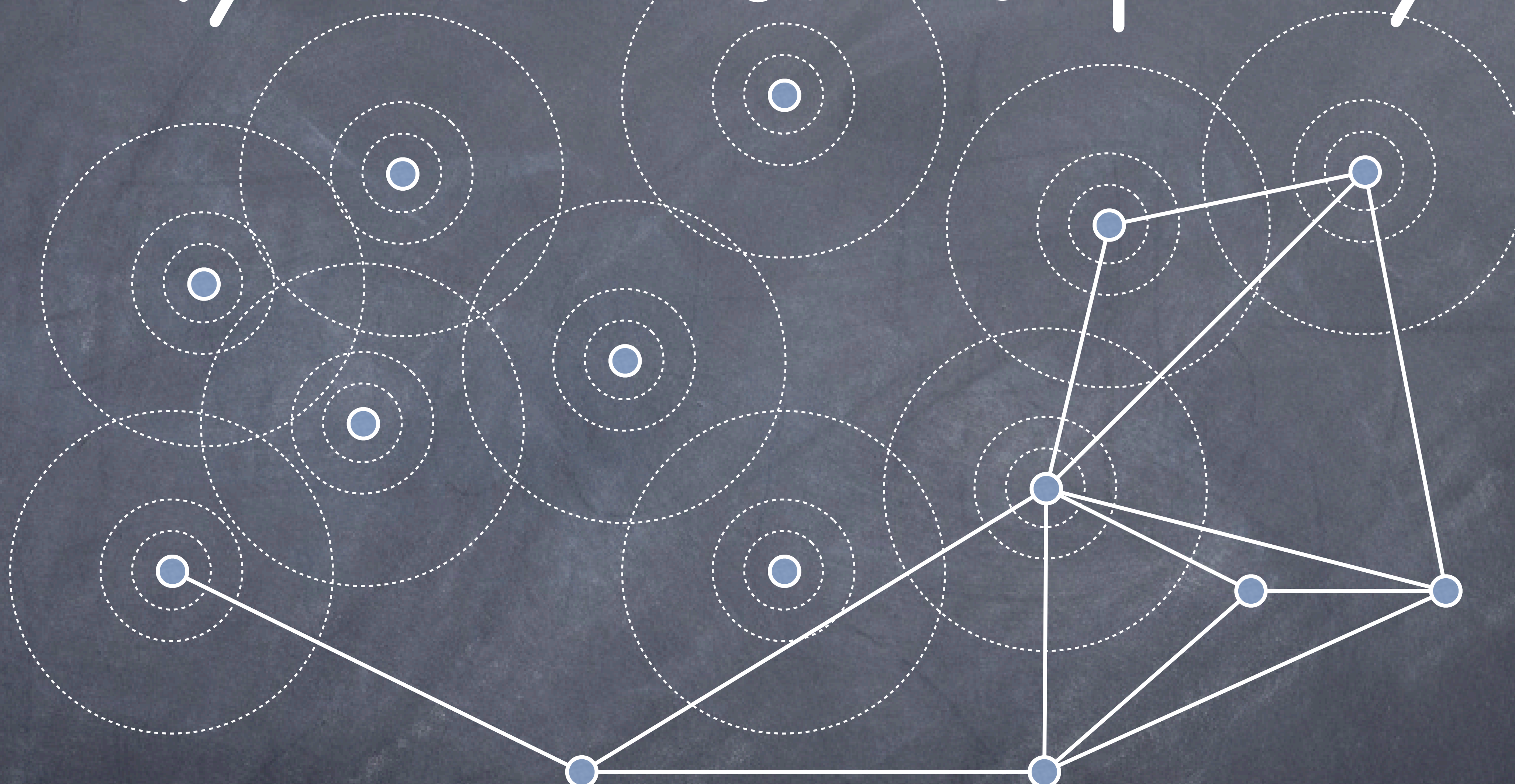
Equivalence Beyond Random Channels

Michelle Effros
California Institute of Technology

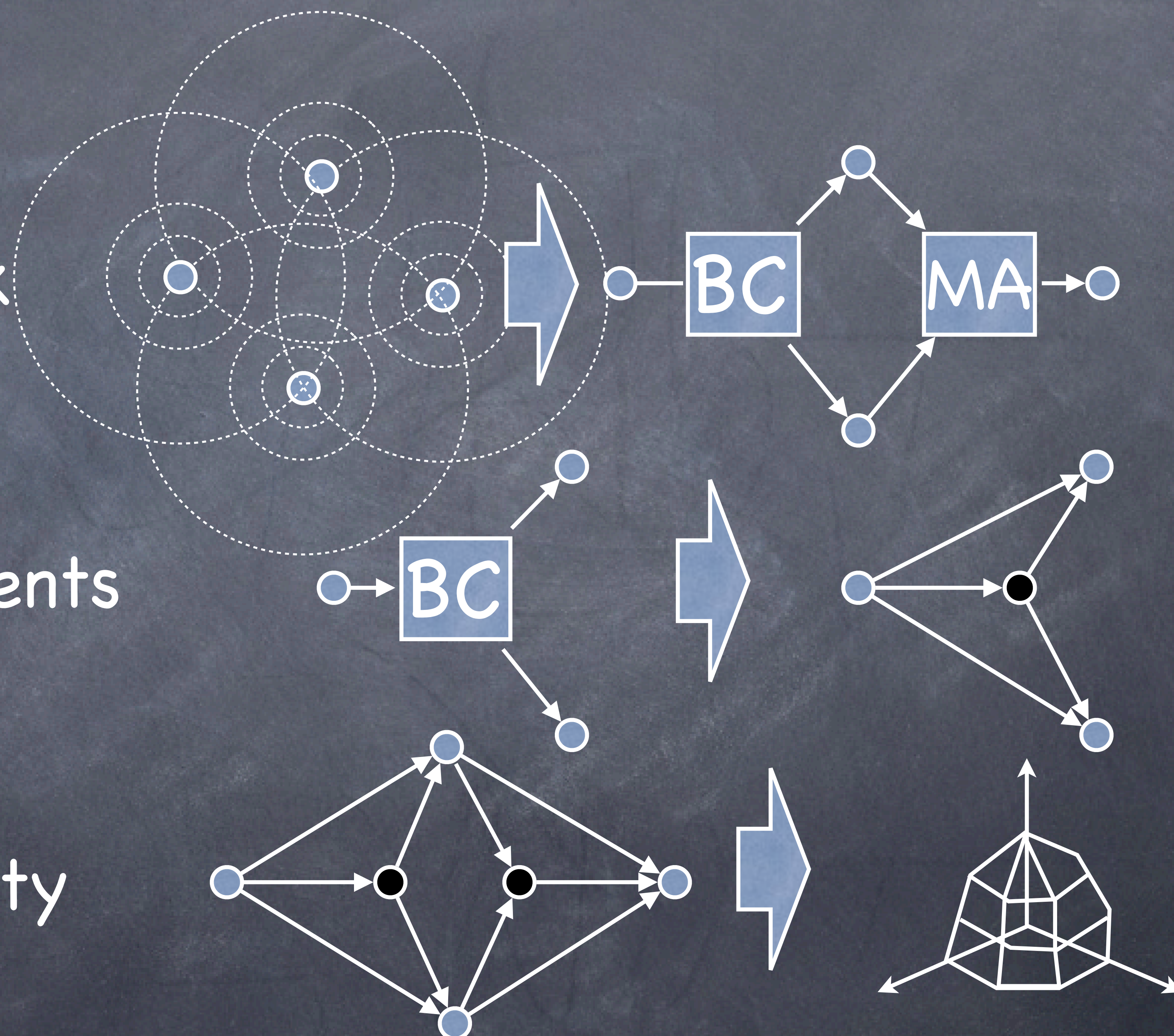
Underlying Challenge:
Analyze Network Capacity



- Given: Arbitrary network, arbitrary demands
- Goal: Find the network capacity

Overview

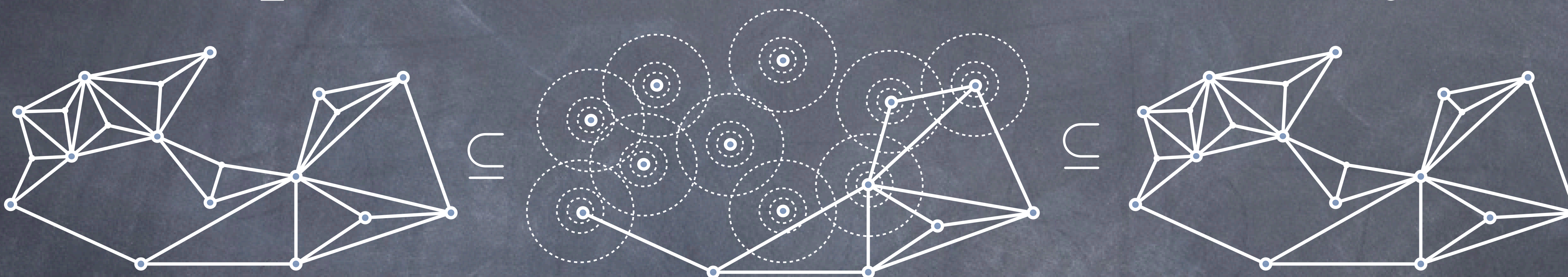
- Factor network
- Model components
- Analyze capacity



M. Effros, January 26, 2011

Bounding Networks

$$\mathcal{N}_L \subseteq \mathcal{N} \subseteq \mathcal{N}_U$$



$$\mathcal{C}(\mathcal{N}_L) \subseteq \mathcal{C}(\mathcal{N}) \subseteq \mathcal{C}(\mathcal{N}_U)$$

- Any collection of connections that is feasible on $\mathcal{N}_L$ is also feasible on $\mathcal{N}$
- Any collection of connections that is feasible on $\mathcal{N}$ is also feasible on $\mathcal{N}_U$

How well do we do?

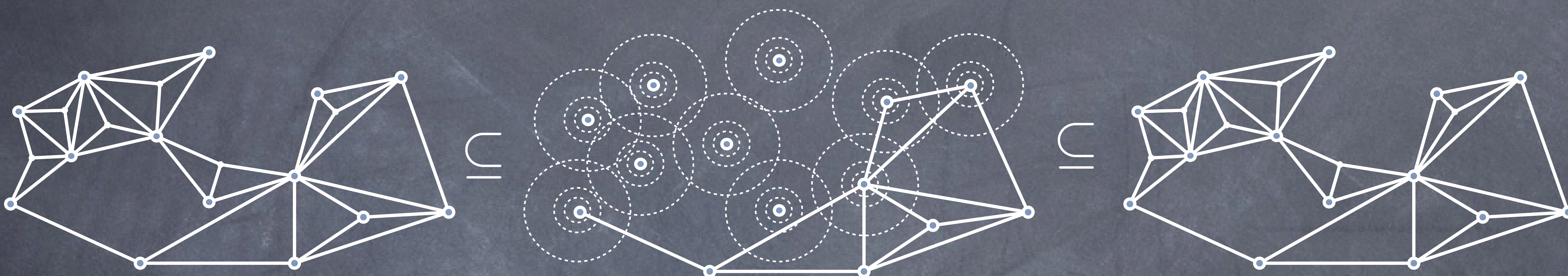
$$\mathcal{N}_L \subseteq \mathcal{N} \subseteq \mathcal{N}_U$$

$$\mathcal{C}(\mathcal{N}_L) \subseteq \mathcal{C}(\mathcal{N}) \subseteq \mathcal{C}(\mathcal{N}_U)$$

<u>Three strategies for tackling this question:</u>

- Calculate capacities
- Multiplicative bounds
- Additive bounds

1. Calculate Capacities

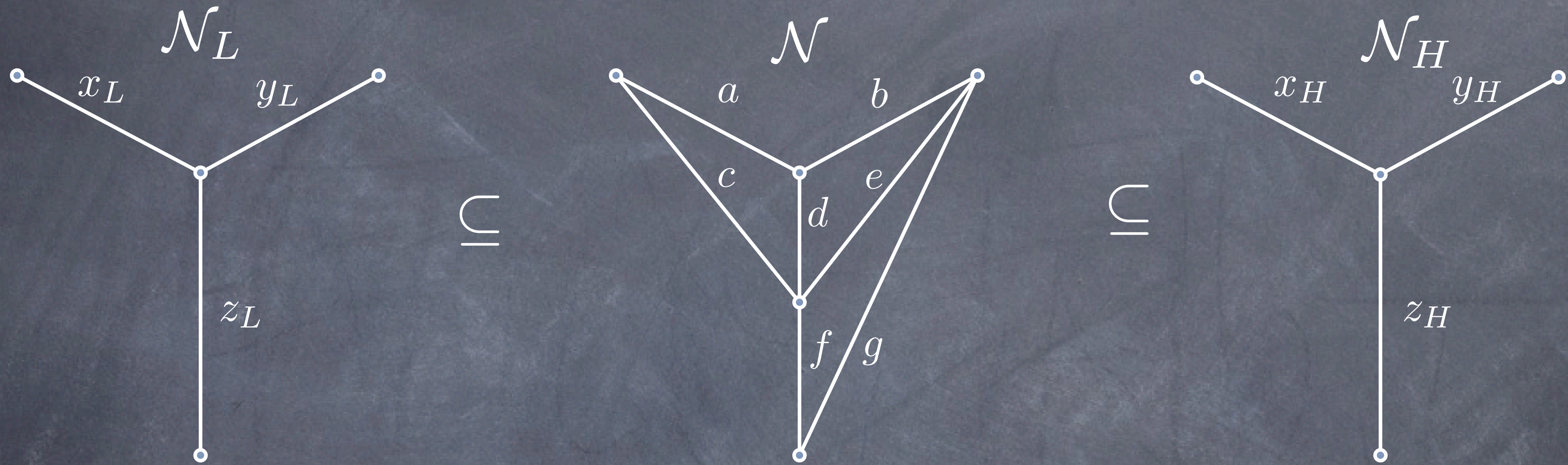


- Tools exist for bounding network coding capacities
  [Song et al. 2003], [Kleinberg et al. 2006], [Subramanian et al. 2008]
- Each solves an LP
- Complexity of LP grows with network size
- Only capacities of small networks can be bounded

M. Effros, January 26, 2011

Hierarchical Analysis

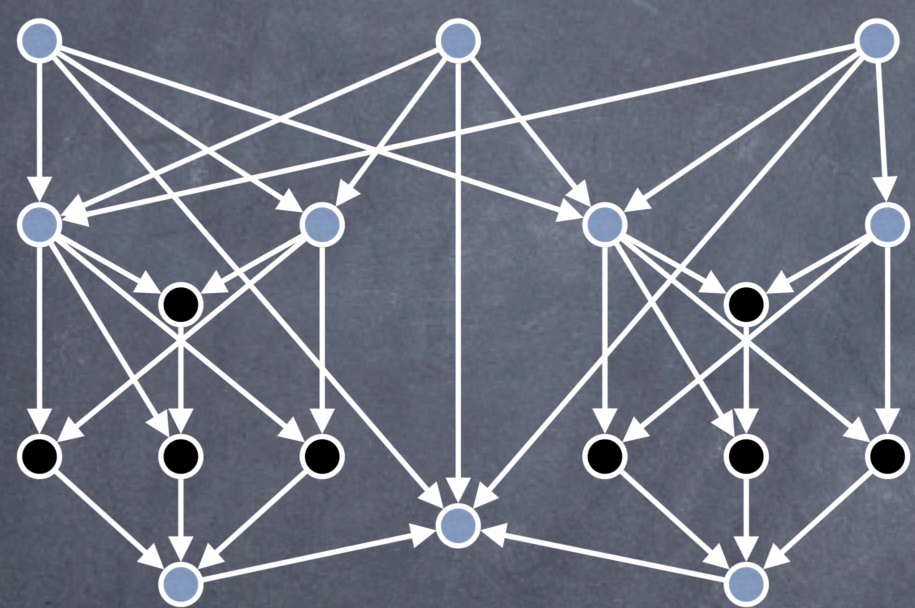
[Ho, Effros, Jalali 2010]



- LP finds minimal values of x_H, y_H, z_H to emulate $\mathcal{N}$
- Node 1 creates a units of flow type A, c units of C
- Node 2 creates b units of B, e units of E, g units of G
- a units of A + b units of B => d units of type D
- c units of C + d units of D + e units of E => f units of F
- Node 3 receives f units of F and g units of G

Hierarchical Network Capacity Analysis

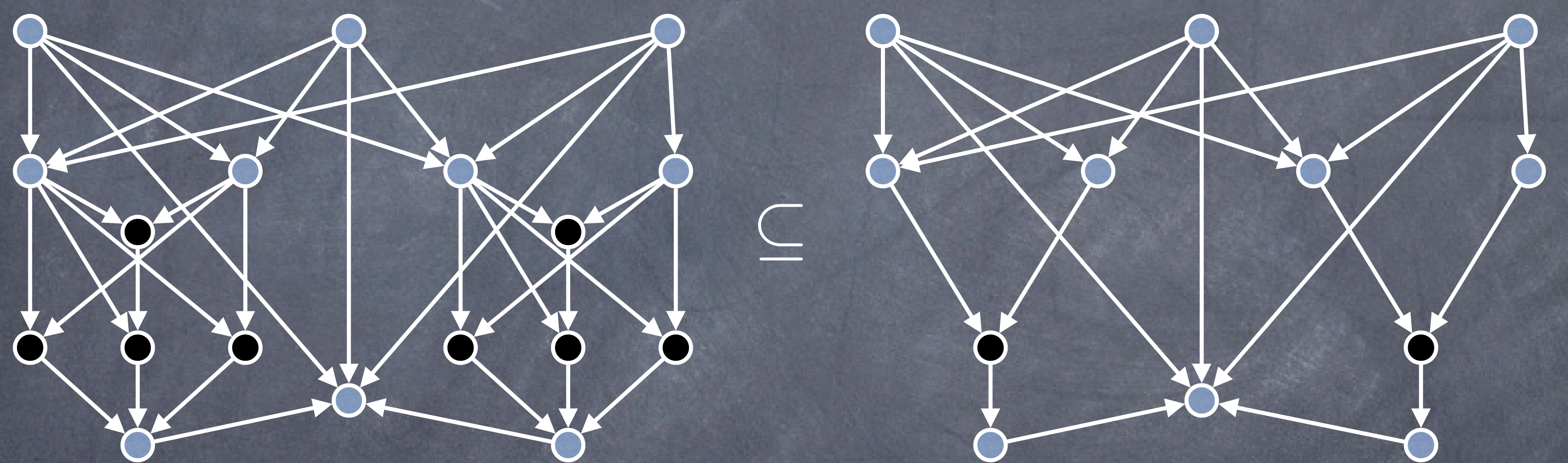
[Jalali, Ho, & Effros in preparation 2010]



M. Effros, May 26, 2010

Hierarchical Network Capacity Analysis

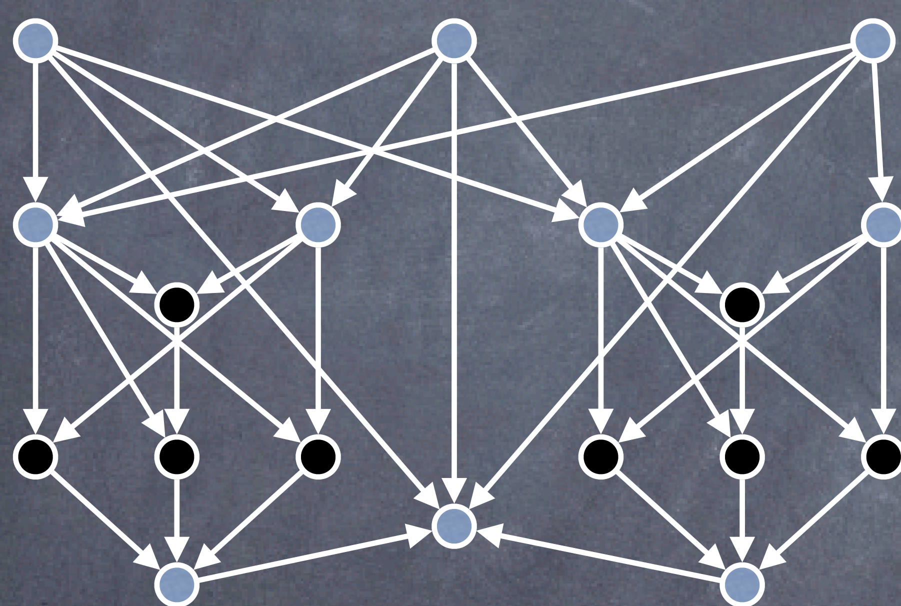
[Jalali, Ho, & Effros in preparation 2010]



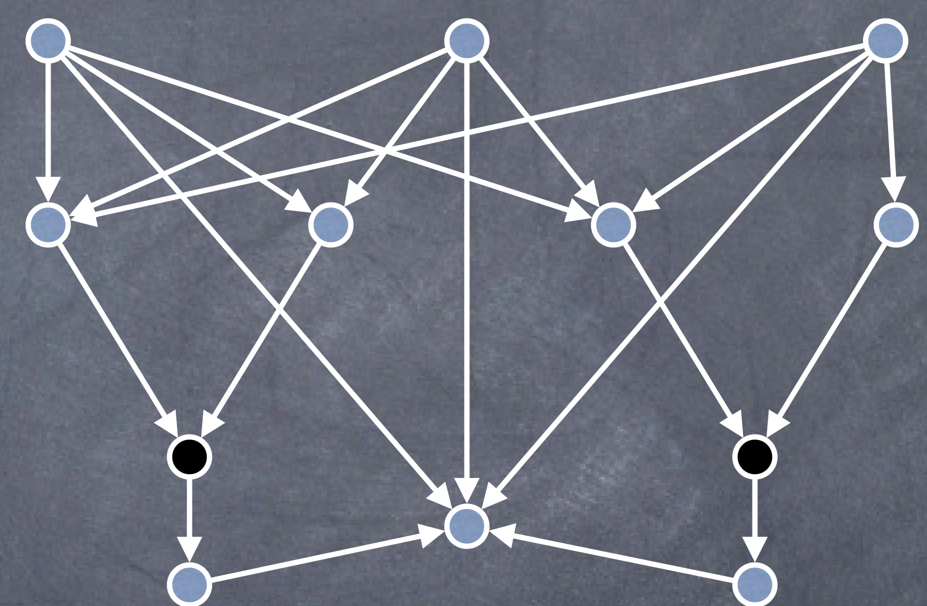
M. Effros, May 26, 2010

Hierarchical Network Capacity Analysis

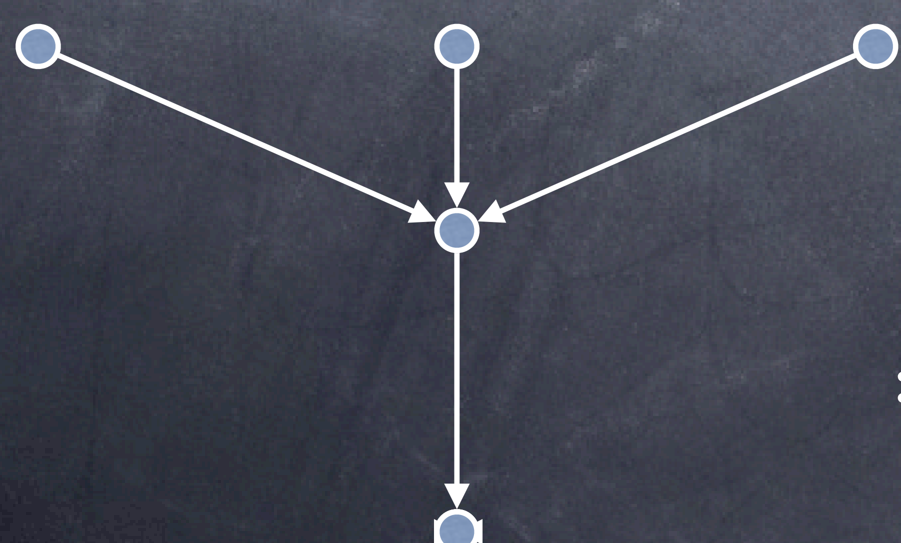
[Jalali, Ho, & Effros in preparation 2010]



$\subseteq$



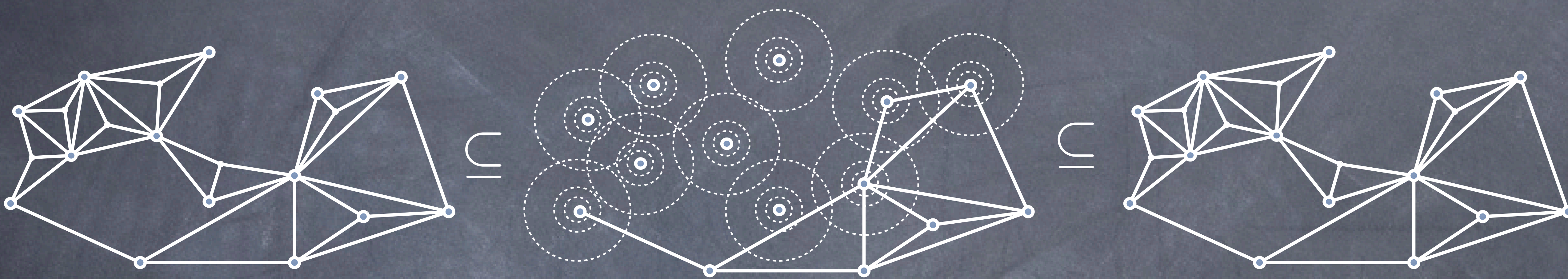
$\subseteq$



Bounding large networks
by smaller networks
=> computational feasibility

M. Effros, May 26, 2010

2. Multiplicative Bounds [Effros, 2010]



- Multiplicative bound on capacity

$$\mathcal{C}(\mathcal{N}_L) \supseteq \left(\min_{e \in E} \frac{C_L(e)}{C_U(e)} \right) \mathcal{C}(\mathcal{N}_U)$$

- Applies to **all** types of connections
- Fails when topologies differ

M. Effros, January 26, 2011

3. Additive Bounds [Effros, 2010]

- Additive bound on capacity

$$\mathcal{C}_{CS}(\mathcal{N}_L) \supseteq \mathcal{C}_{CS}(\mathcal{N}_U) - \mathbf{1} \cdot \max_{\text{cut}}[C_U(\text{cut}) - C_L(\text{cut})]$$

- Succeeds even when topologies differ

- Applies only when cuts characterize capacity
  (e.g., multicast, multi-source multicast, single-source with non-overlapping demands, single-source w/non-overlapping + multicast demands)

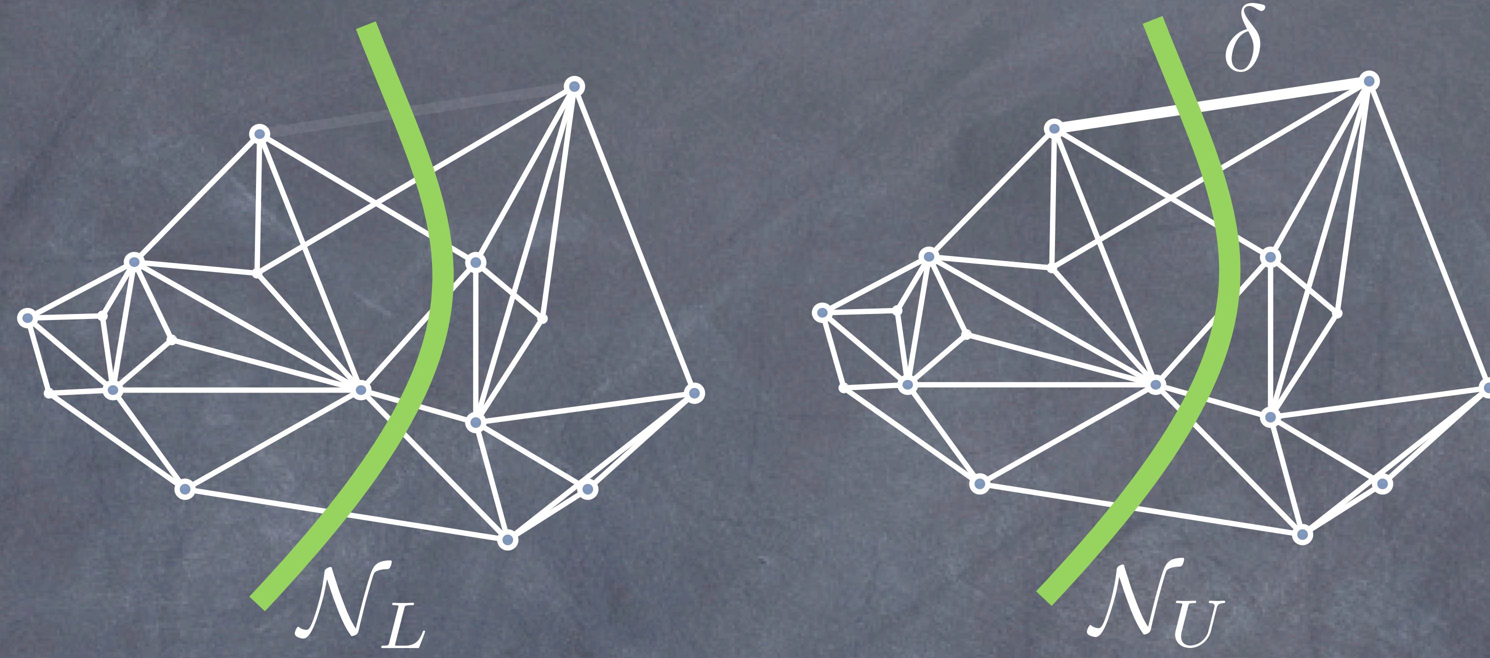
M. Effros, January 26, 2011

3. Additive Bounds (cont.)

[Ho, Effros, Jalali 2010] [Jalali, Effros, Ho 2011]

- Can removing a single edge of capacity δ reduce the capacity by more than $1 \cdot \delta$??
- No (so far...)
  - When cut-set bounds are tight
  - Single-source, arbitrary demands
  - Networks with bottlenecks

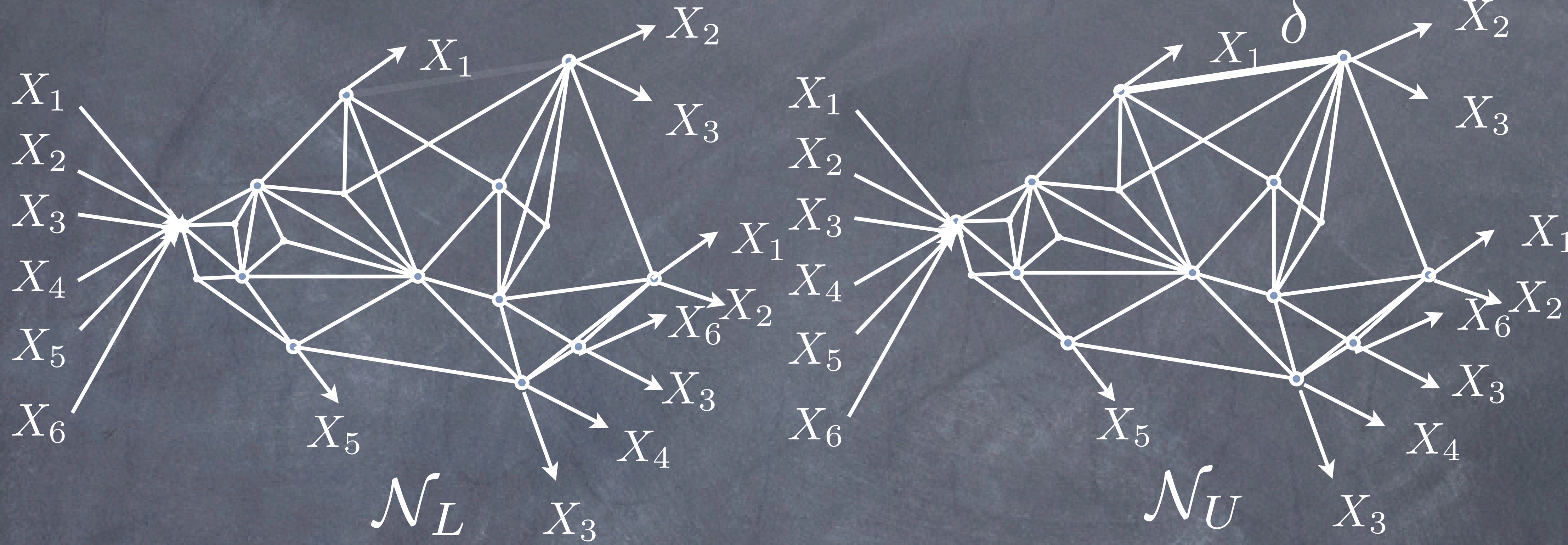
When Cut-Set Bounds are Tight



- Cut capacities determine capacity
- Removing a single edge of capacity δ changes each cut capacity by at most δ

Single Source, Arbitrary Demands

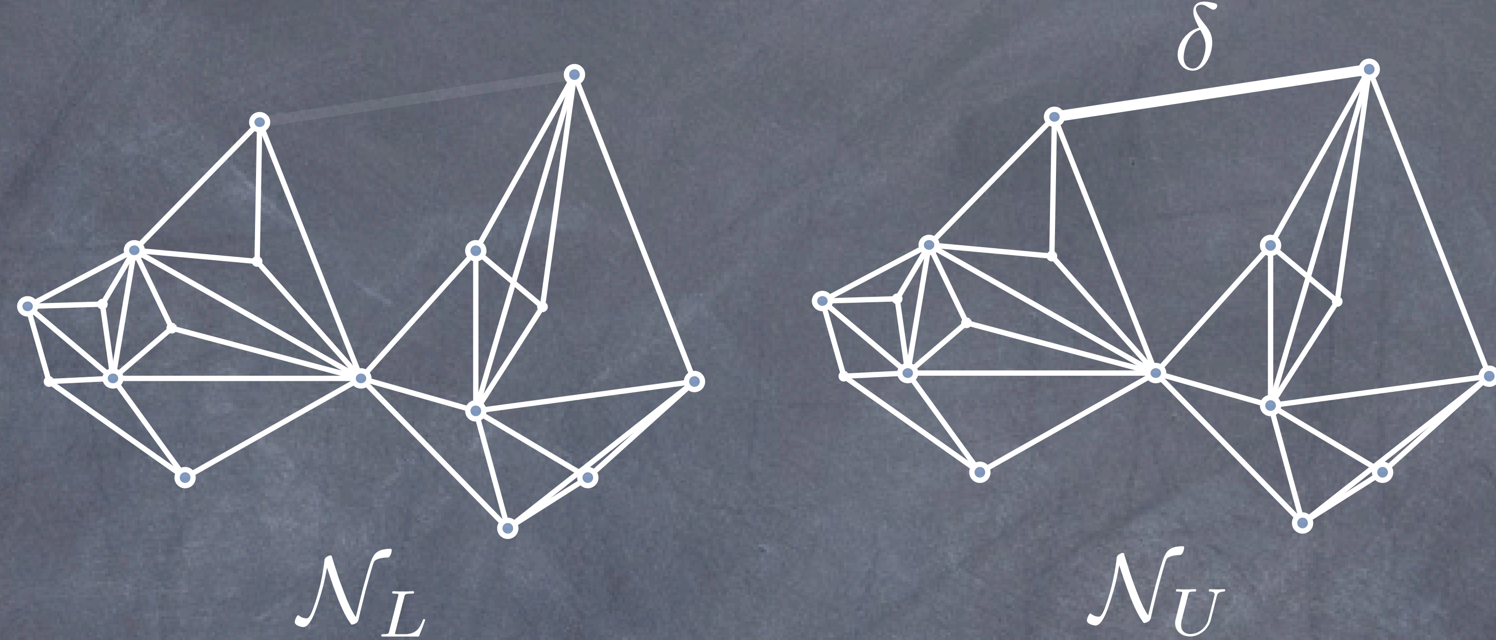
[Ho, Effros, Jalali 2010]



- For any R point in the capacity region of $\mathcal{N}_U$, there exists a corresponding sequence of codes
- The code of blocklength n carries $2^{n\delta}$ values on edge e, at least one of which is used for at least $2^{n(R-\delta)}$ input sequences
- Run the code for $\mathcal{N}_U$ on $\mathcal{N}_L$ by fixing that value on edge e and only transmitting those $2^{n(R-\delta)}$ input sequences

Networks with Bottlenecks

[Jalali, Effros, Ho 2011]



- As in the previous example, we use a code for $\mathcal{N}_U$ to build a code for $\mathcal{N}_L$
- A code at rate R for $\mathcal{N}_U$ is used to build
  - MA code to node v
  - BC code from node v
- Together give a code at rate $R - \delta$ for $\mathcal{N}_L$