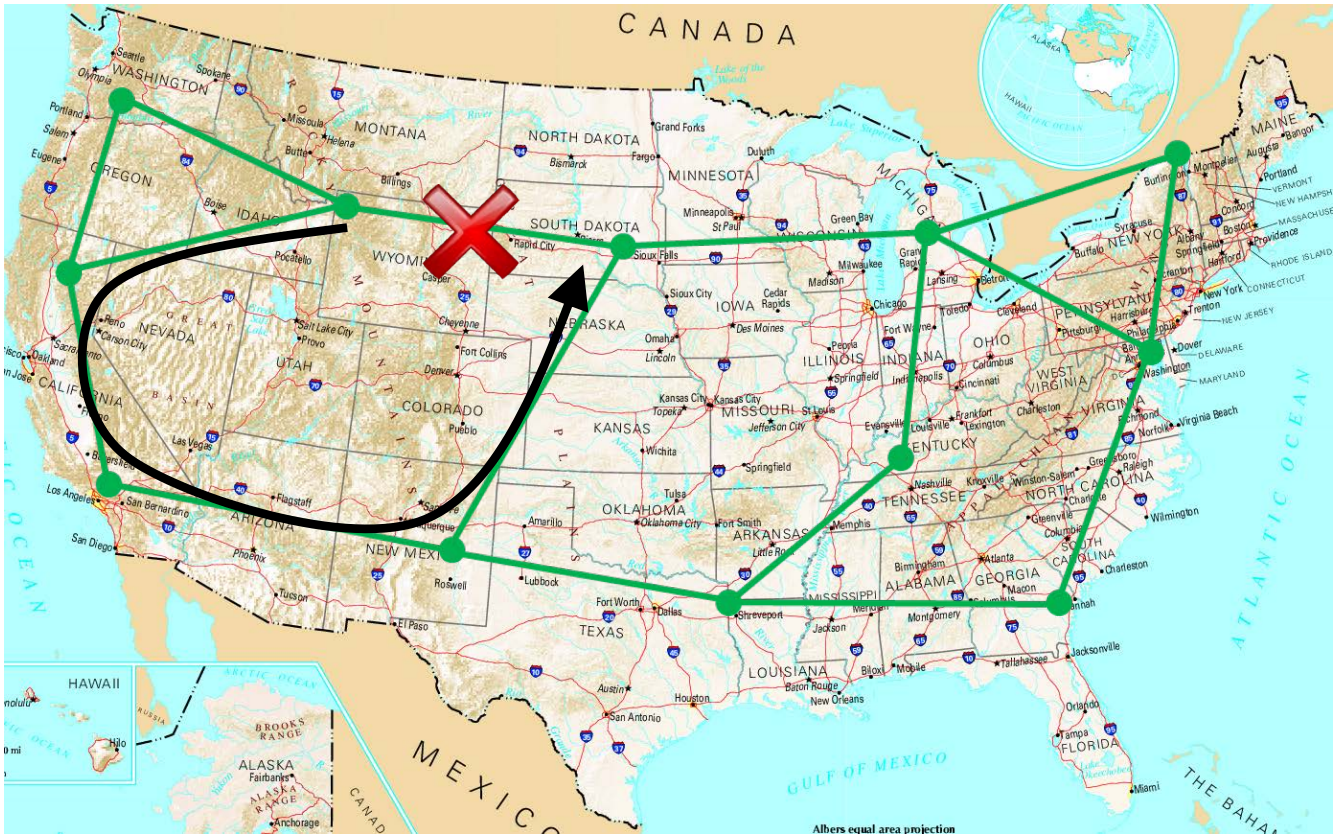
- Router may consume 200 - 400W plus 100 W per line card
  - Power draw does not scale with load

chassis

line card

Turn Off Unneeded Routers / Links?

- Turning off links has drawbacks
  - Increased RTT, fragility, etc.

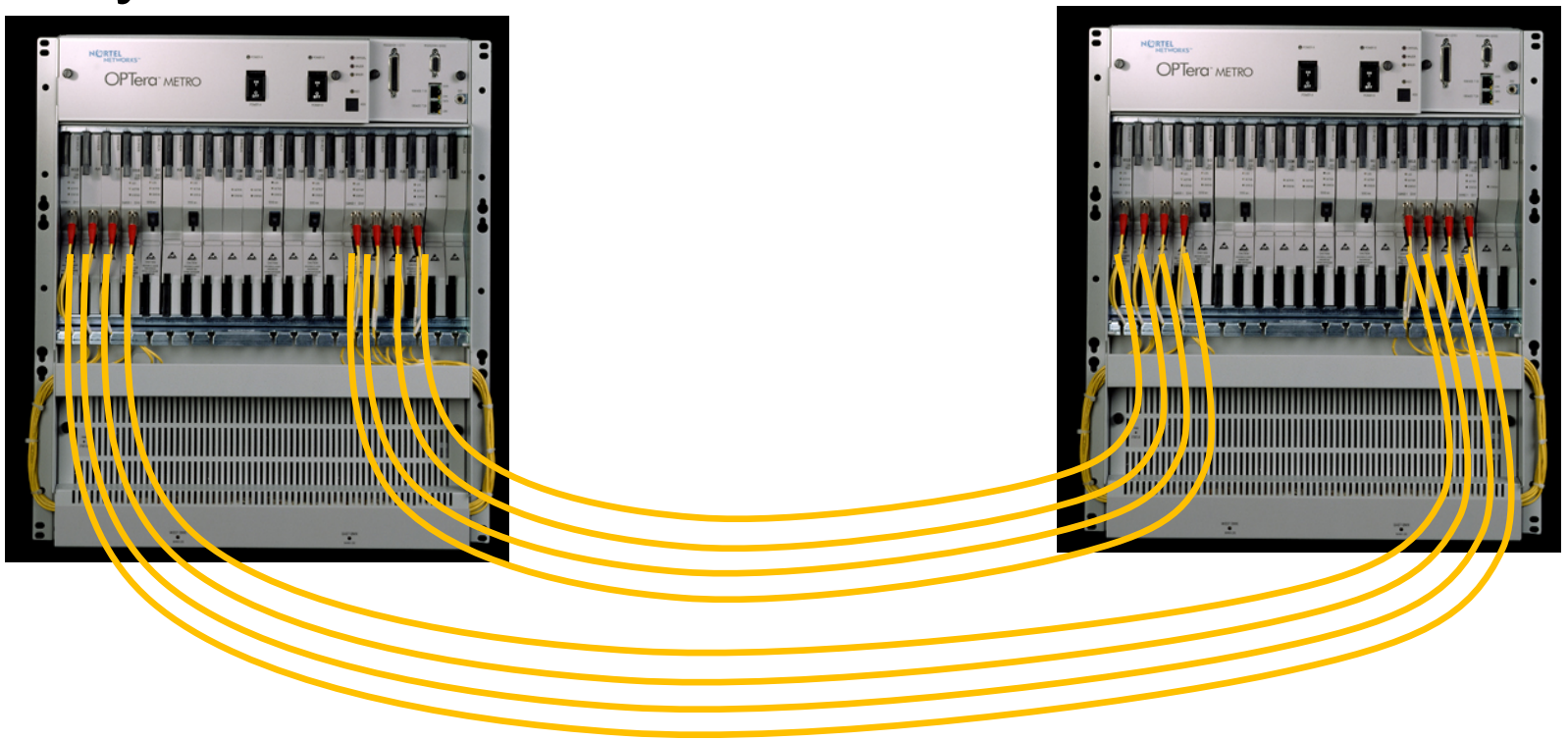


Links in Large Backbone Networks

- Links come in bundles
  - Gradual upgrades, hardware costs, etc.

City A

City B



Links in Large Backbone Networks

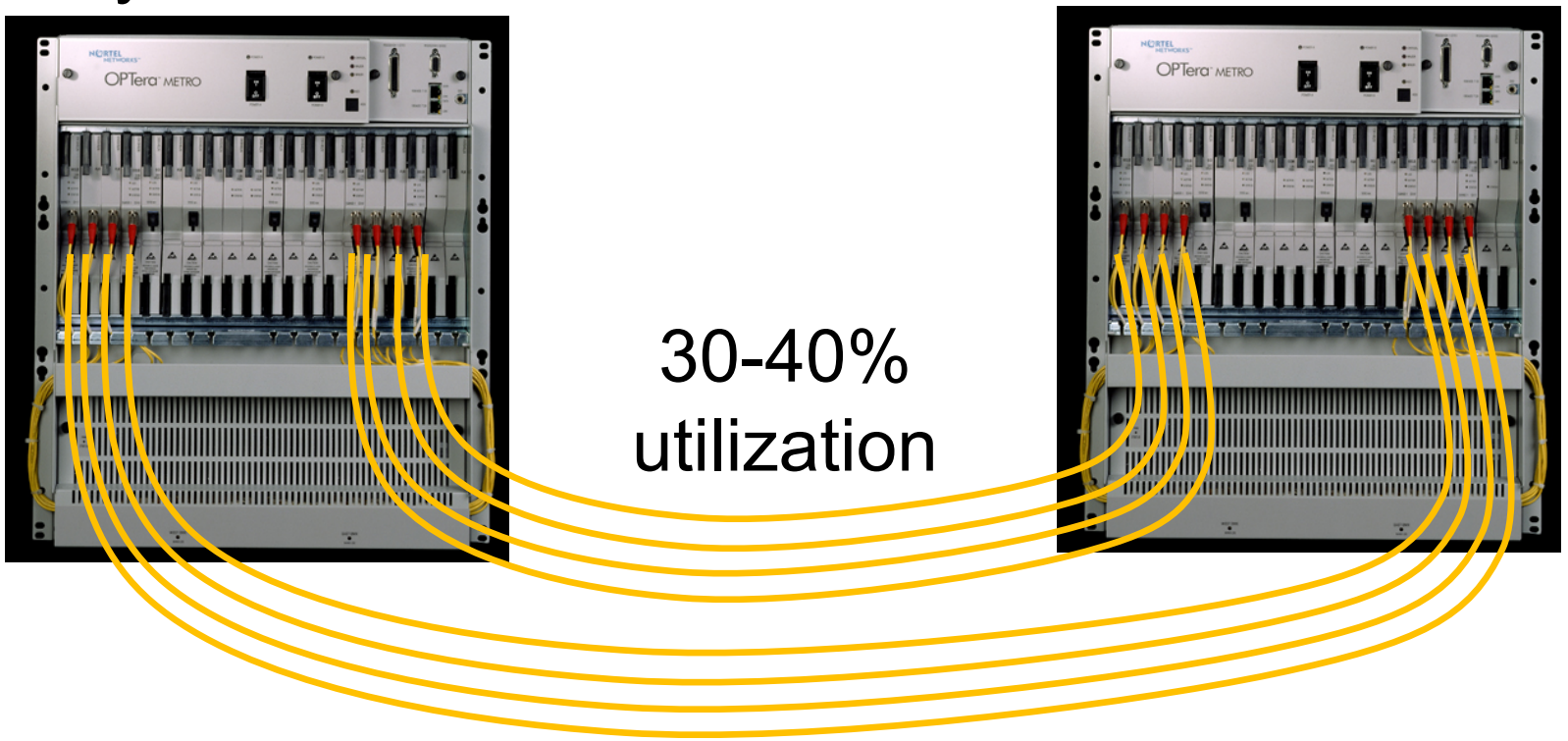
- Links come in bundles
  - Gradual upgrades, hardware costs, etc.
- Bundled link behaves as a single logical link
- In large networks
  - Majority of links bundled
  - Bundle sizes 2 – 20 cables per link

The Problem…

- Powering all cables is wasteful

City A

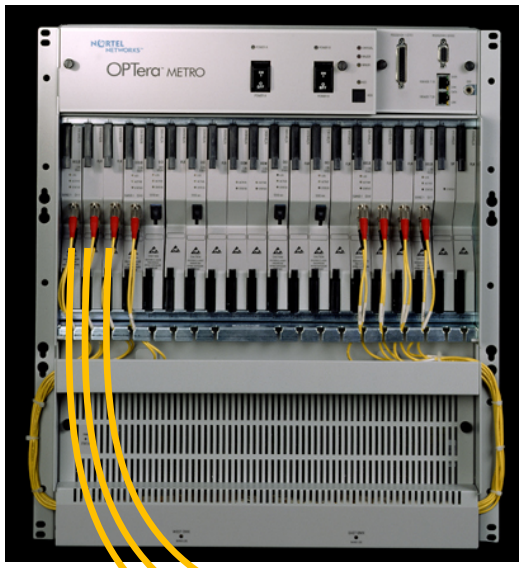
City B



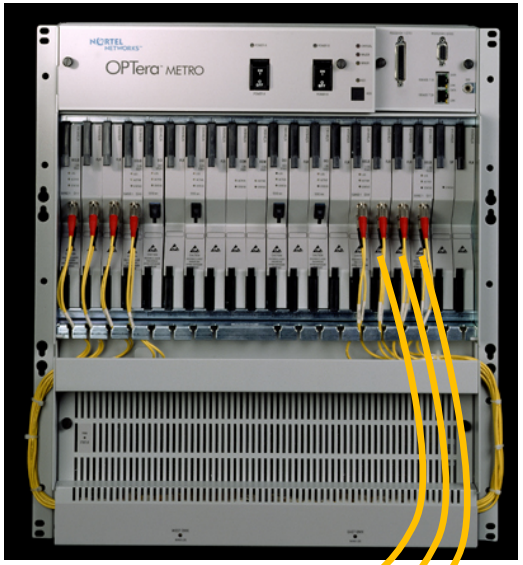
Proposed Solution

- Only power up cables that are needed
  - Transparent to routing protocols

City A



City B



80%
utilization

Overview

I. Optimization problem formulation & solution

II. Experimental evaluation of energy savings

III. Conclusion

The Problem

- Algorithm used by network operators
  - Input: network configuration and load
  - Output: list of powered cables
- Integer linear program:

  min # powered cables
  s.t. link loads ≤ capacities
  flow conservation
  carries all traffic demands

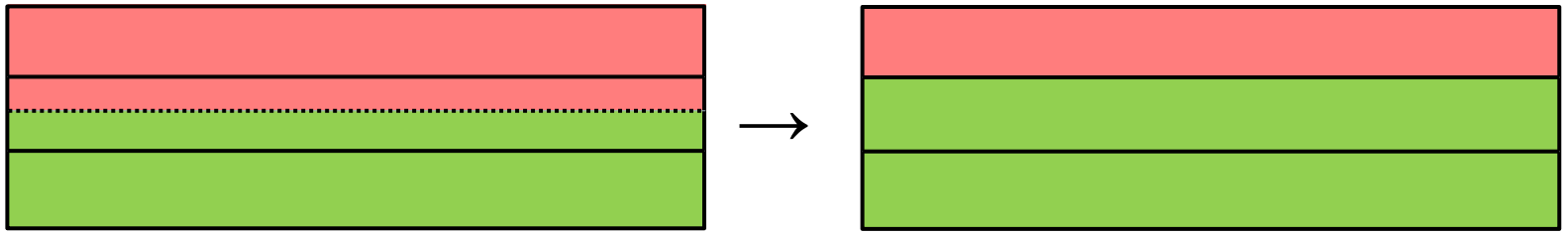
- NP-hard in general → need heuristics

Related Tractable Problem

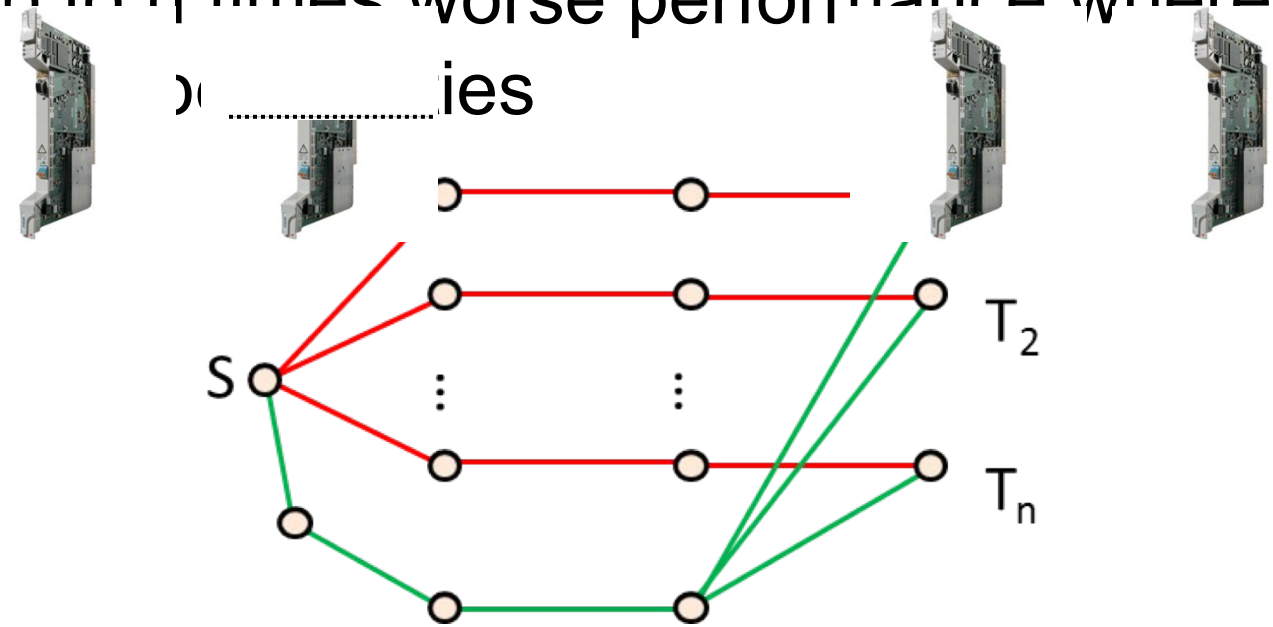
- How would the solution look like if energy consumption was proportional to link load
- Minimize sum of link loads rather than the number of powered cables
  - Fractional vs. integer linear program
- Benefits: tractable, provides upper and lower bound on potential power saving, starting point for heuristics

First Attempt – Naïve Solution

- Always round up:



- Up to n times worse performance where n



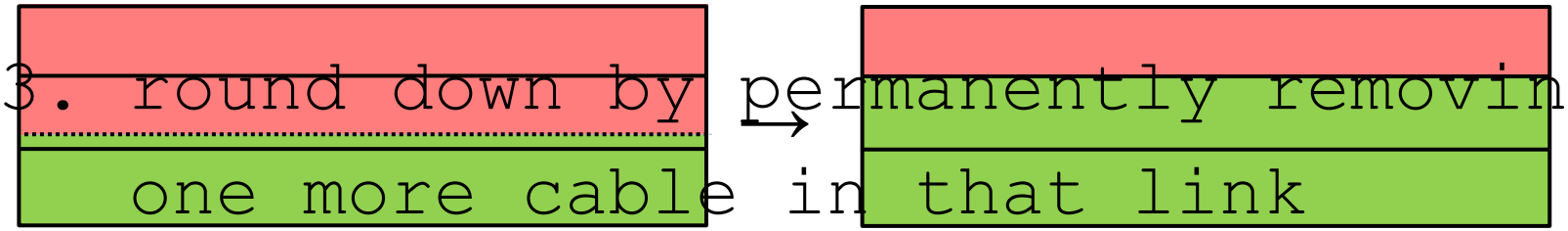
Three Increasingly Sophisticated Heuristics

- Start with the naïve solution and perform local search

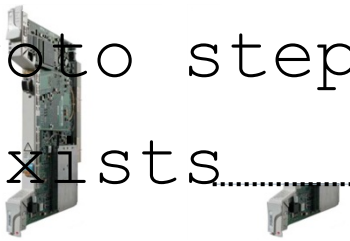
1. Fast Greedy Heuristic (FGH)
2. Exhaustive Greedy Heuristic (EGH)
3. Bi-level Greedy Heuristic (BGH)

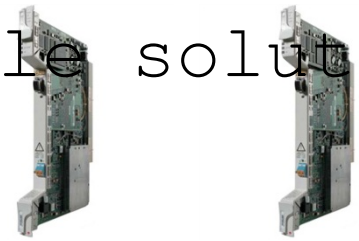
The Fast Greedy Heuristic (FGH)

1. solve the fractional linear program and round up
2. identify link with greatest rounding up
3. round down by permanently removing one more cable in that link



4. goto step 1 if feasible solution exists





Overview

I. Optimization problem formulation & solution

II. Experimental evaluation of energy savings

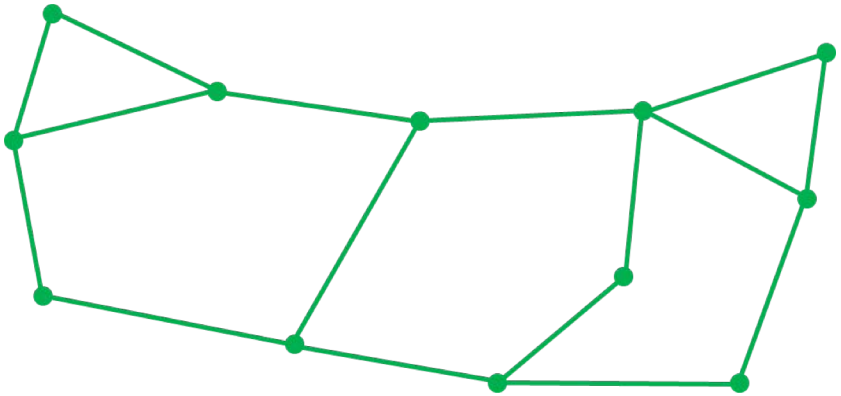
III. Conclusion

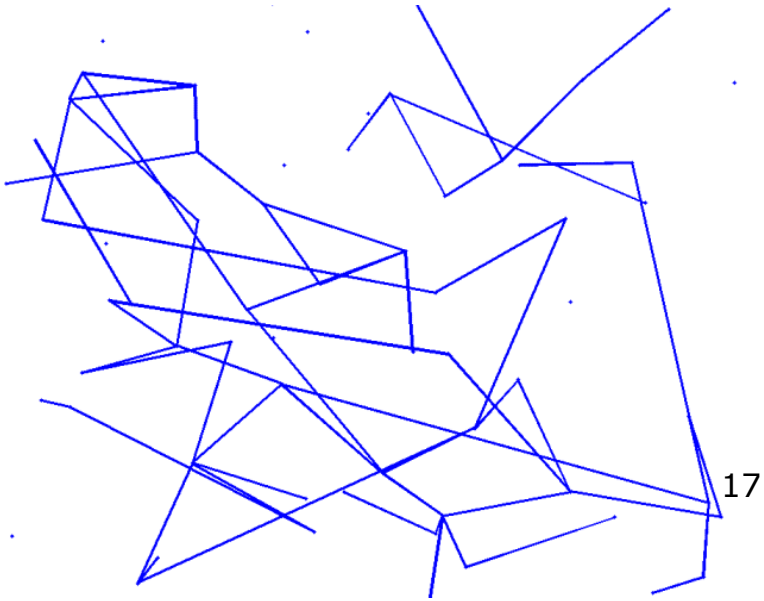
Experimental Setup

- Determine energy savings and ability to run algorithms in real time
- Solve the simple linear program using AMPL / CPLEX
- Experiments repeated for bundle sizes 1 to 12

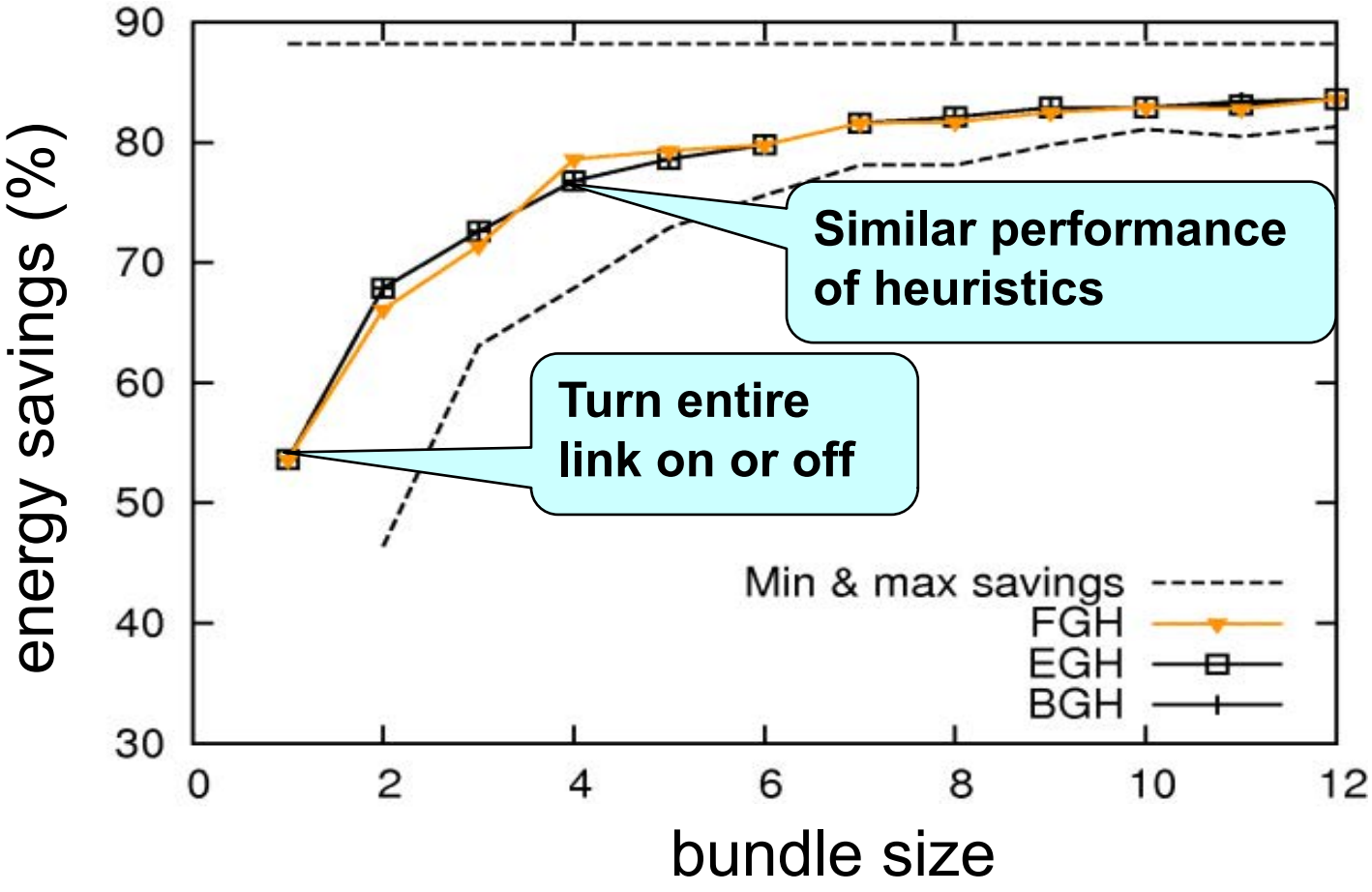
Topologies

- Abilene with real measured traffic demands
- Waxman and hierarchical topologies



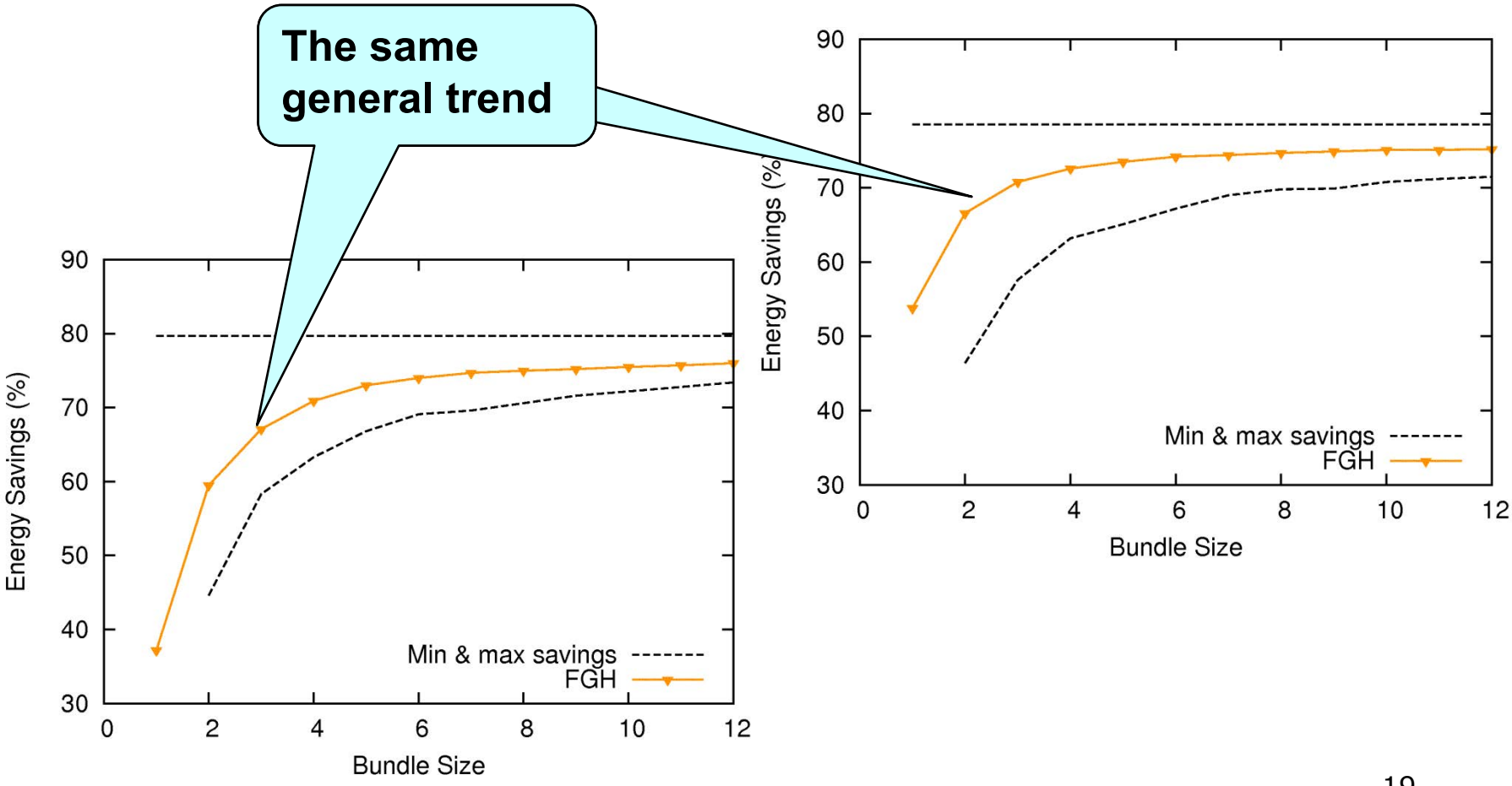


Energy Savings– Abilene Topology



Energy savings depend on the bundle size.

Waxman and Hierarchical Graphs



Execution Time

Topology	FGH	EGH	BGH
Abilene	8 sec	50 sec	5 min
Large synthetic	minutes	minutes to hours	

FGH is well suited for real-time execution

Overview

I. Optimization problem formulation & solution

II. Experimental evaluation of energy savings

III. Conclusion

Conclusion

- Powering all links in a bundle usually not needed

- Design and evaluation of simple heuristics
  - Significant energy savings
  - Low computational complexity
  - Can choose the simplest heuristic (FGH)

Thank You!