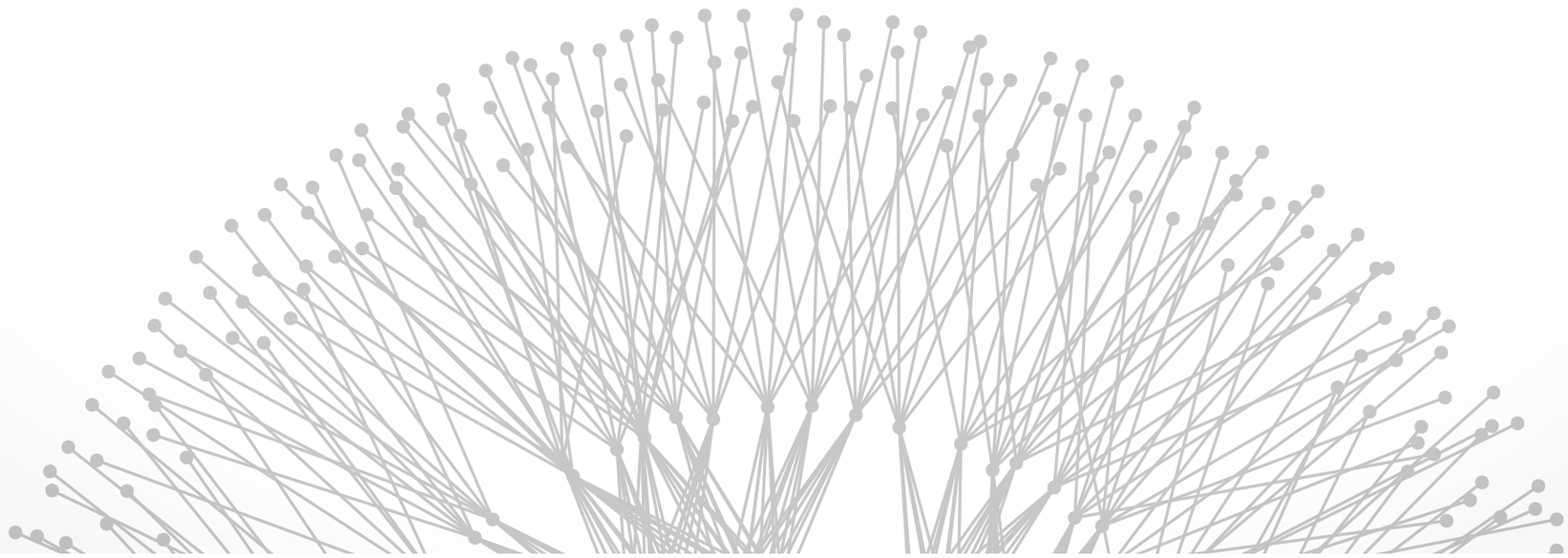
High Throughput Data Center Topology Design

Ankit Singla, P. Brighten Godfrey, Alexandra Kolla



Sardona
Burnomilia
Burno
Bromona
ad pretorum
Castello firmani
Cupra Maritima
Castro novo
Pinna
Salinas
Hadria
Ostia Aterni
Pinnum
Priferno
Firmo viceno
Interocrio
Reate
ad novas
Creto
Fidenis
Nomento
Tibori
Preneste
Tabios
Cabios
Carsulis
Marrubio
Corfinio
Sulmone
Ostia Aterni
In monte grani
Inmonte carbonario
ad quintanas
ad statuas
ad pactas
ad birium
Compito Anagnino
Ferentinum
Bovellas
Aricia
Sub lanubio
Tres tabernas
Antium
Astura
Clostris
ad turres
Hostis
Laurento
Lavinium
Via Appia
Via Latina
Via Lavicana
Via Prenestina
Via Tiburtina
Via Salaria
Via Aurelia
Via Clodia
Via Hostiensis
ad sextum
ad rubras
aqua viva
Careias
Turres
Pyrgos
Punicum
Alsium
Bebiana
Lorio
ad Scm. Petrum
Roma
Maxula
Adaquas
ad mercurium
Carthagine colon.
Thuni
Mediocera
Aggersel
Ulisippira
Missua
Thurarla
Tuburbomaius
Onellana
Bibae
Manange
Silesua
Thasarte
thuburos
Gallum Gallinatium
Thuburbominus
Clucar
Chisiduo
Vallis
Siguibba
Aulodes
Chipea

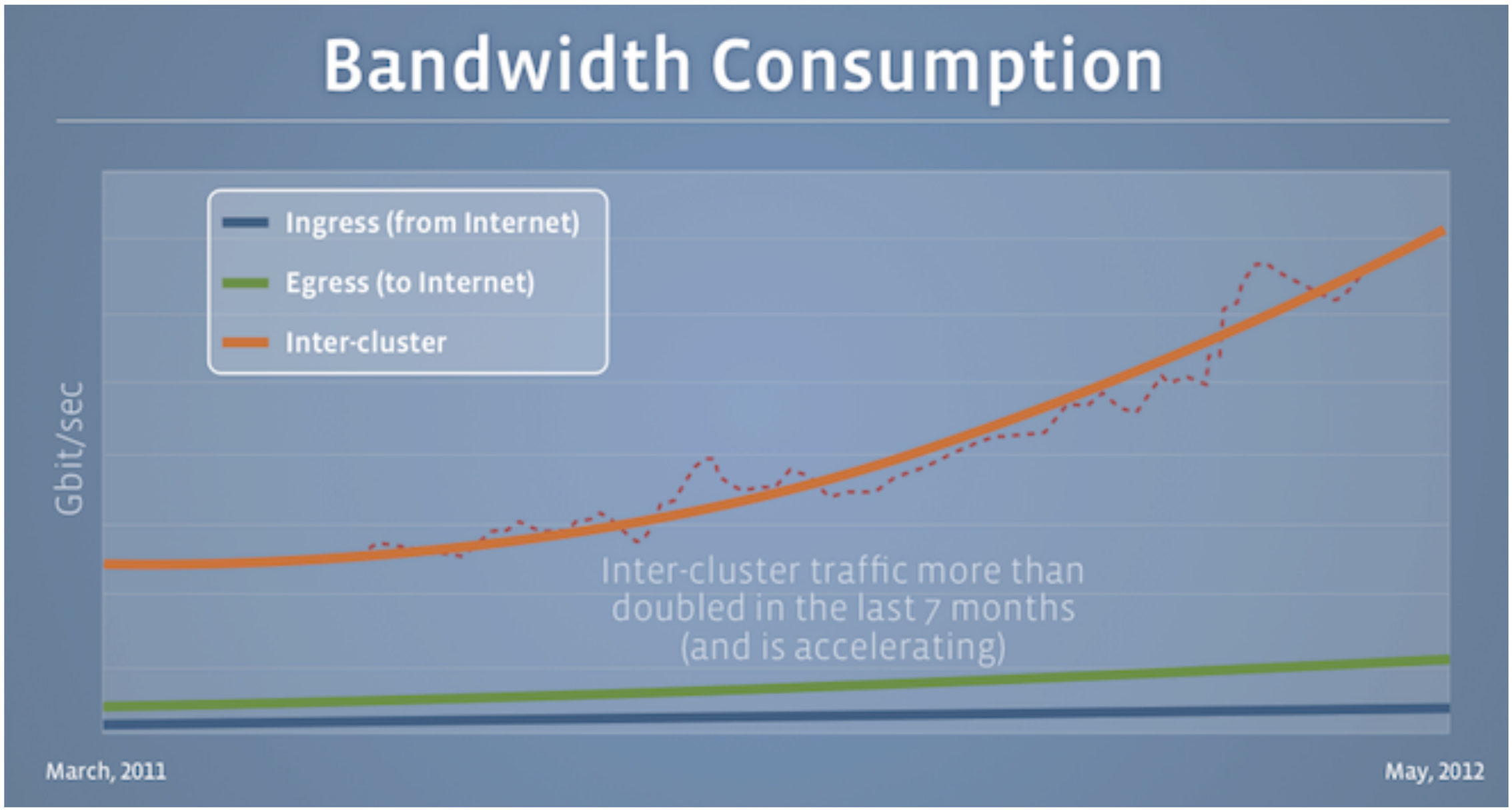

Gt Britain
Holland
Germany
Russia
France
Austria
Hungary
Turkey

"How long must we wait until our pigeon system rivals those of the Continental Powers?"

- The Nineteenth Century, 1899

Google | google.com/datacenters

The need for throughput

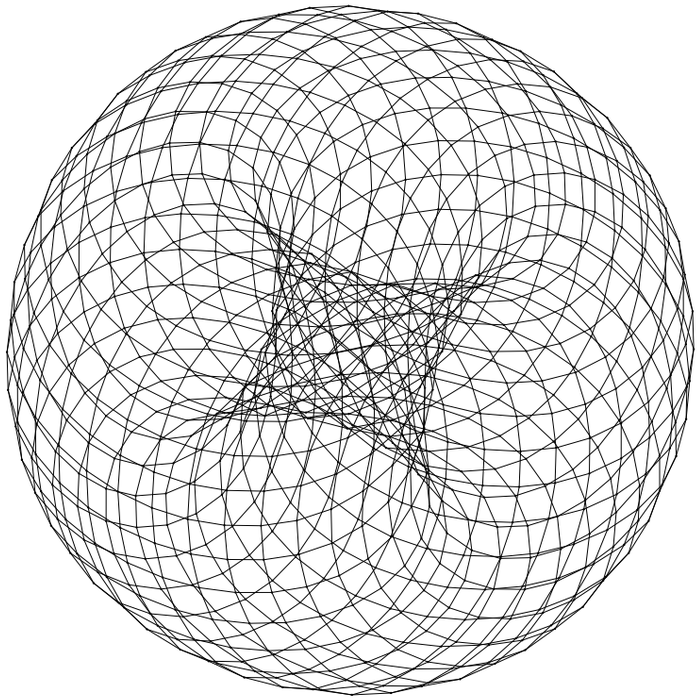


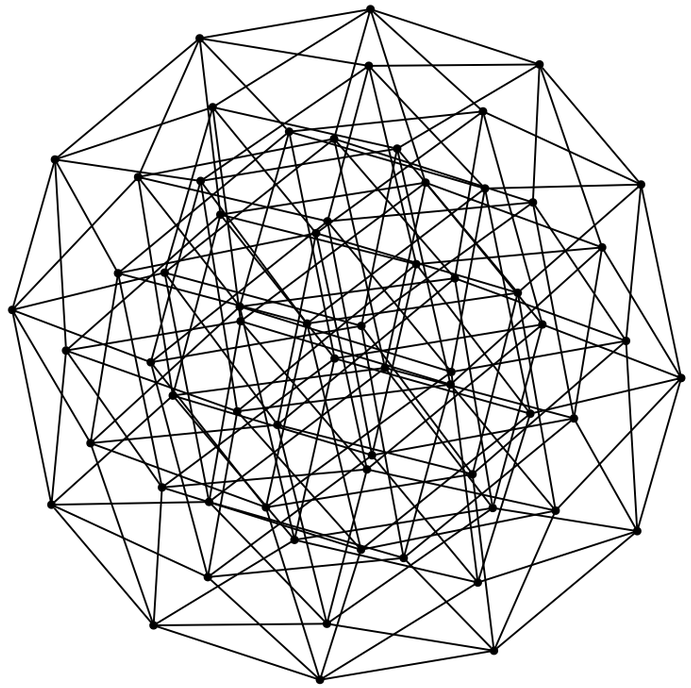
March
2011

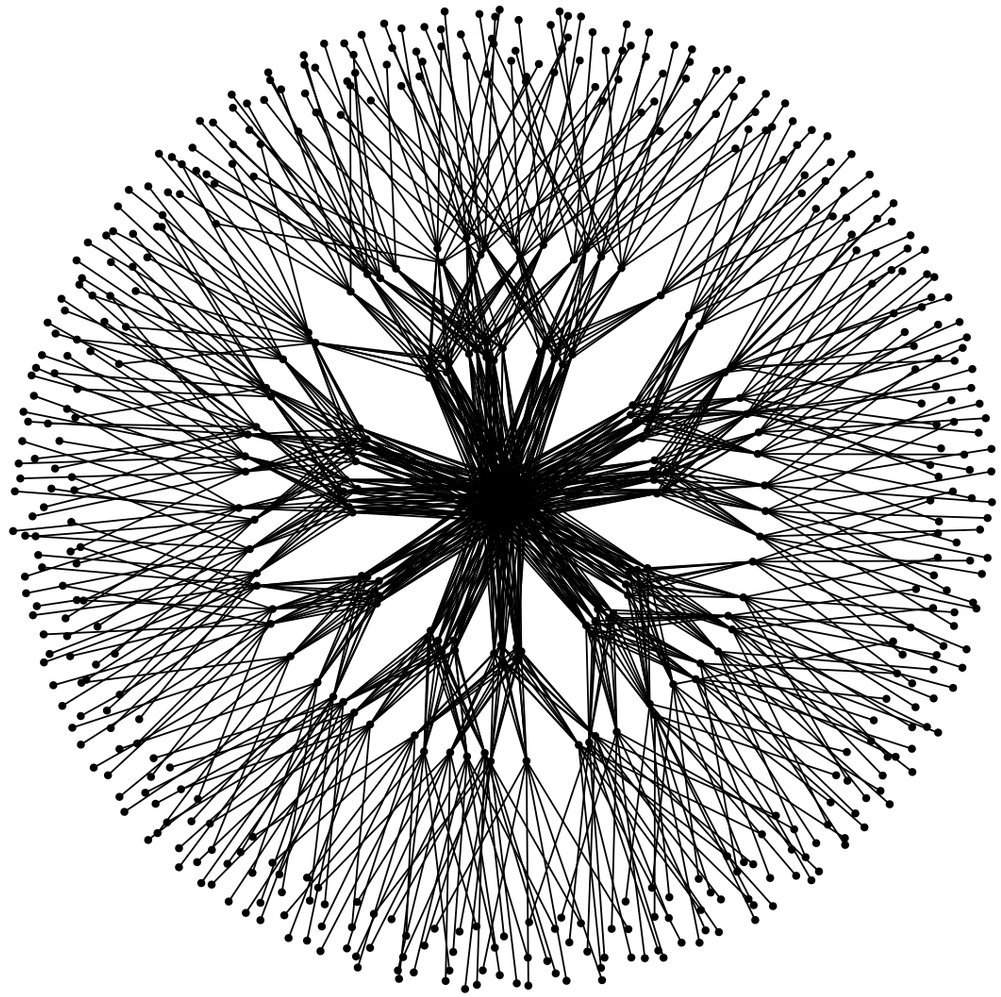
May
2012

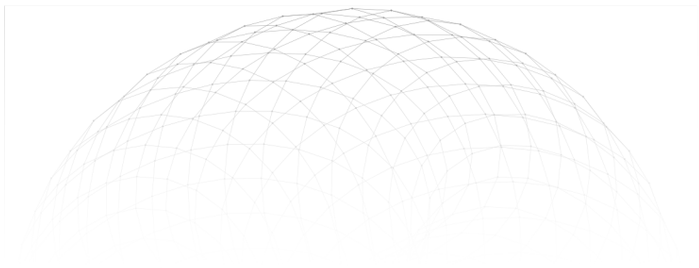
[Facebook, via Wired]

Many topology options …









How do we design throughput optimal network topologies?

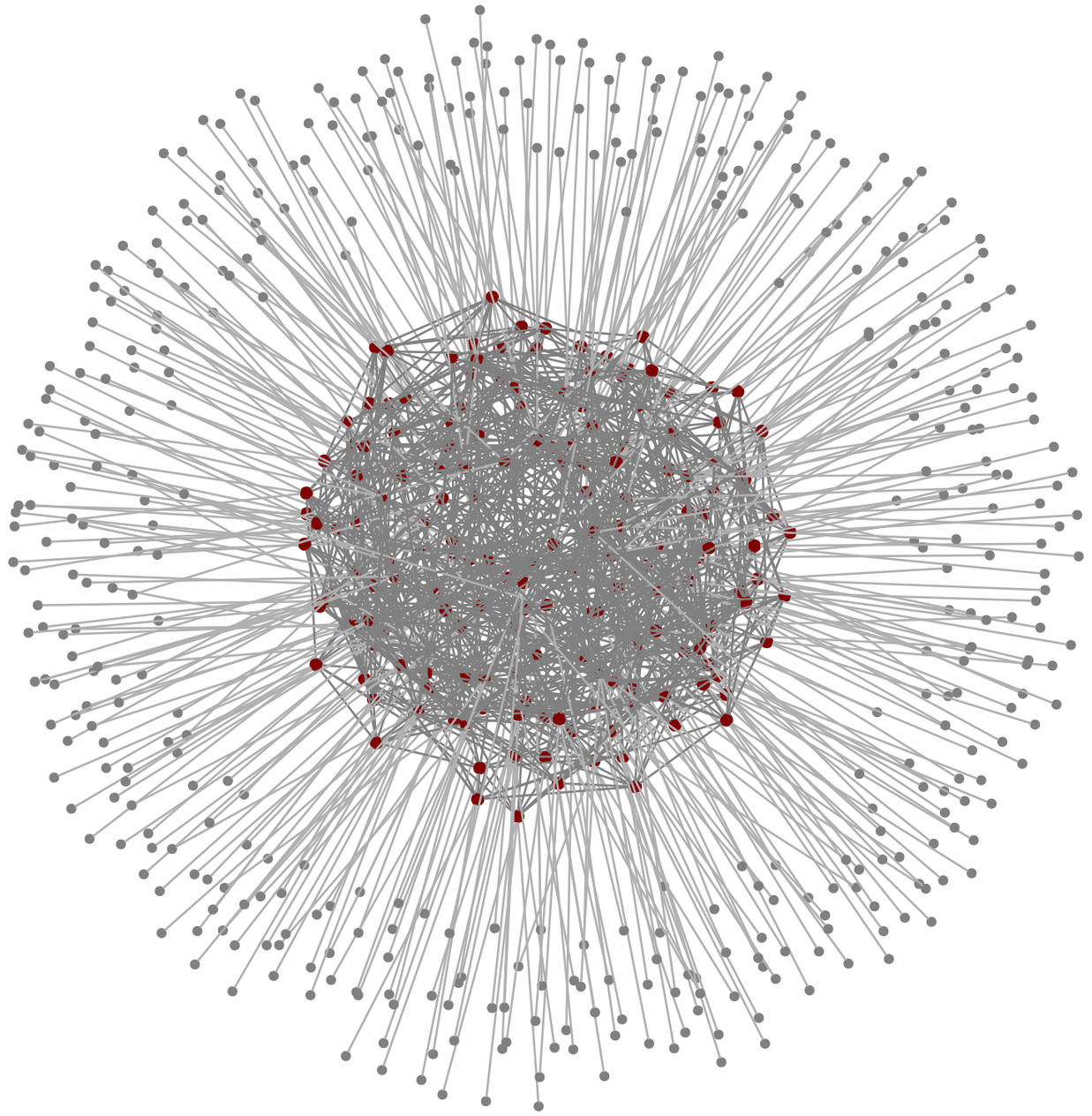
How do we design throughput optimal network topologies?

1 How close can we get to optimal network capacity?

1. How close can we get to optimal network capacity?

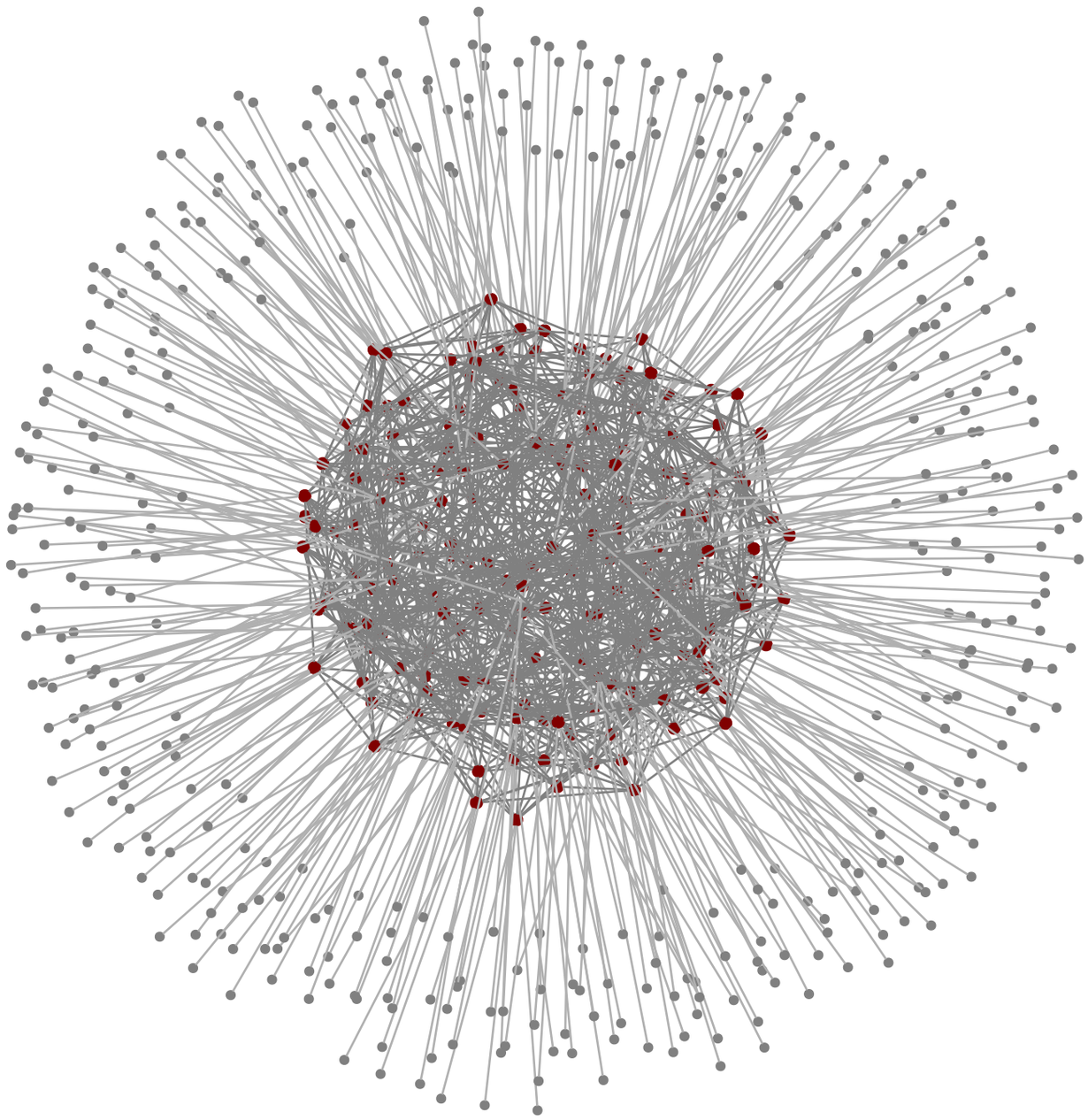
2. How do we handle heterogeneity?

Jellyfish: Networking Data Centers Randomly



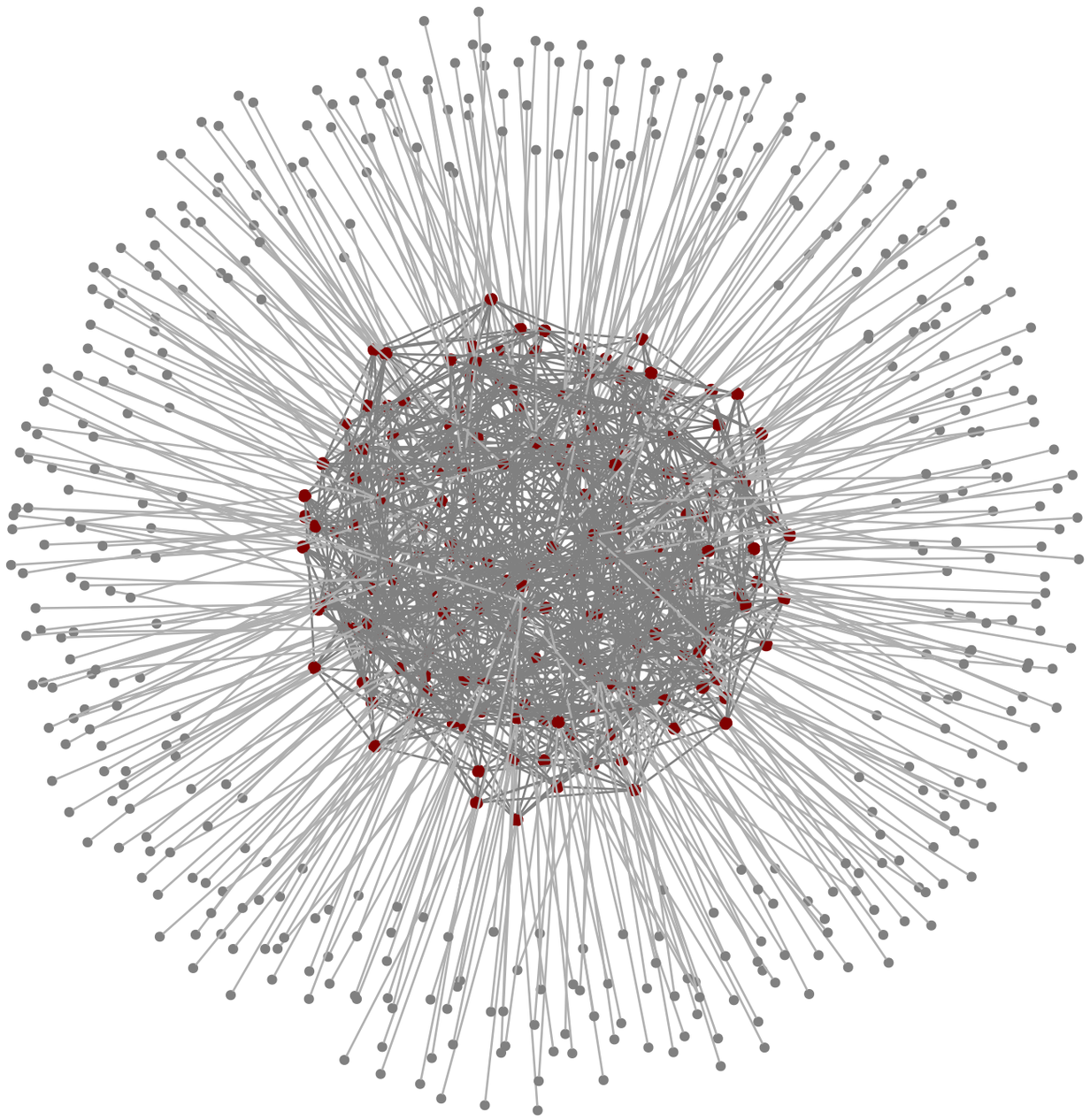
[NSDI 2012: Singla, Hong, Popa, Godfrey]

Jellyfish: Networking Data Centers Randomly



[NSDI 2012: Singla, Hong, Popa, Godfrey]

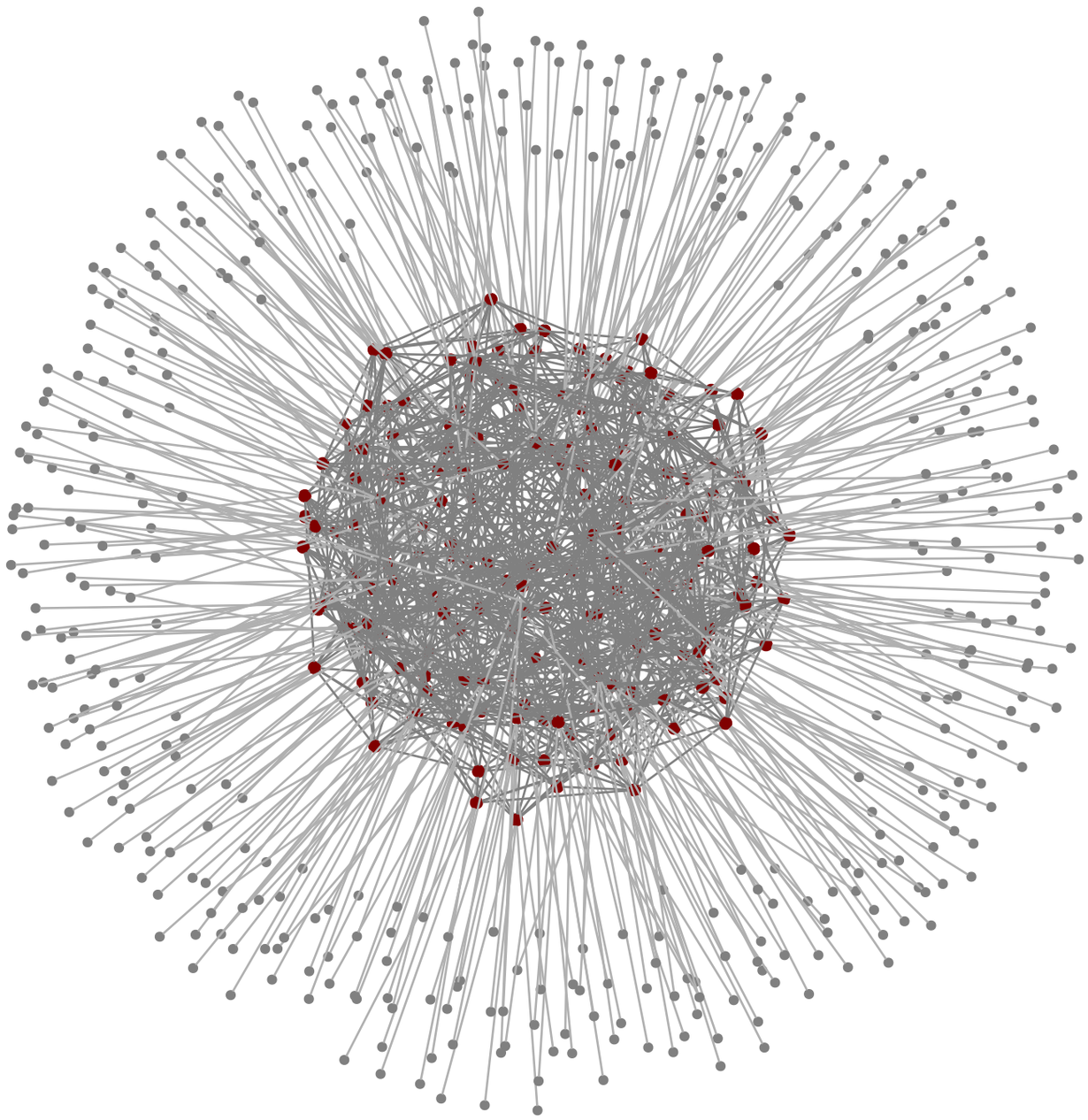
Jellyfish: Networking Data Centers Randomly



- High capacity
  - Beat fat-trees by 25%+

[NSDI 2012: Singla, Hong, Popa, Godfrey]

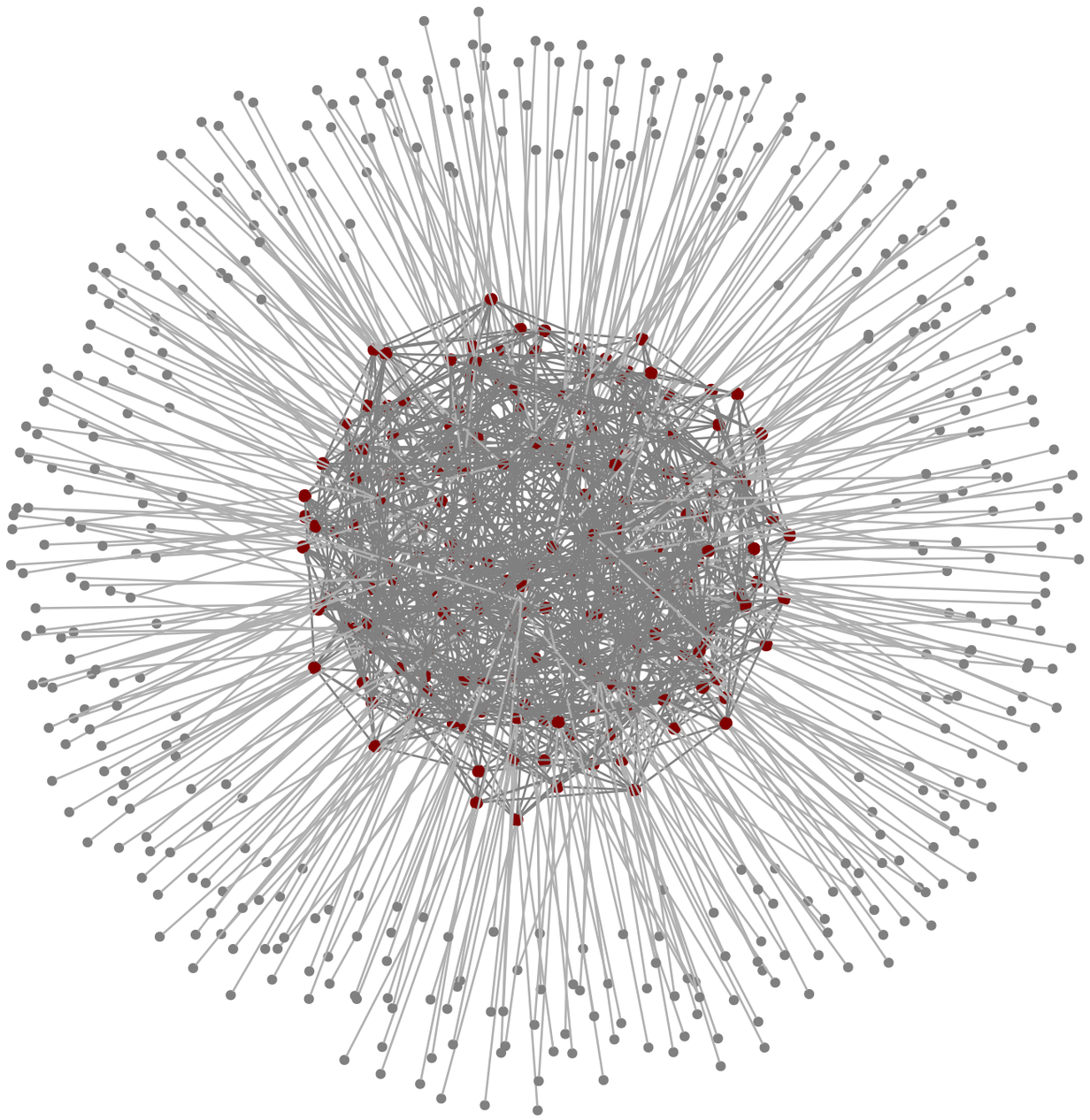
Jellyfish: Networking Data Centers Randomly



- High capacity
  - Beat fat-trees by 25%+
- Easier to expand
  - 60% cheaper expansion

[NSDI 2012: Singla, Hong, Popa, Godfrey]

Jellyfish: Networking Data Centers Randomly



- High capacity
  - Beat fat-trees by 25%+
- Easier to expand
  - 60% cheaper expansion
- Routing and cabling are solvable problems

[NSDI 2012: Singla, Hong, Popa, Godfrey]

1 How close can we get to optimal network capacity?

1. How close can we get to optimal network capacity?

2. How do we handle heterogeneity?

How do we measure throughput?

How do we measure throughput?

Maximize the minimum flow

How do we measure throughput?

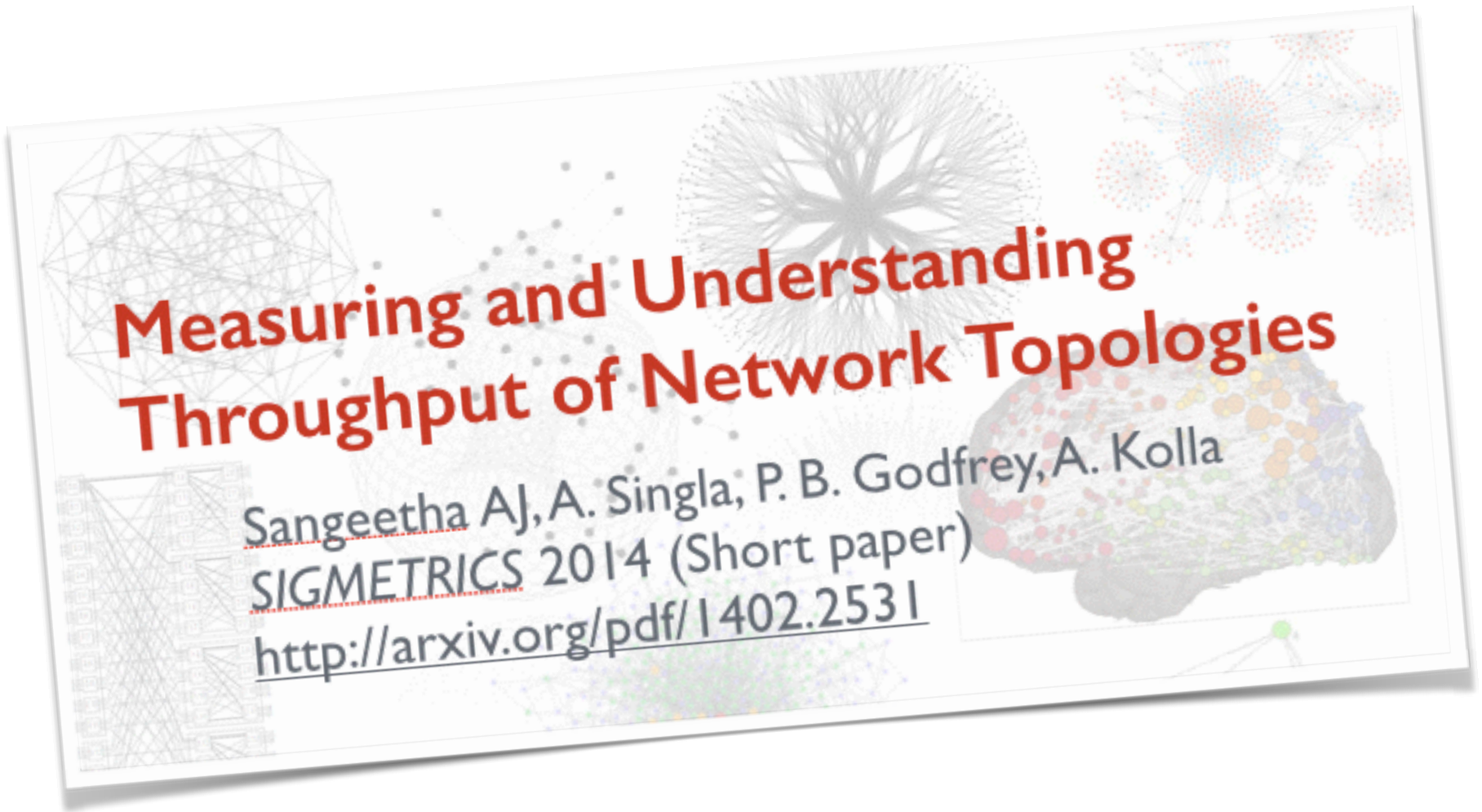
Maximize the minimum flow

under random permutation traffic

How do we measure throughput?

Maximize the minimum flow

under random permutation traffic



How do we measure throughput?

How do we measure throughput?

- Bisection bandwidth ≠ throughput

How do we measure throughput?

- Bisection bandwidth ≠ throughput
- Near-worst case traffic patterns

How close can we get to optimal network capacity?

A simple upper bound

A simple upper bound

flows

A simple upper bound

flows • capacity used per flow

A simple upper bound

flows • capacity used per flow

≤ total capacity

A simple upper bound

flows • capacity used per flow

≤ total capacity

A simple upper bound

flows • throughput per flow • mean path length

≤ total capacity

A simple upper bound

$$\text{throughput per flow} \leq \frac{\text{total capacity}}{\text{\# flows} \bullet \text{mean path length}}$$

A simple upper bound

$$\text{throughput per flow} \leq \frac{\sum_{\text{links}} \text{capacity}(\textit{link})}{\#\ \text{flows} \bullet \text{mean path length}}$$

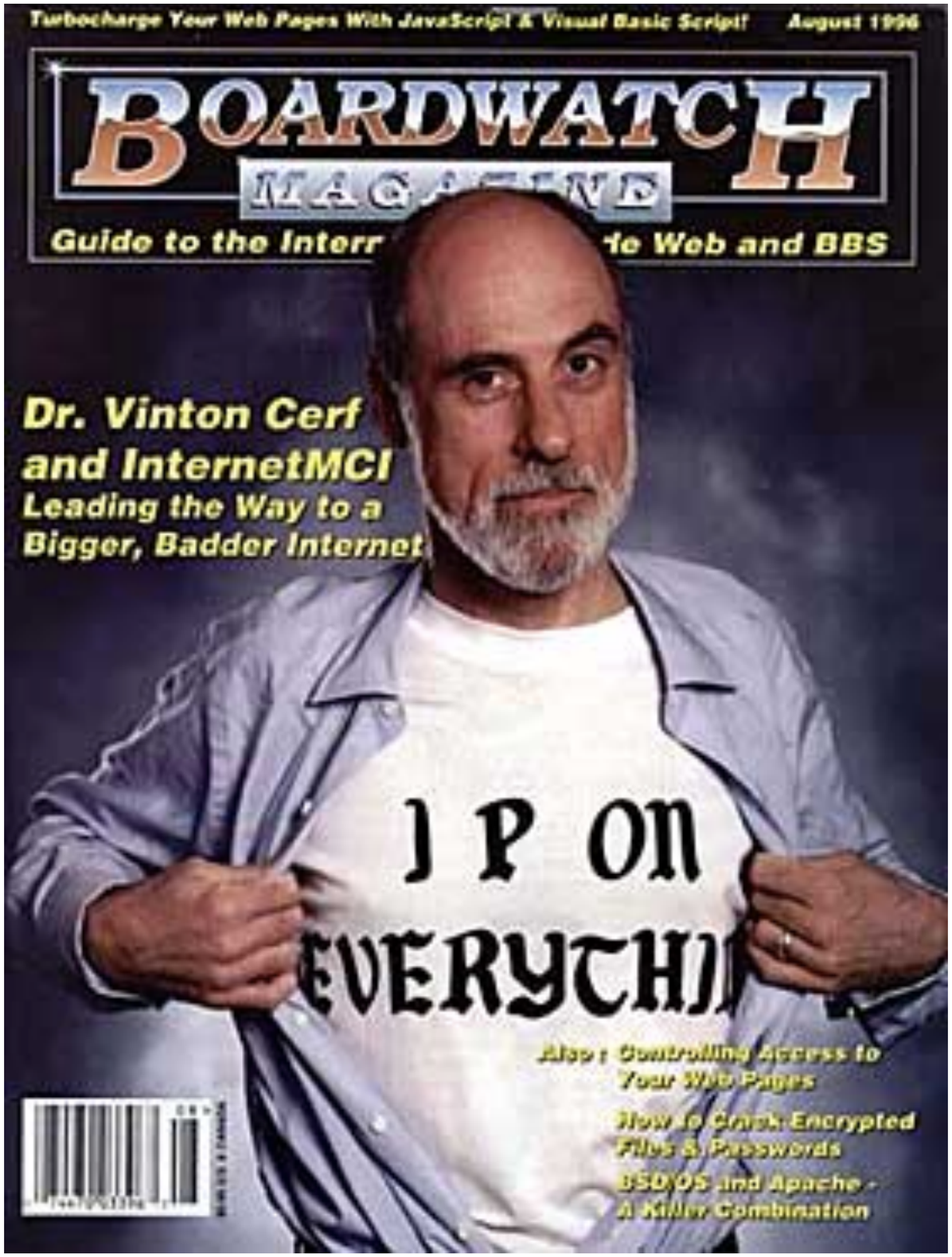
A simple upper bound

$$\text{throughput per flow} \leq \frac{\sum_{\text{links}} \text{capacity}(\textit{link})}{\#\ \text{flows} \bullet \text{mean path length}}$$

Lower bound this!

Lower bound on mean path length

Lower bound on mean path length



Lower bound on mean path length

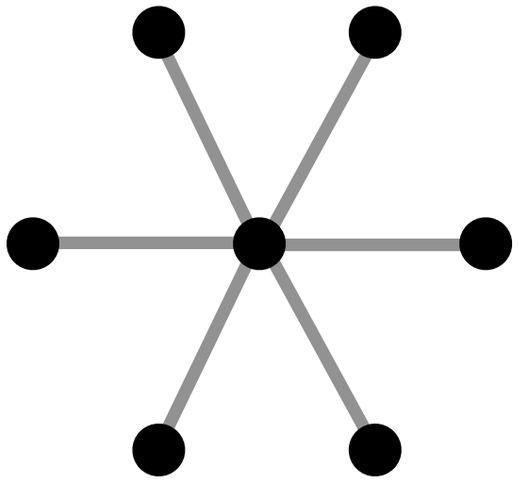
[Cerf et al., “A lower bound on the average shortest path length in regular graphs”, 1974]

Lower bound on mean path length

Distance	# Nodes

[Cerf et al., “A lower bound on the average shortest path length in regular graphs”, 1974]

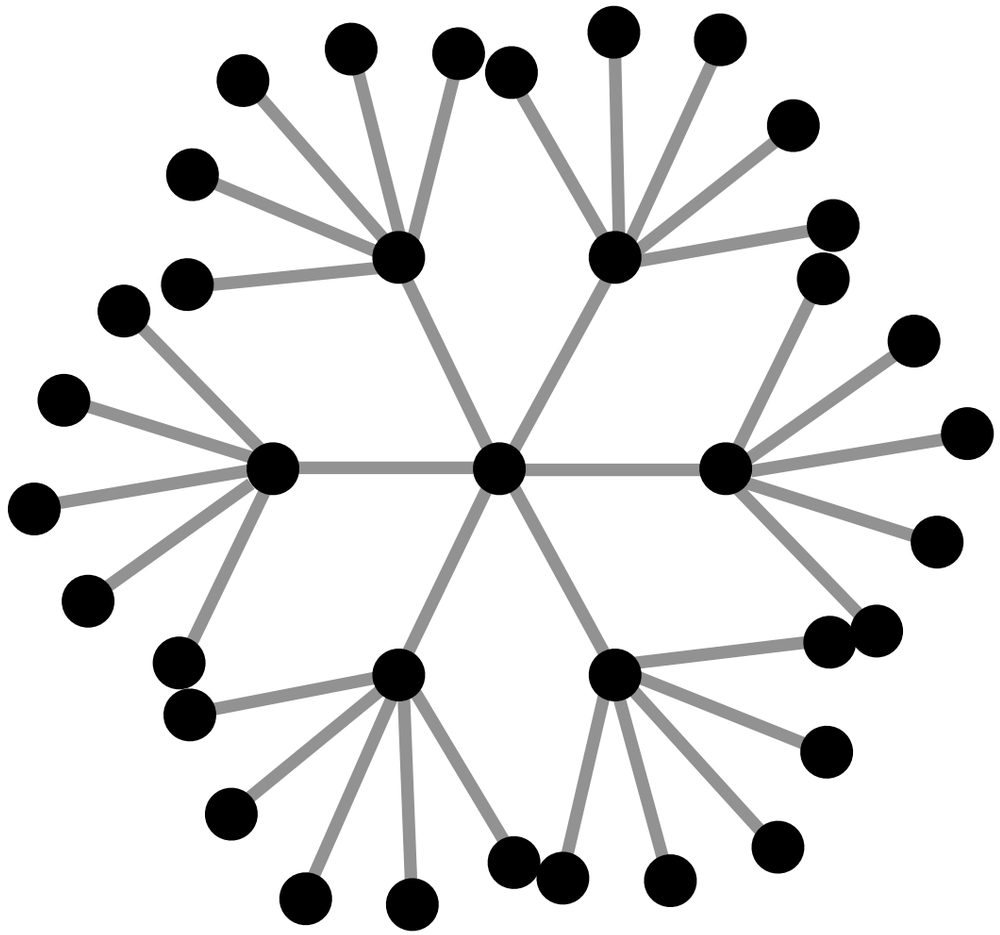
Lower bound on mean path length



Distance	# Nodes
1	6

[Cerf et al., “A lower bound on the average shortest path length in regular graphs”, 1974]

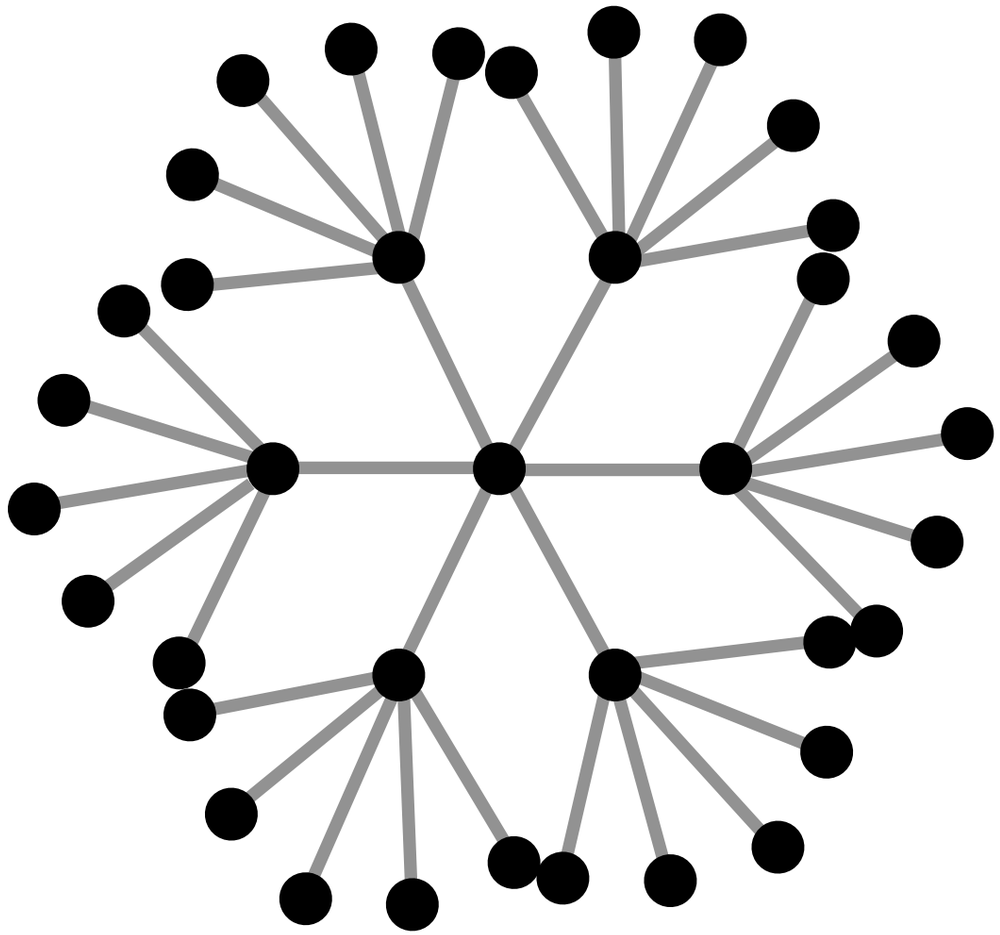
Lower bound on mean path length



Distance	# Nodes
1	6
2	6

[Cerf et al., “A lower bound on the average shortest path length in regular graphs”, 1974]

Lower bound on mean path length

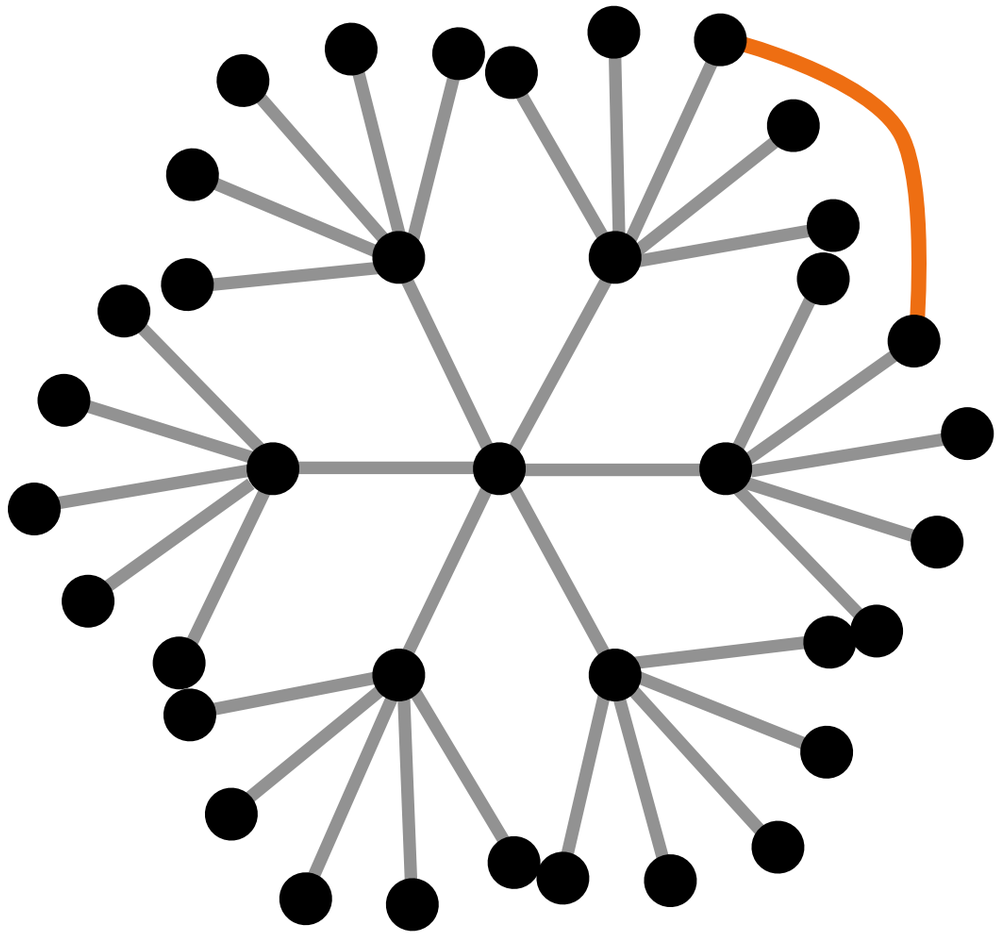


Distance	# Nodes
1	6
2	6

(Ugliness omitted)

[Cerf et al., “A lower bound on the average shortest path length in regular graphs”, 1974]

Lower bound on mean path length

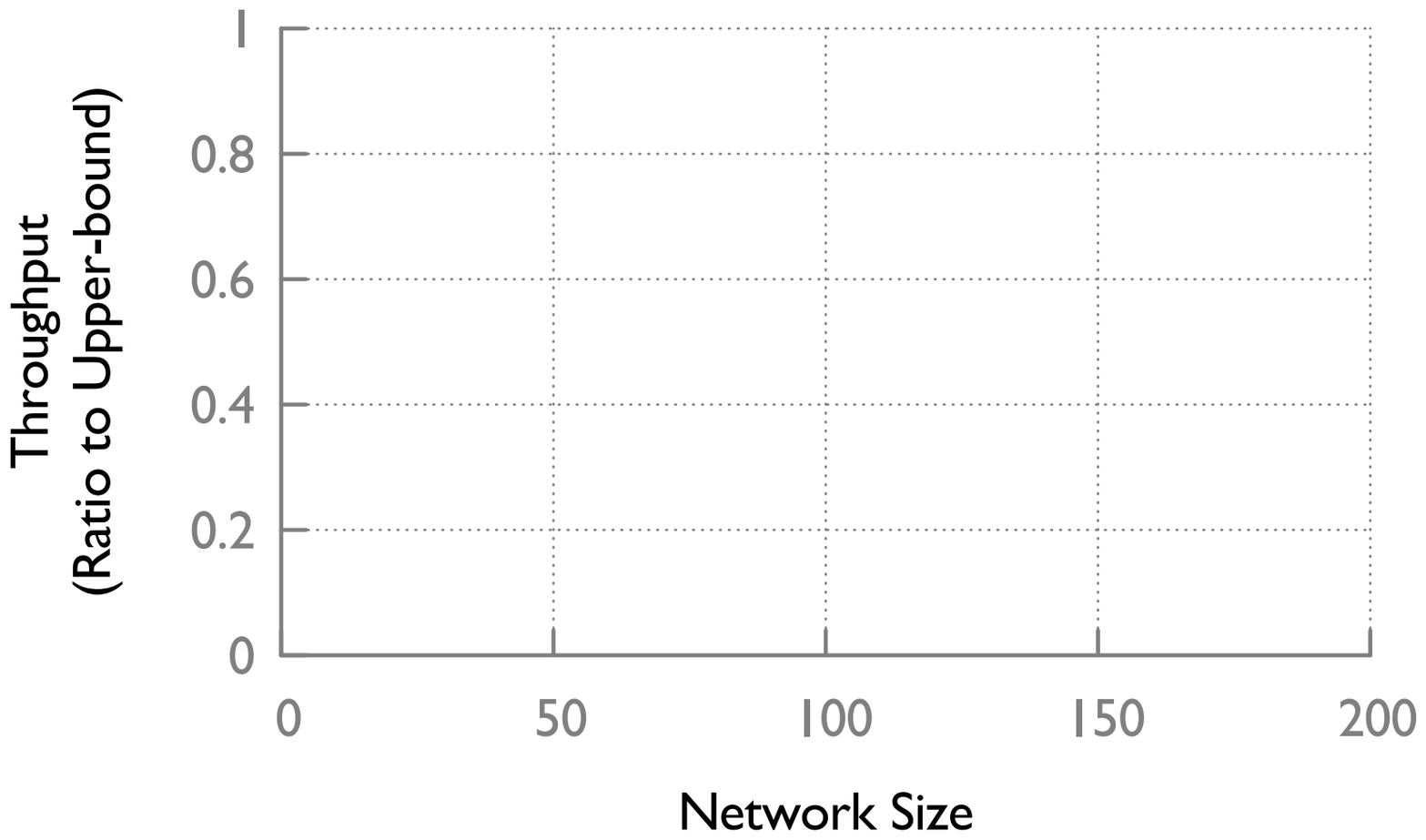


Distance	# Nodes
1	6
2	6

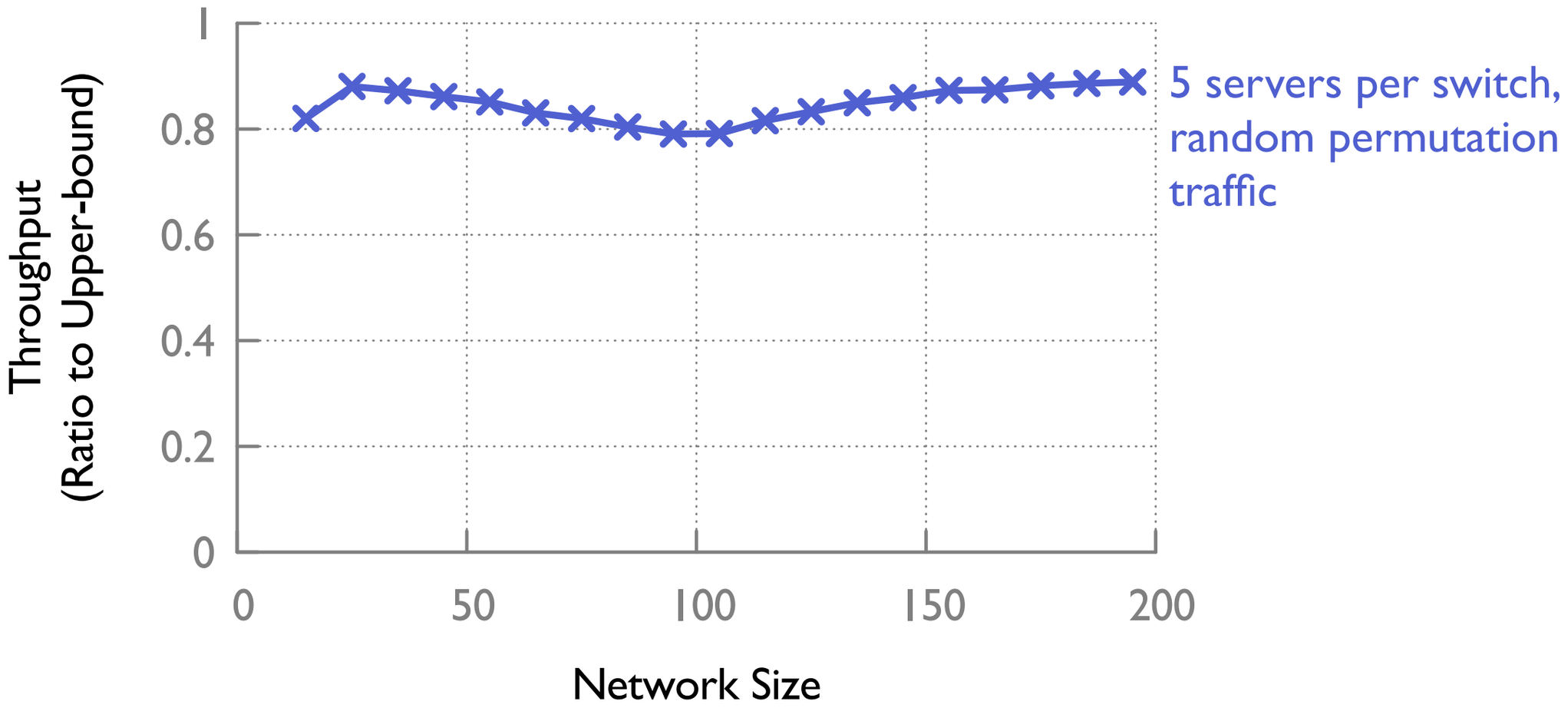
(Ugliness omitted)

[Cerf et al., “A lower bound on the average shortest path length in regular graphs”, 1974]

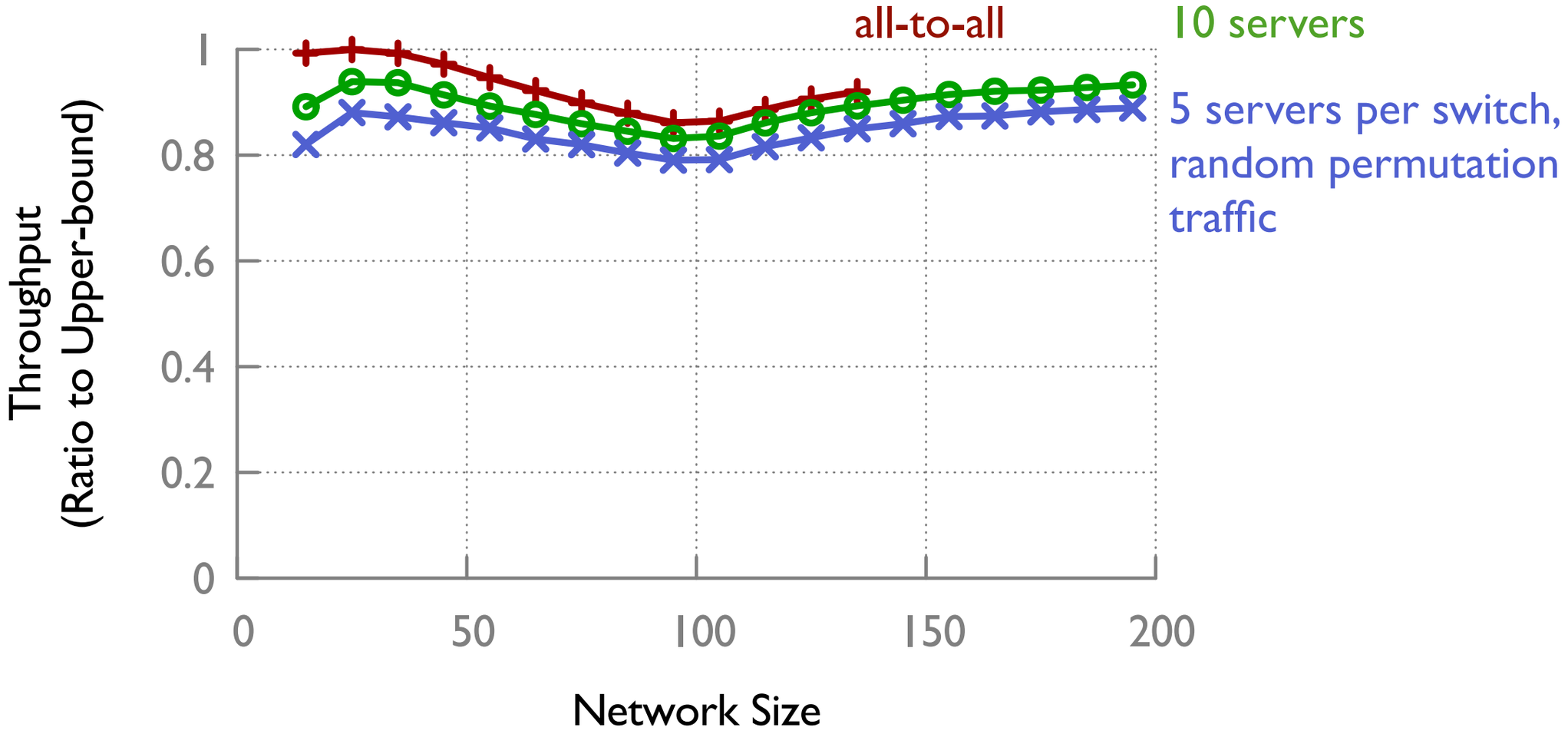
Random graphs vs. bound



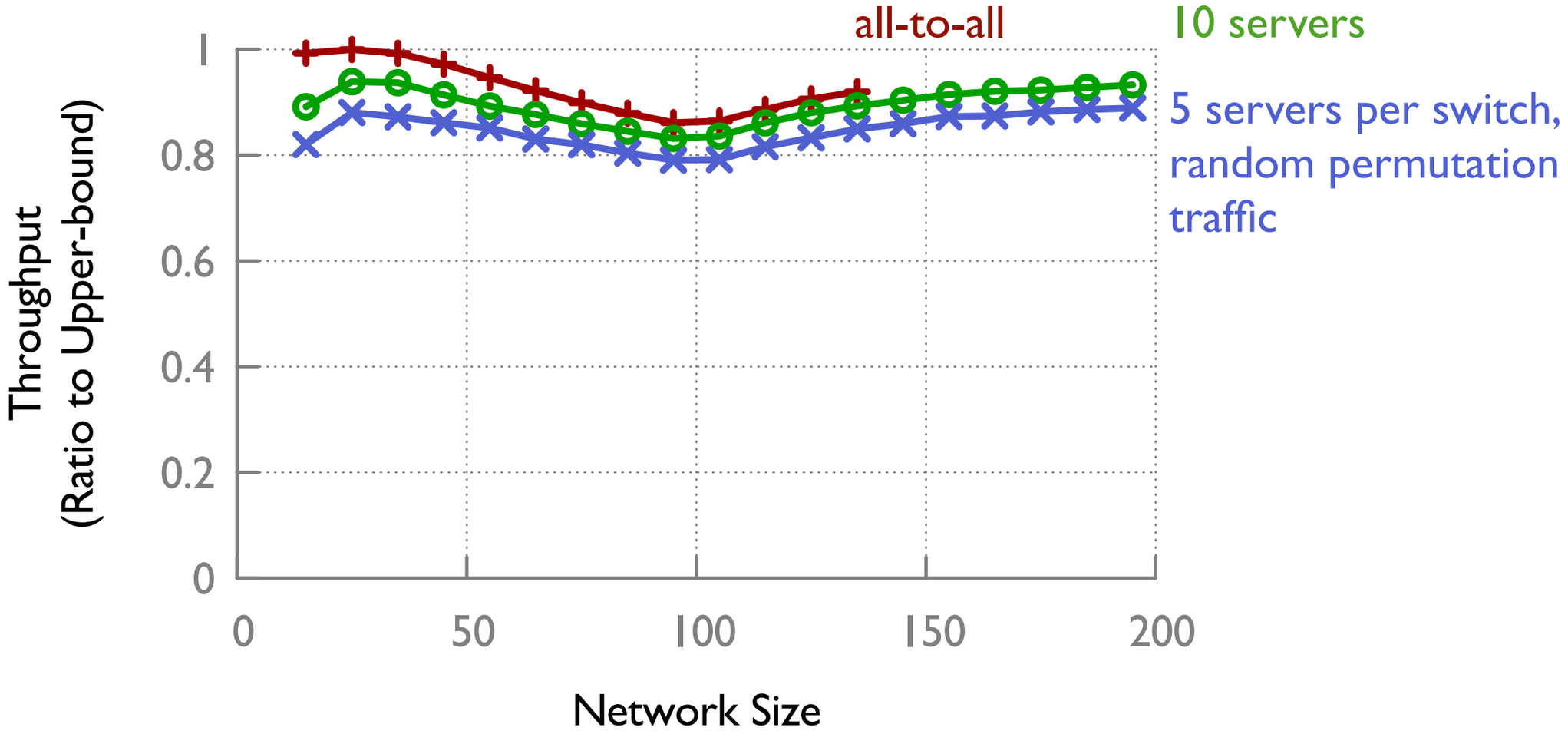
Random graphs vs. bound



Random graphs vs. bound

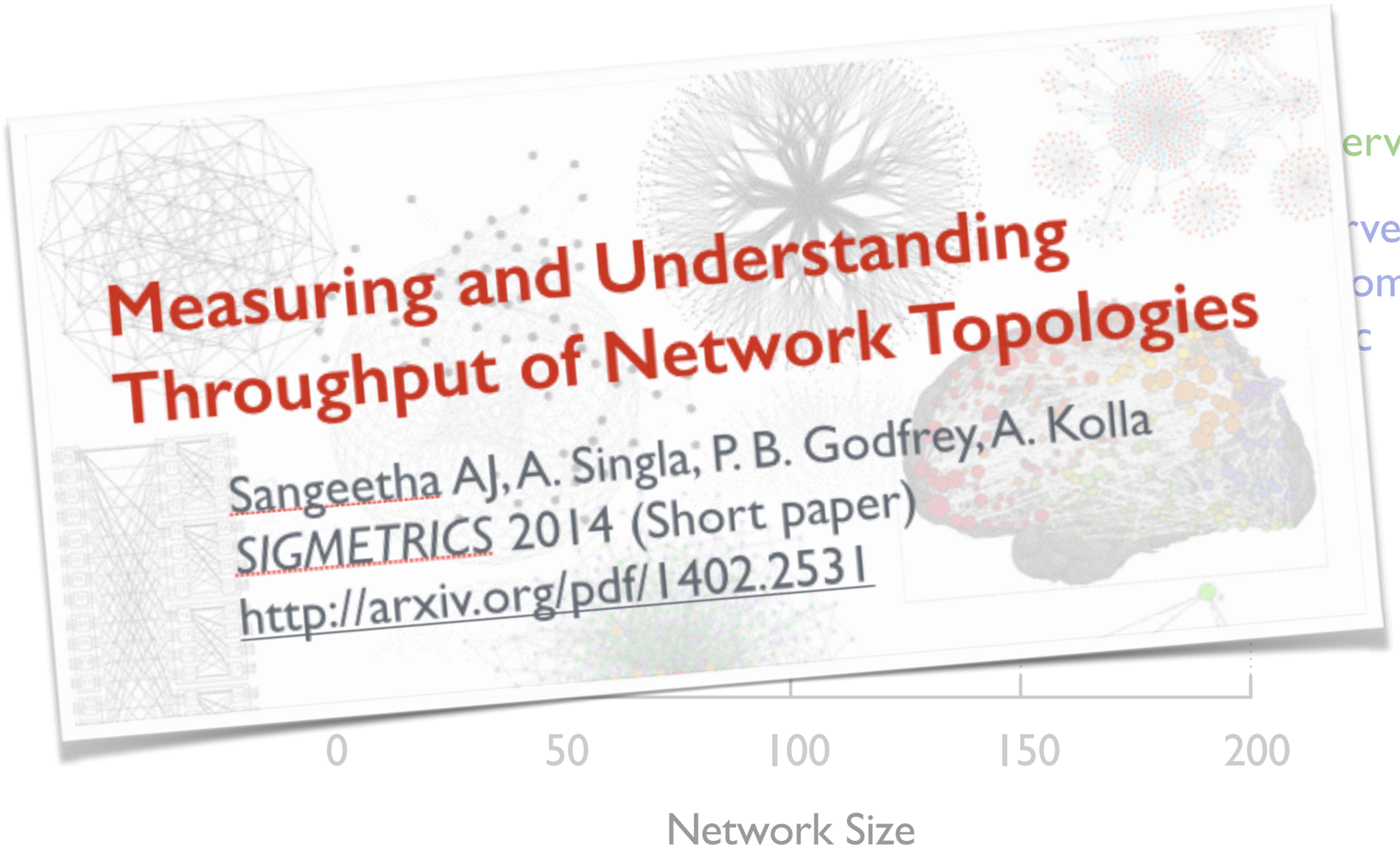


Random graphs vs. bound



Random graphs within a few percent of optimal!

Random graphs vs. bound



Random graphs within a few percent of optimal!

Random graphs exceed throughput of other topologies

How close can we get to
optimal network capacity?

Very close!!

How do we handle heterogeneity?

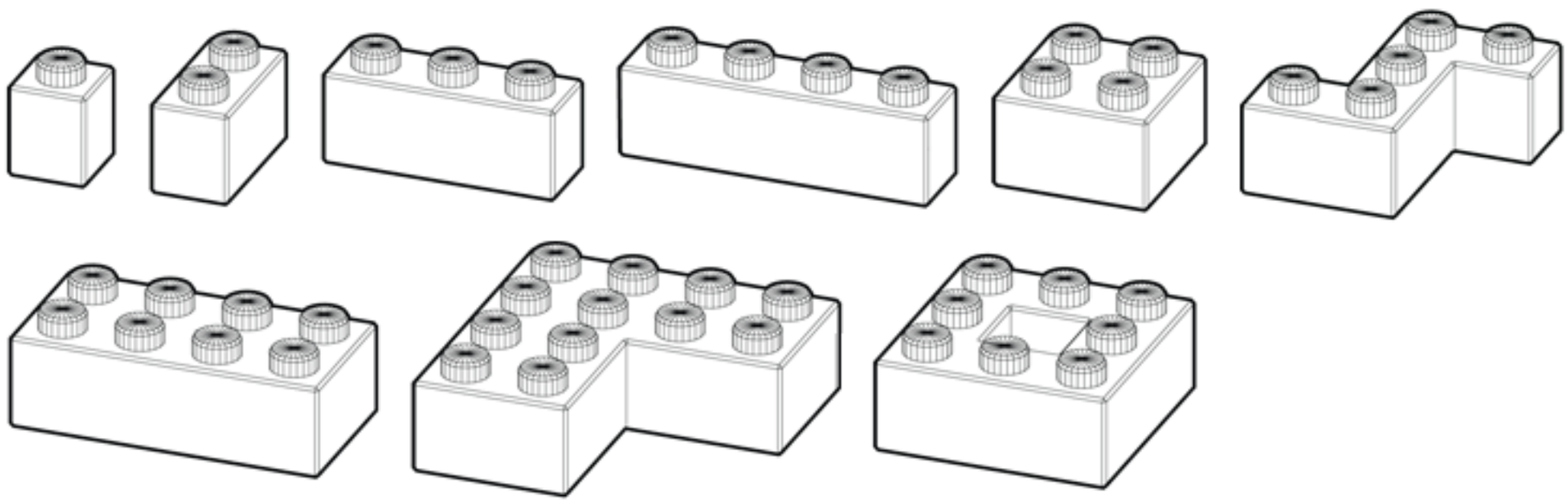
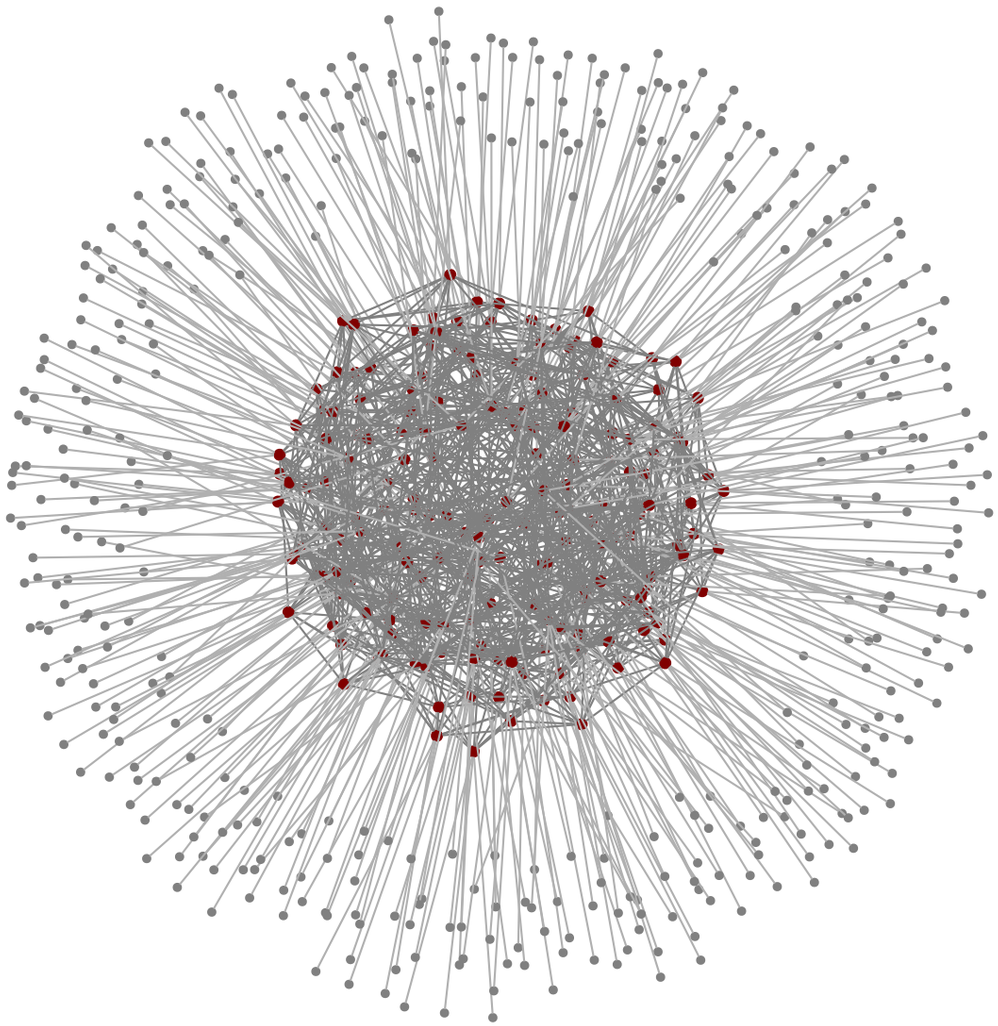
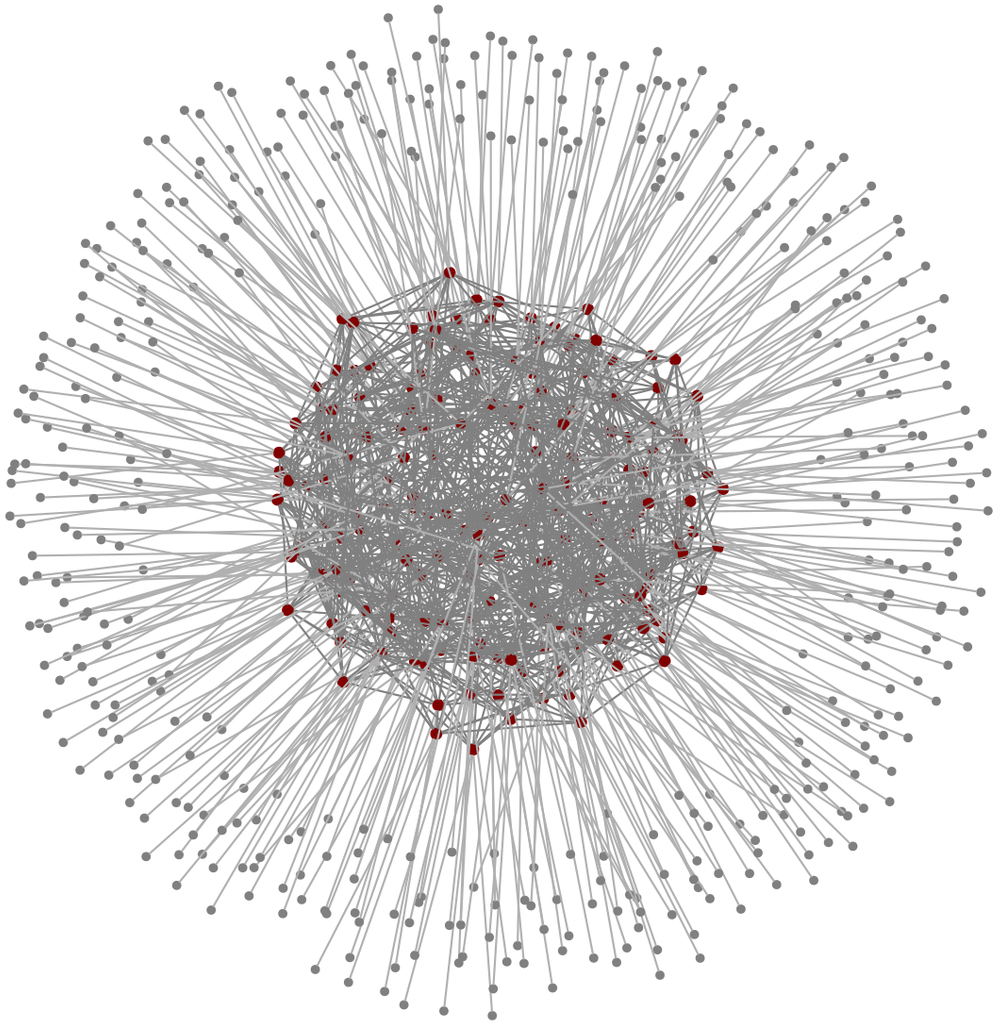


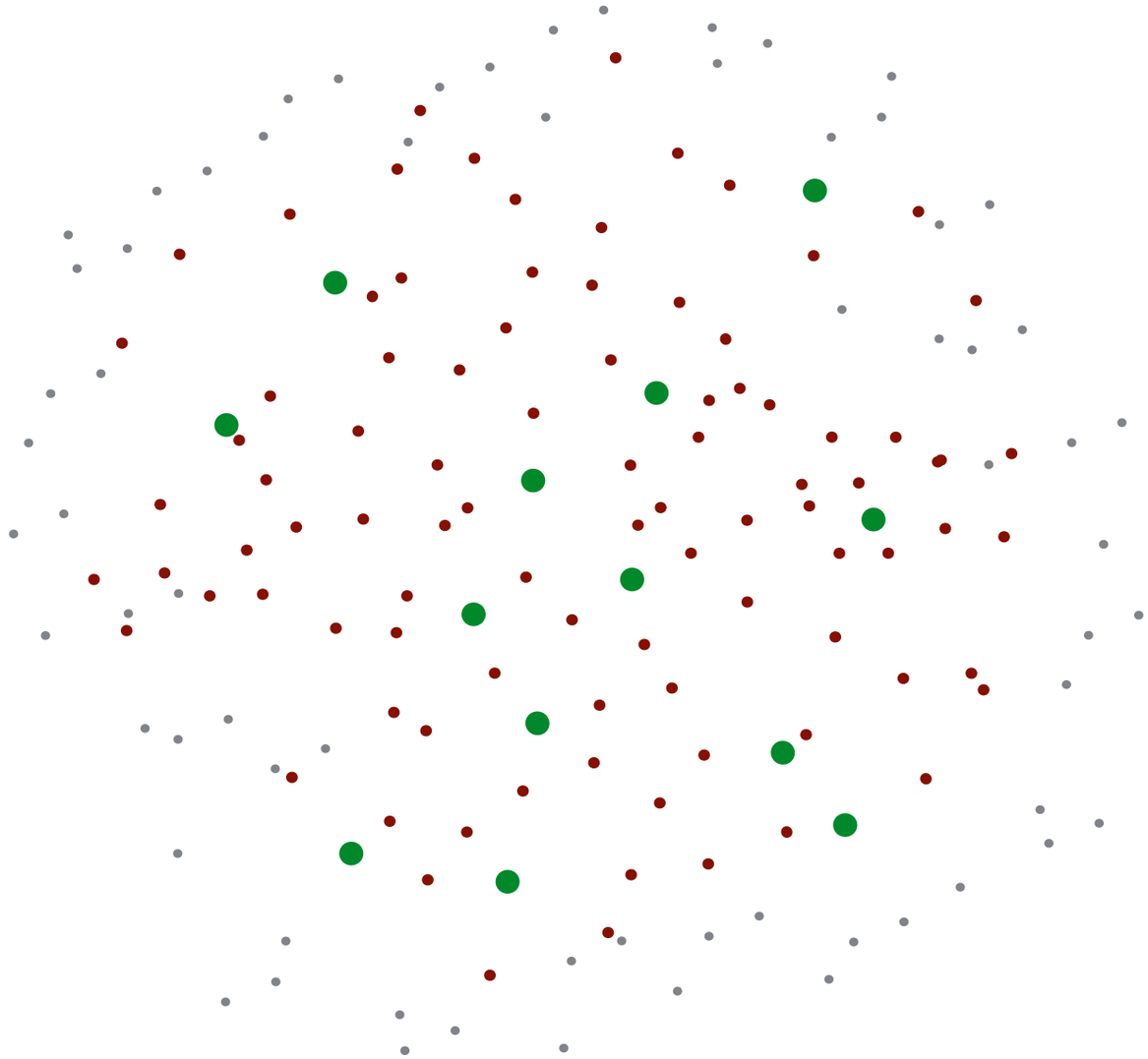
Image credit: Legolizer (www.drububu.com)

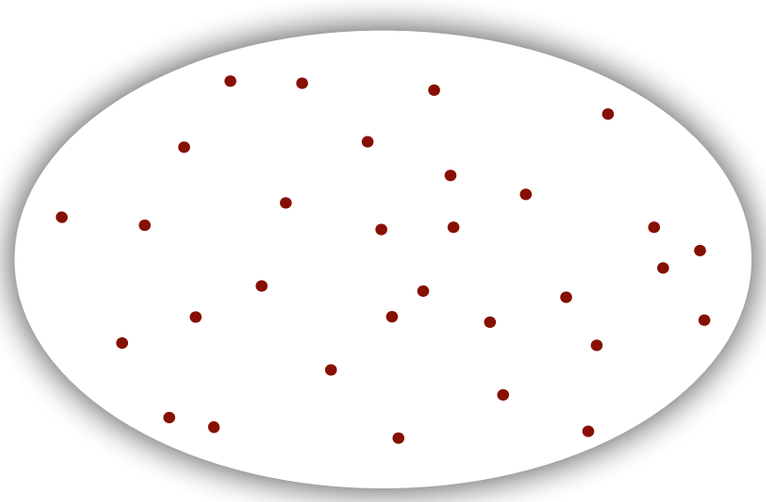
Heterogeneity

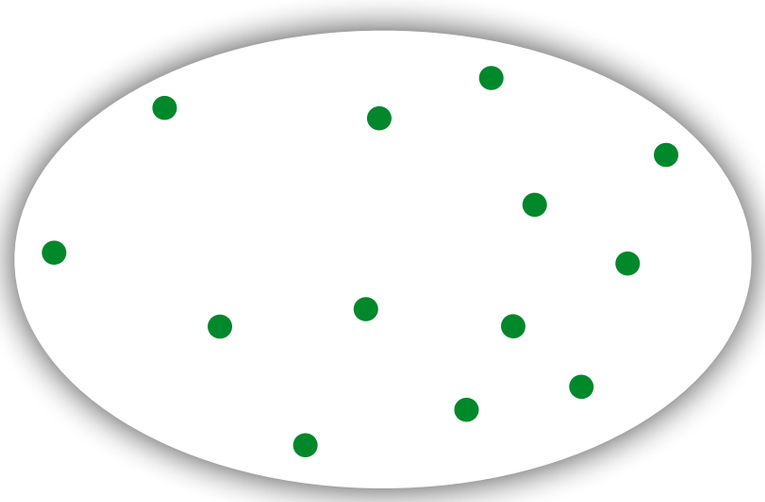


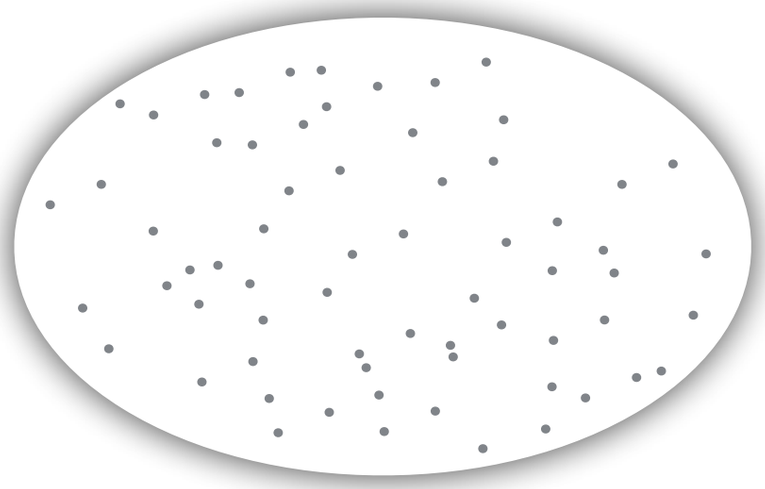
Heterogeneity

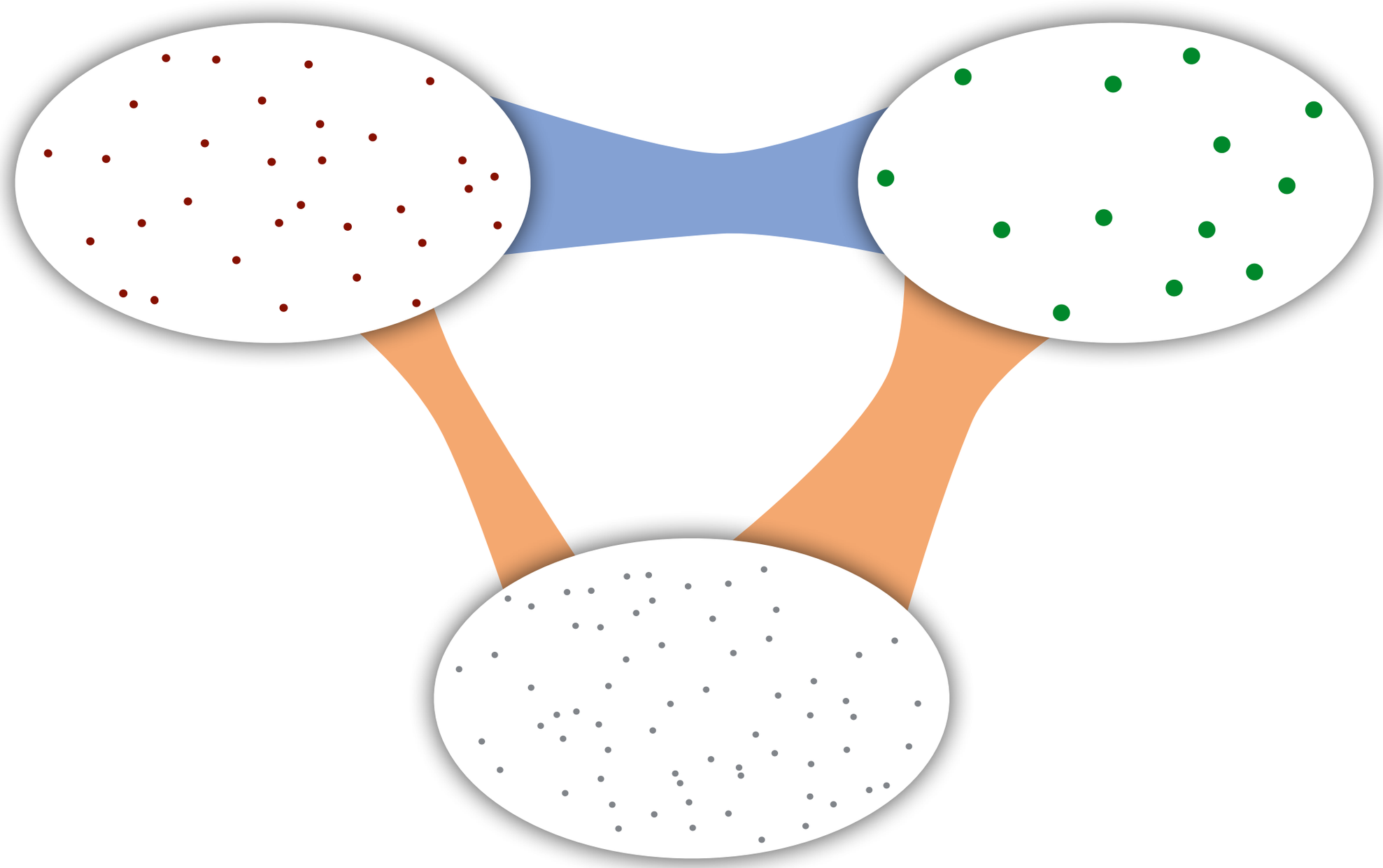




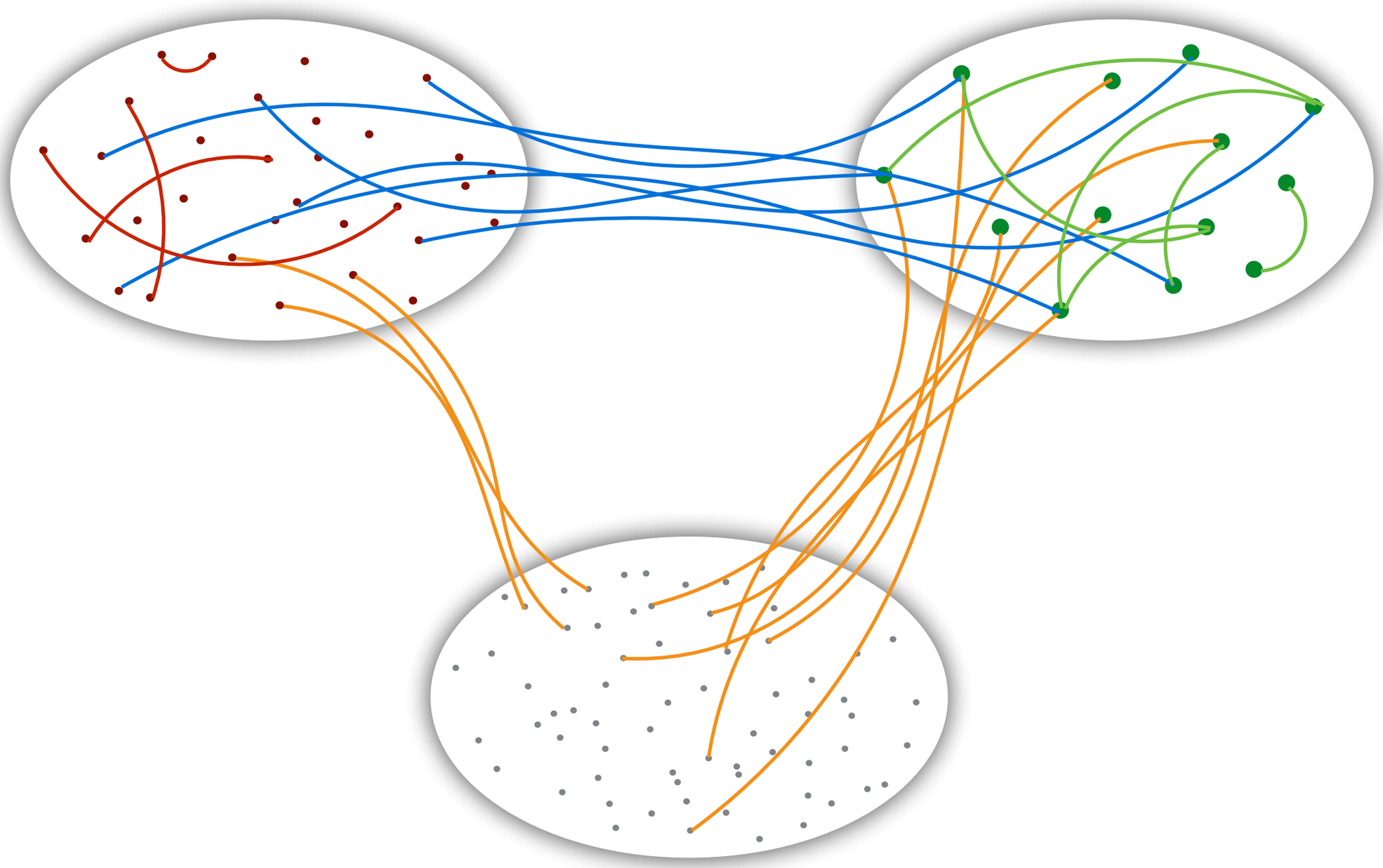








Random graphs as a building block



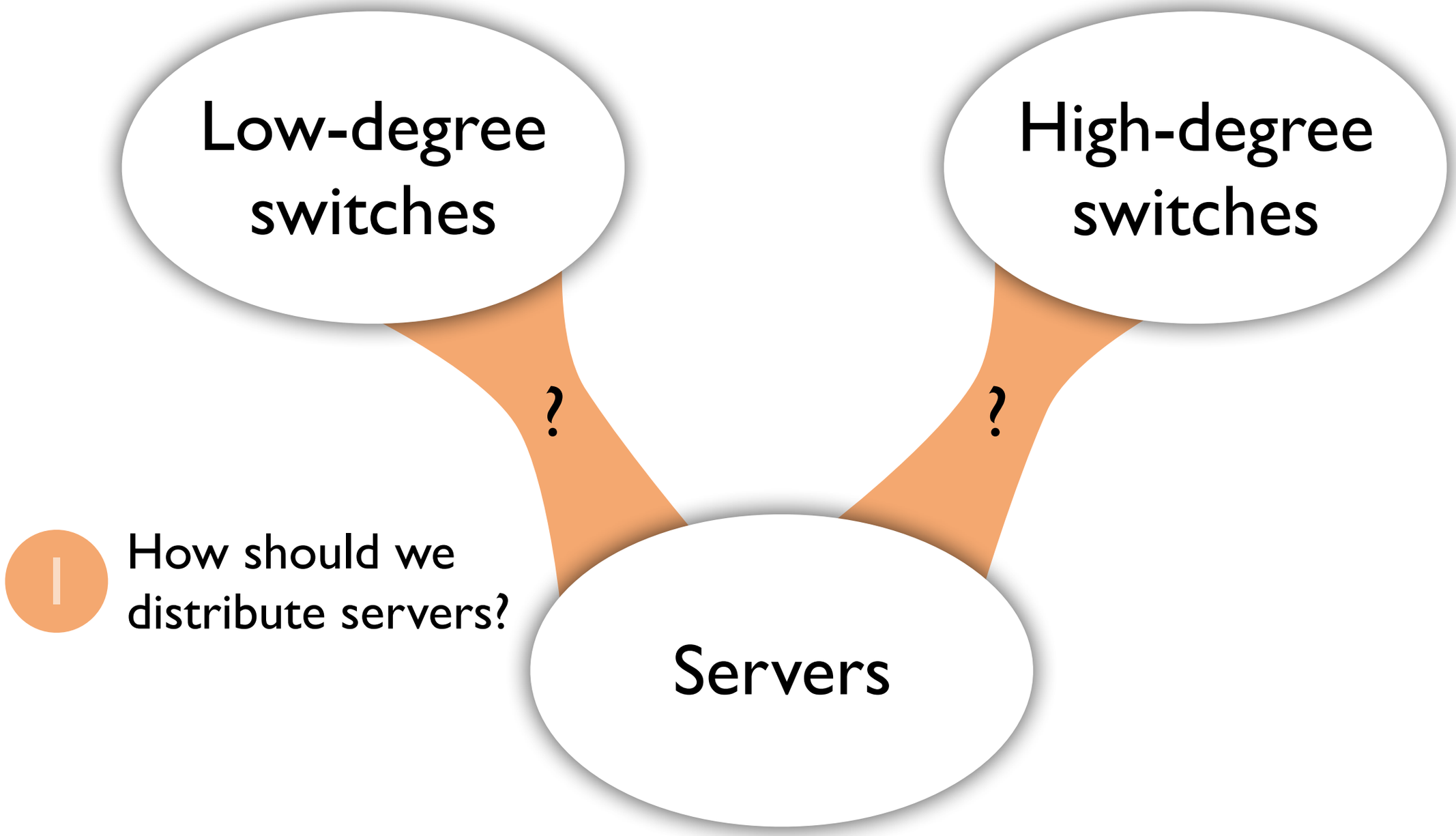
Random graphs as a building block

Low-degree switches

High-degree switches

Servers

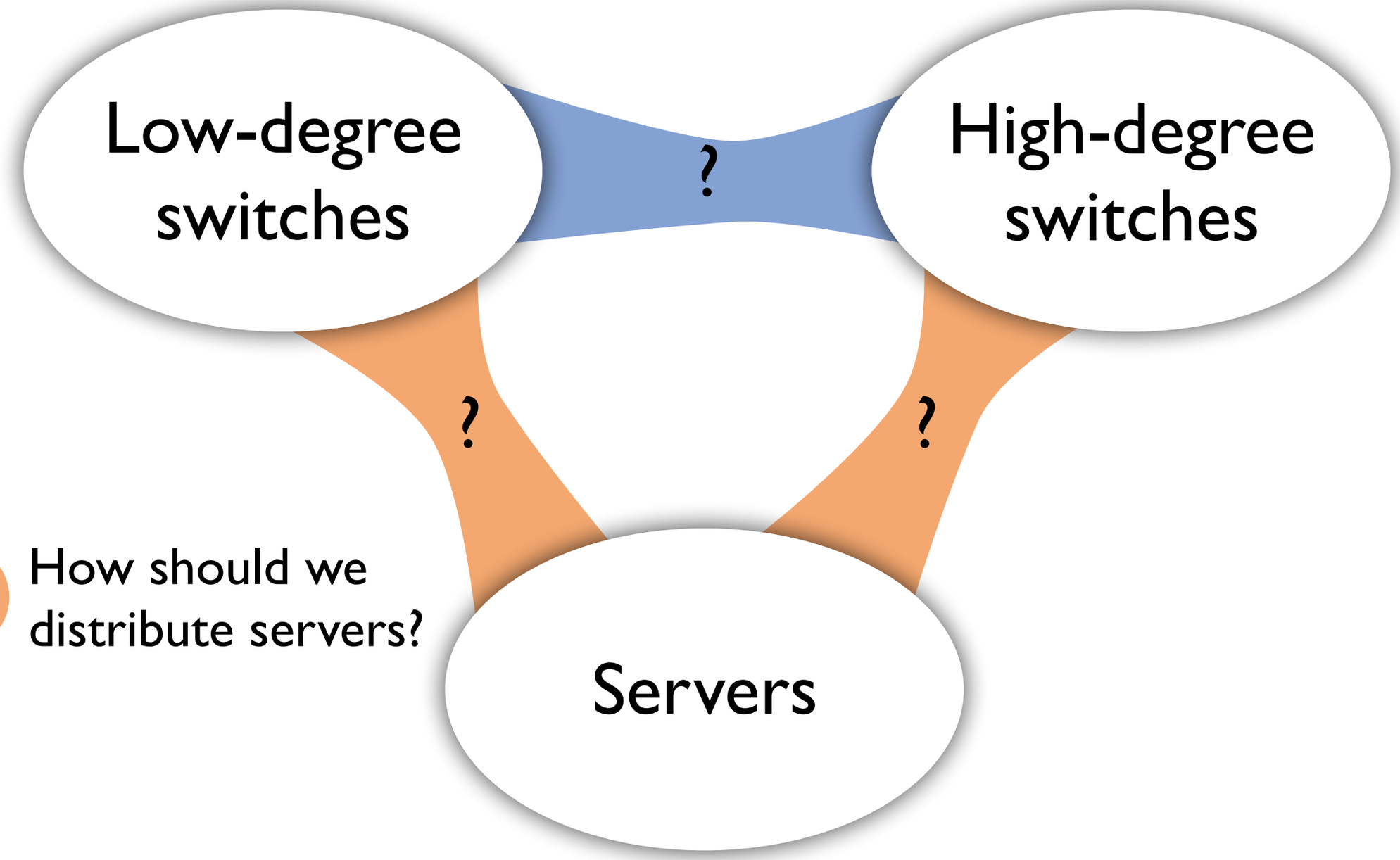
Random graphs as a building block



1. How should we distribute servers?

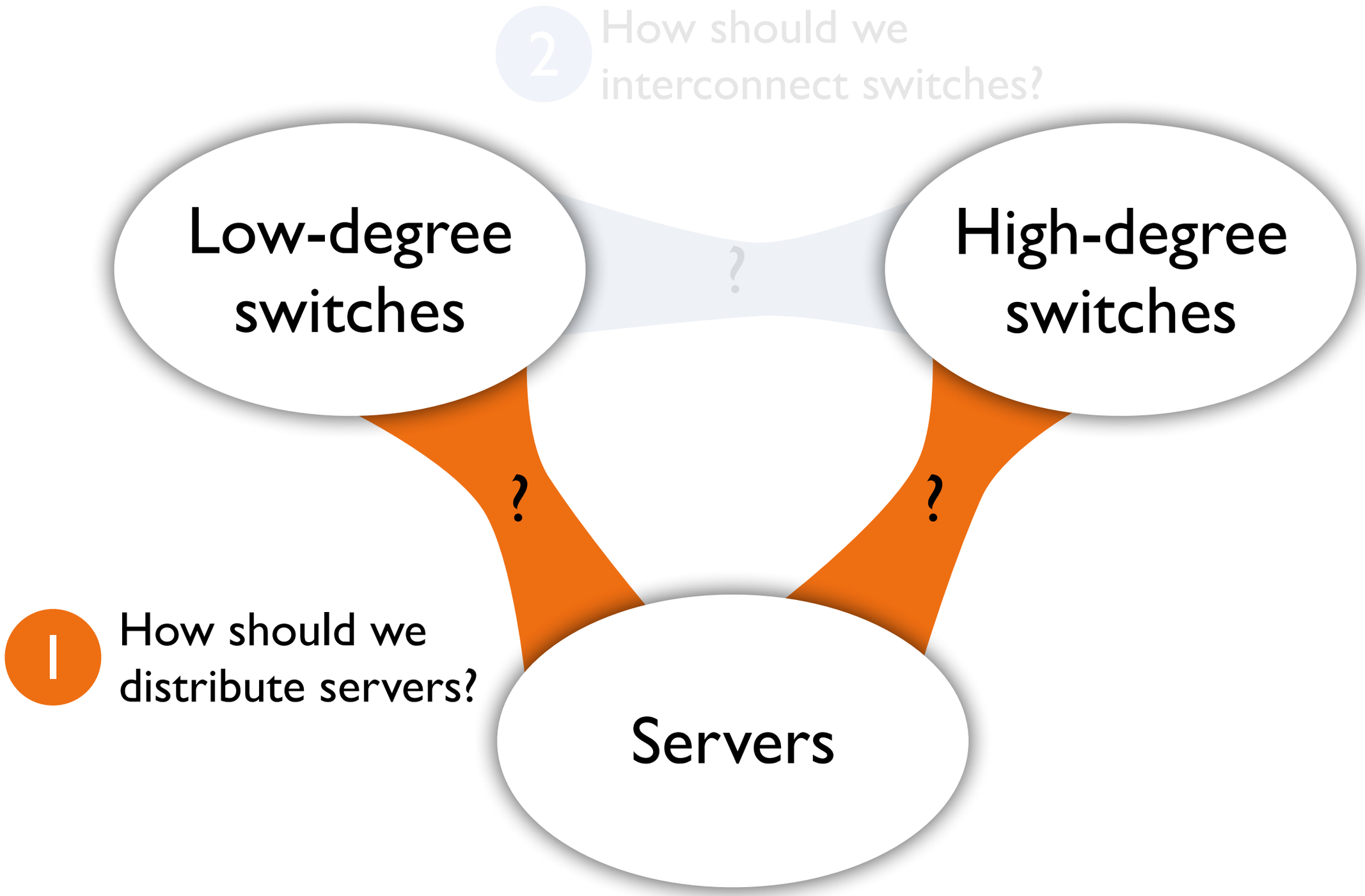
Random graphs as a building block

2 How should we interconnect switches?



1 How should we distribute servers?

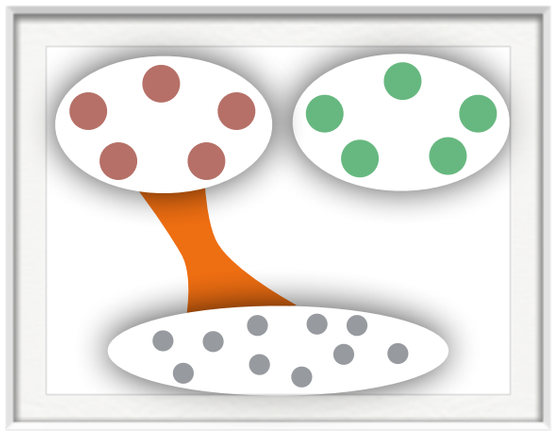
Random graphs as a building block



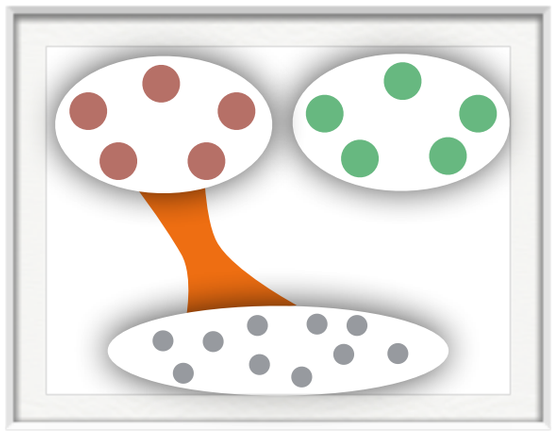
1 How should we distribute servers?

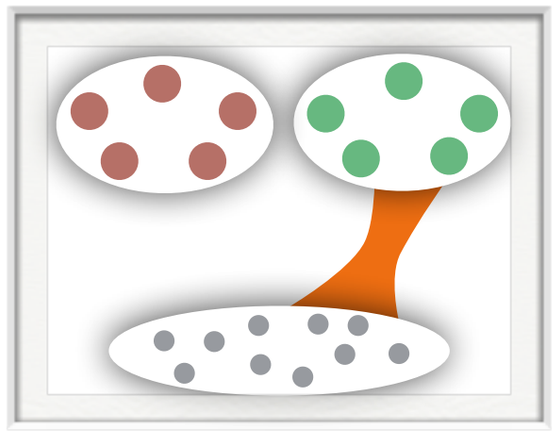
Distributing servers

Distributing servers

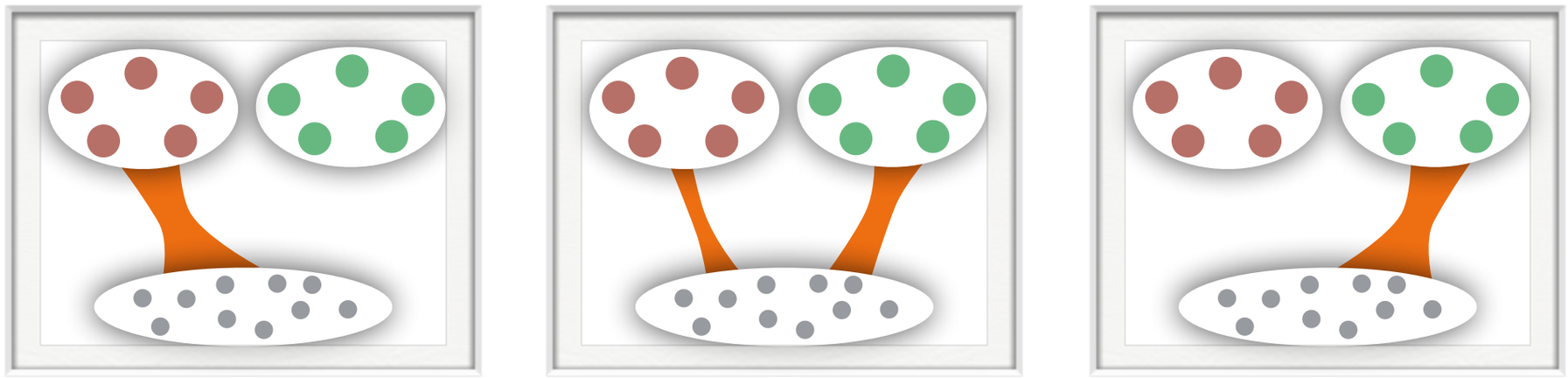


Distributing servers

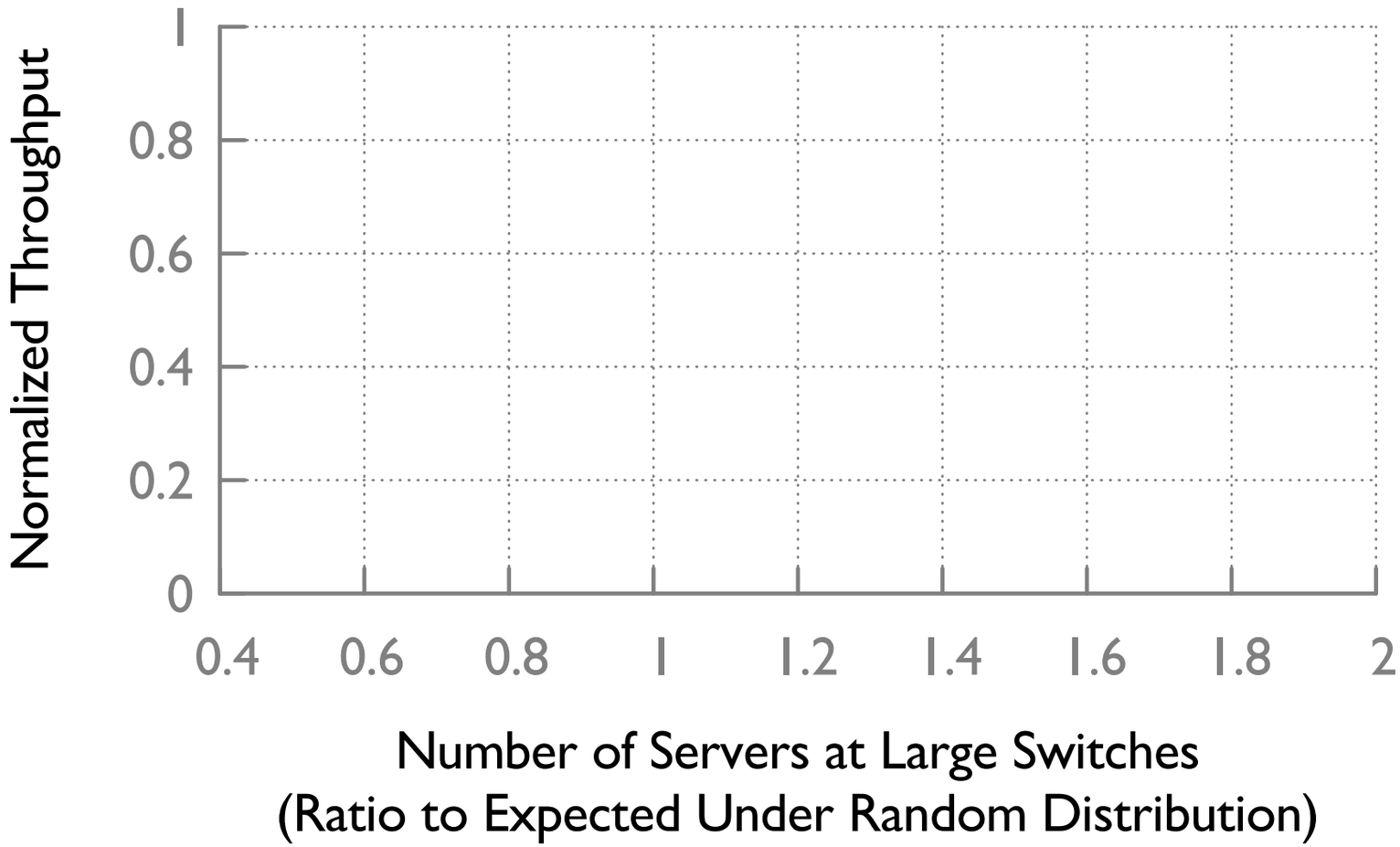


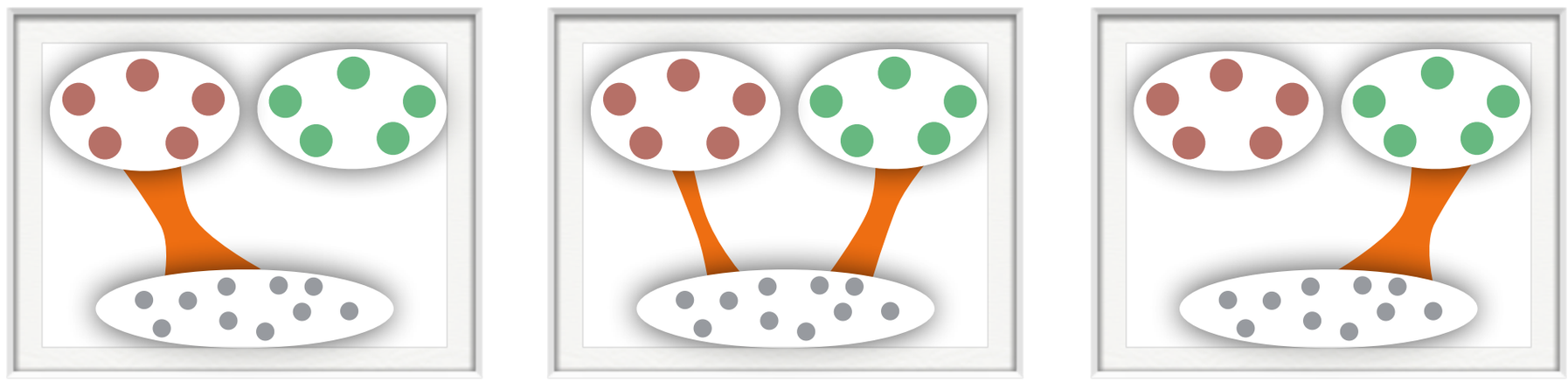


Distributing servers

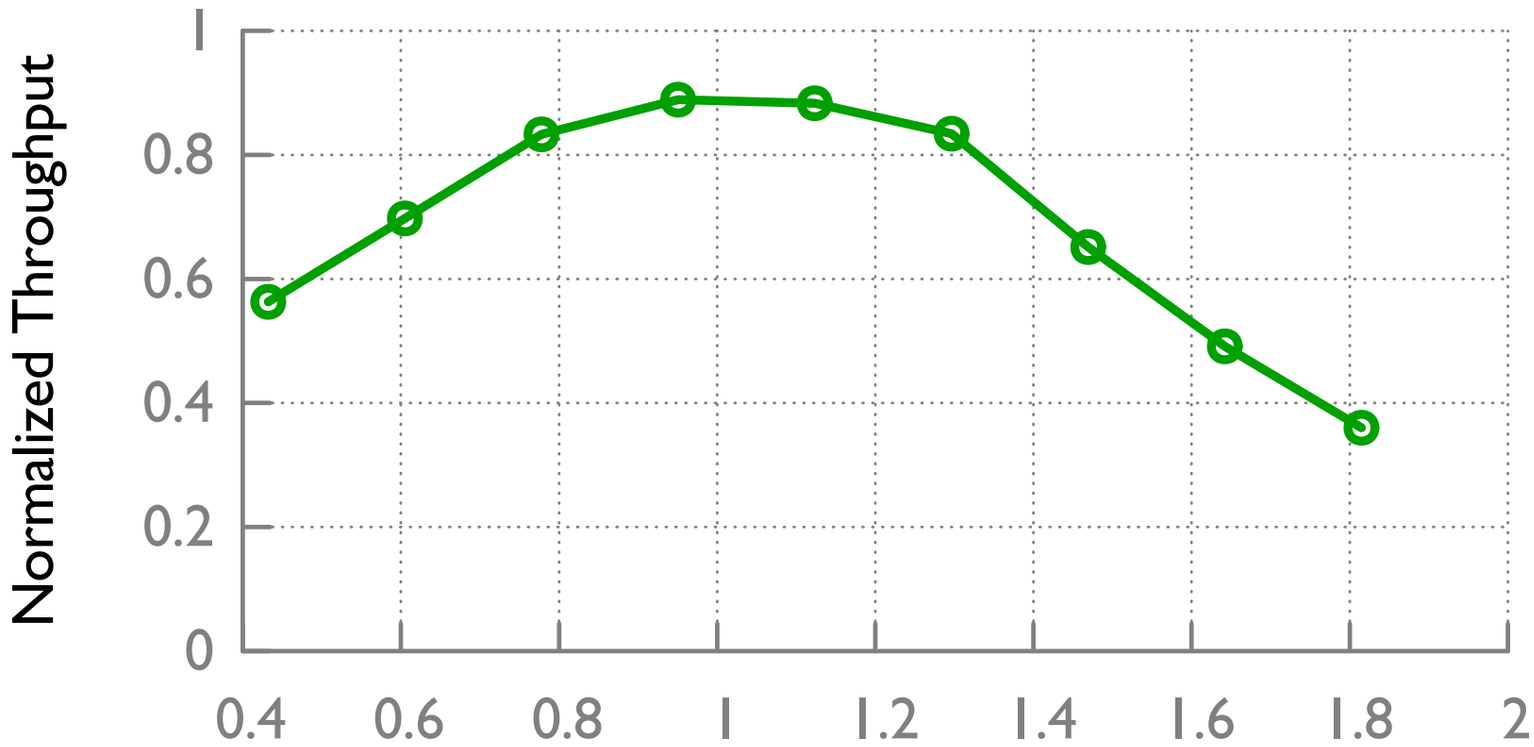


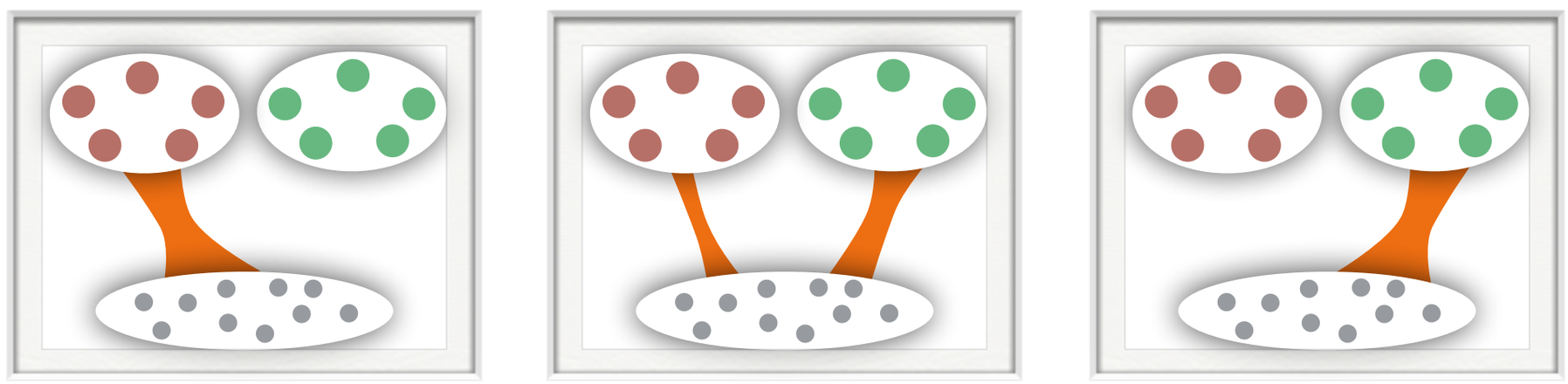
Distributing servers



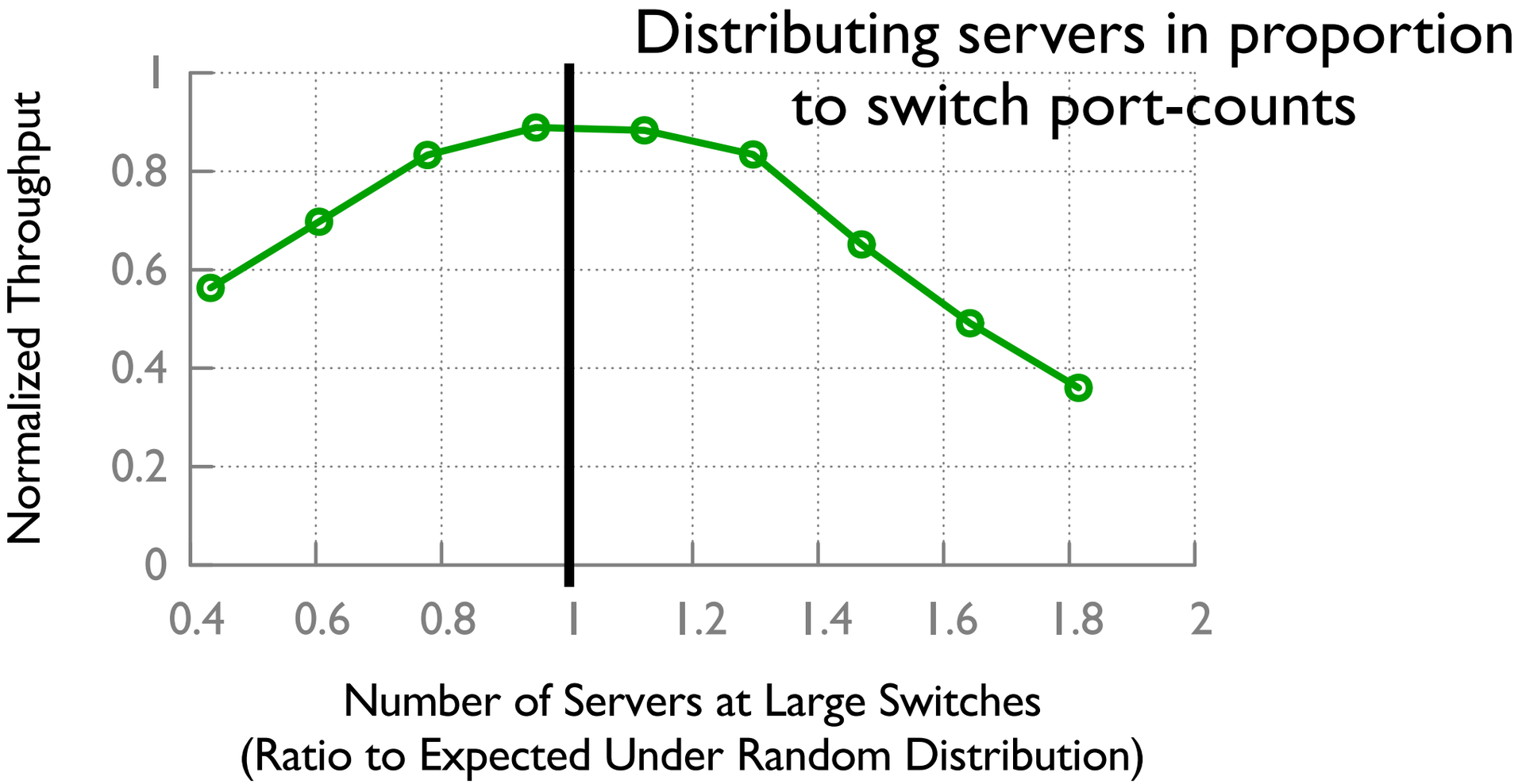


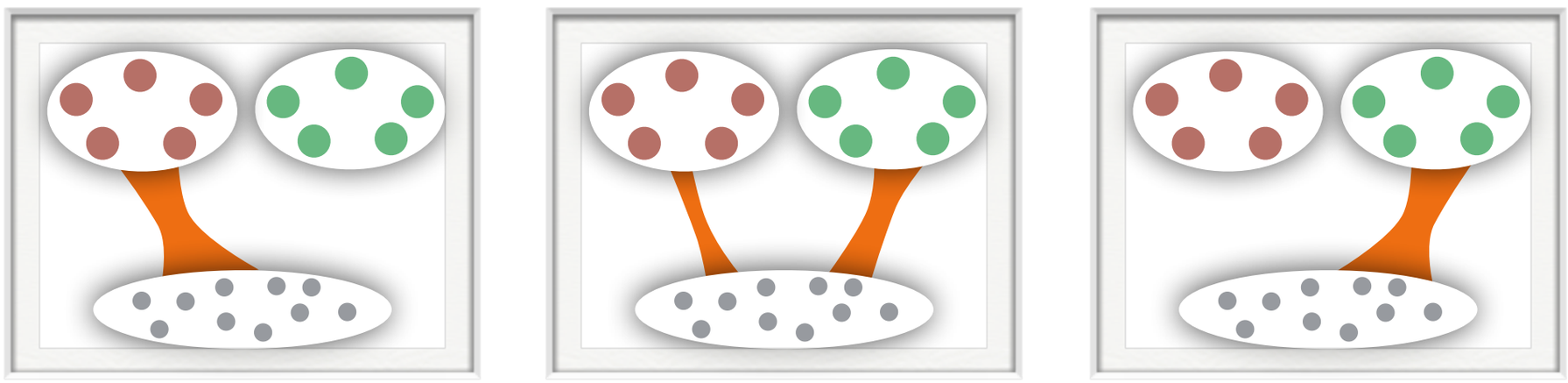
Distributing servers



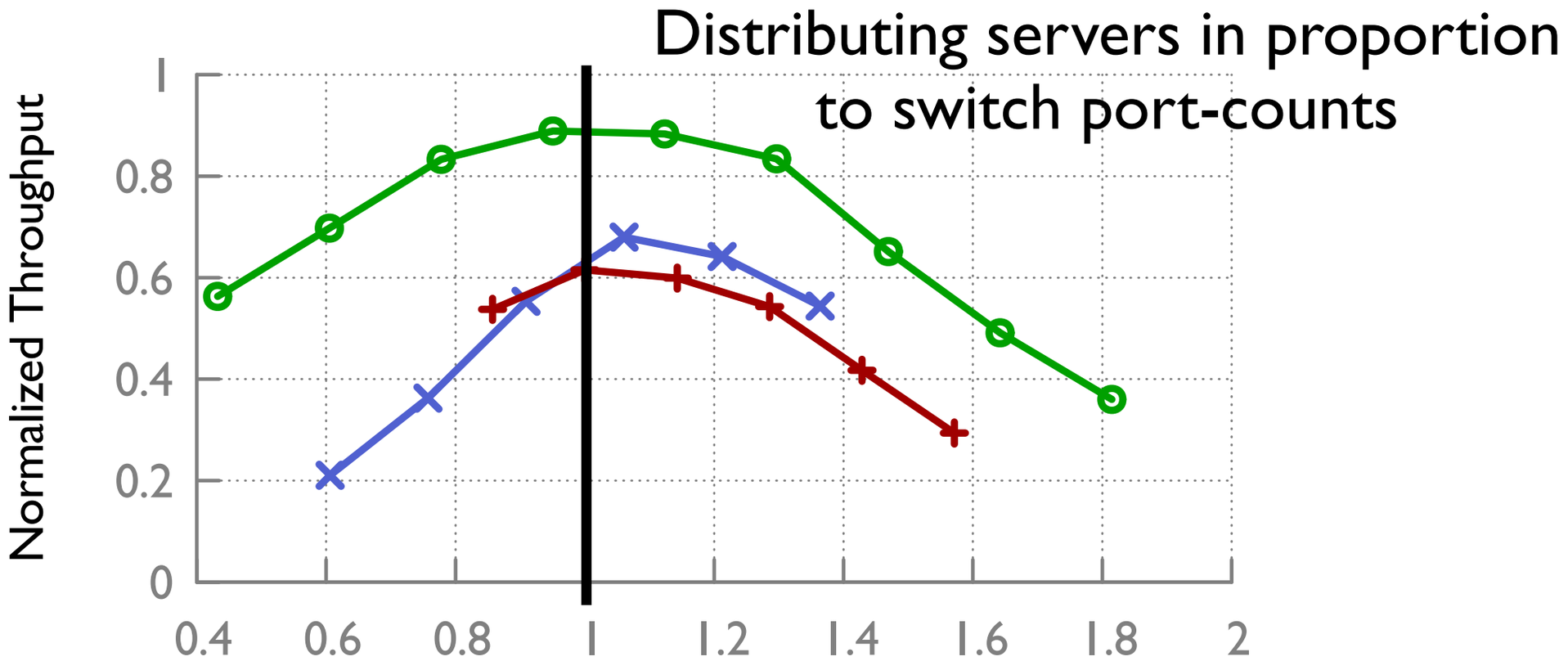


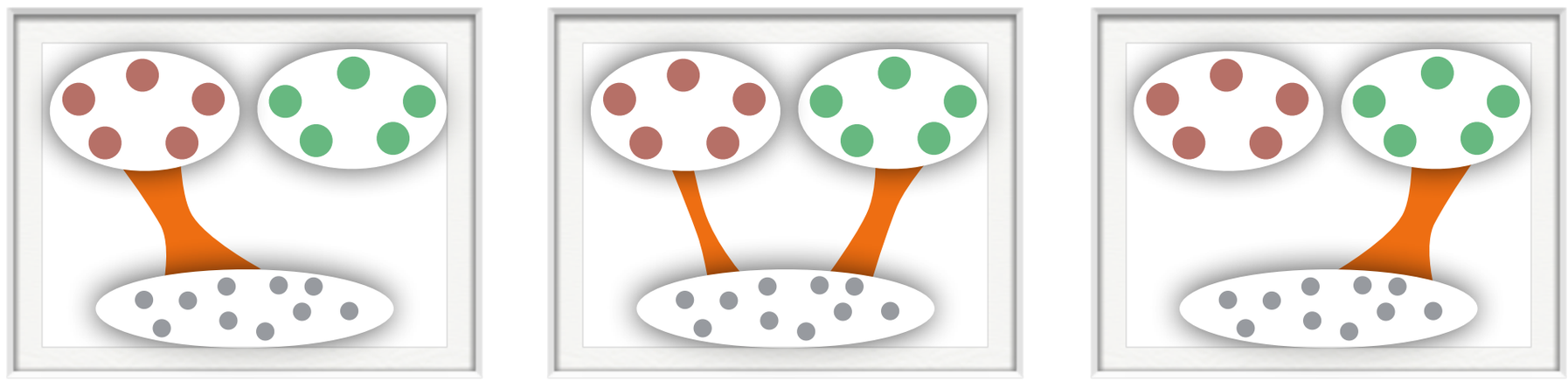
Distributing servers



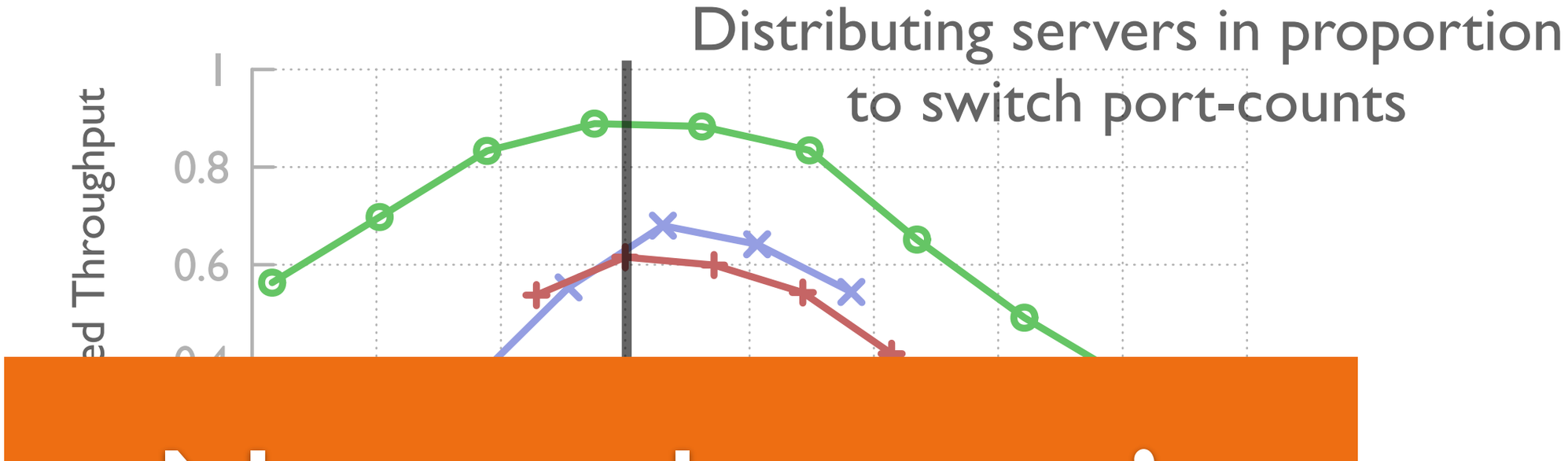


Distributing servers

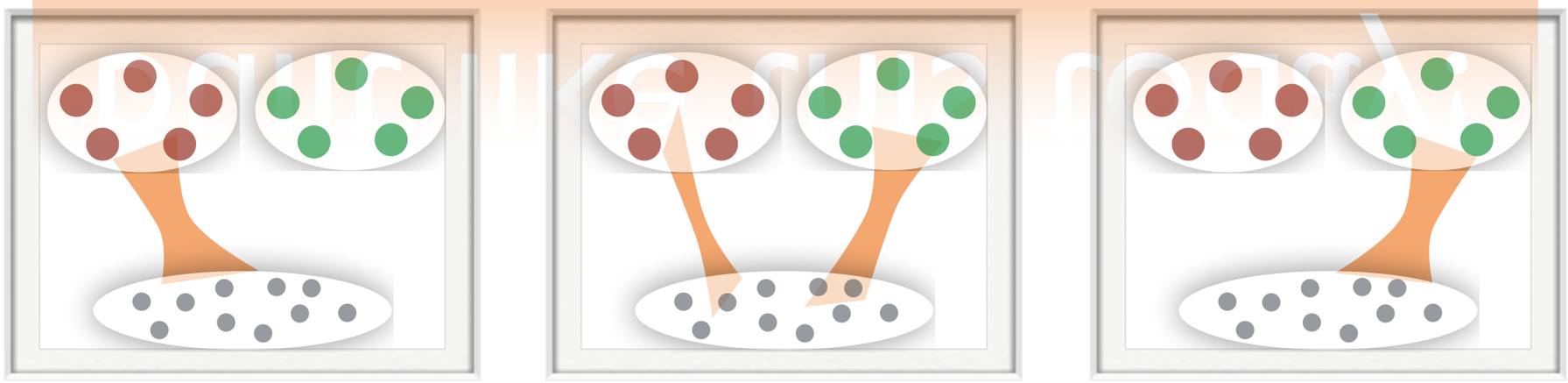




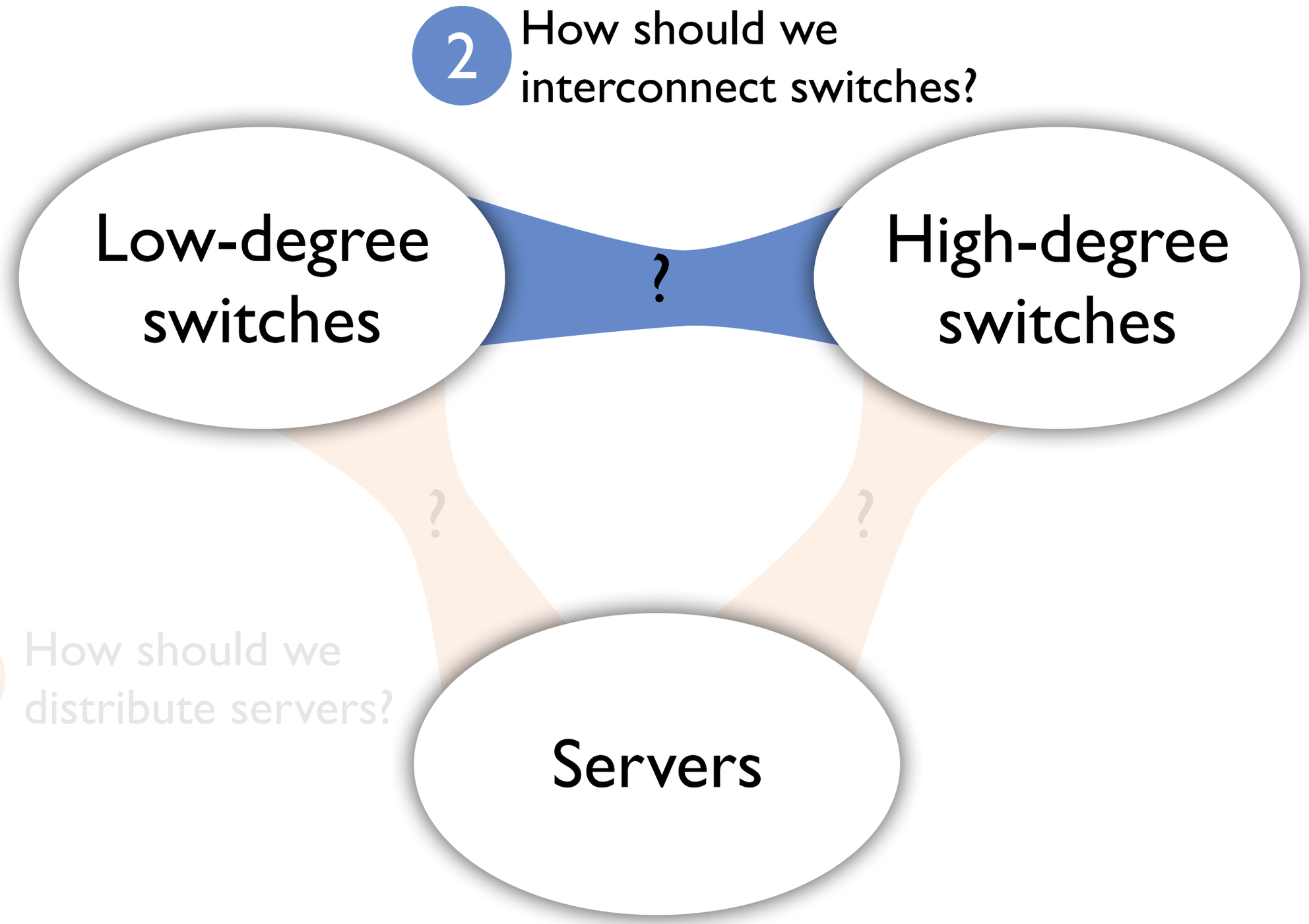
Distributing servers



Networks aren't built like this today!

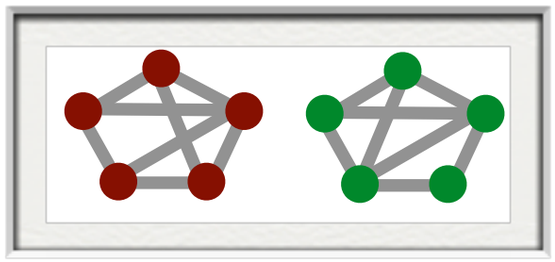


Random graphs as a building block

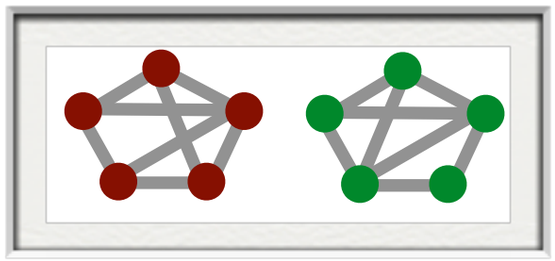


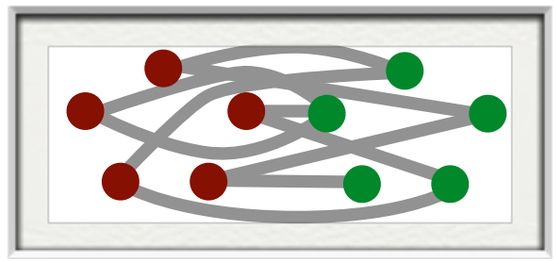
Interconnecting switches

Interconnecting switches

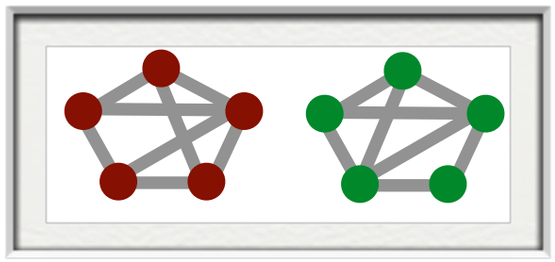


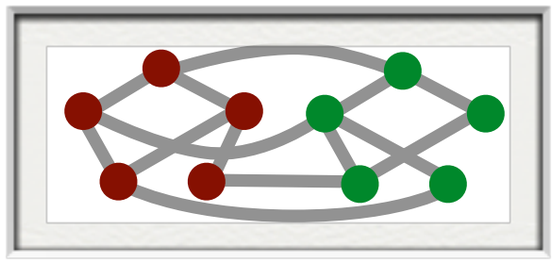
Interconnecting switches

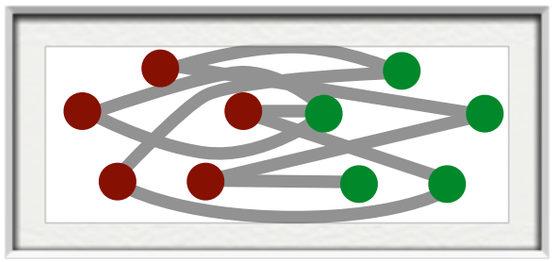




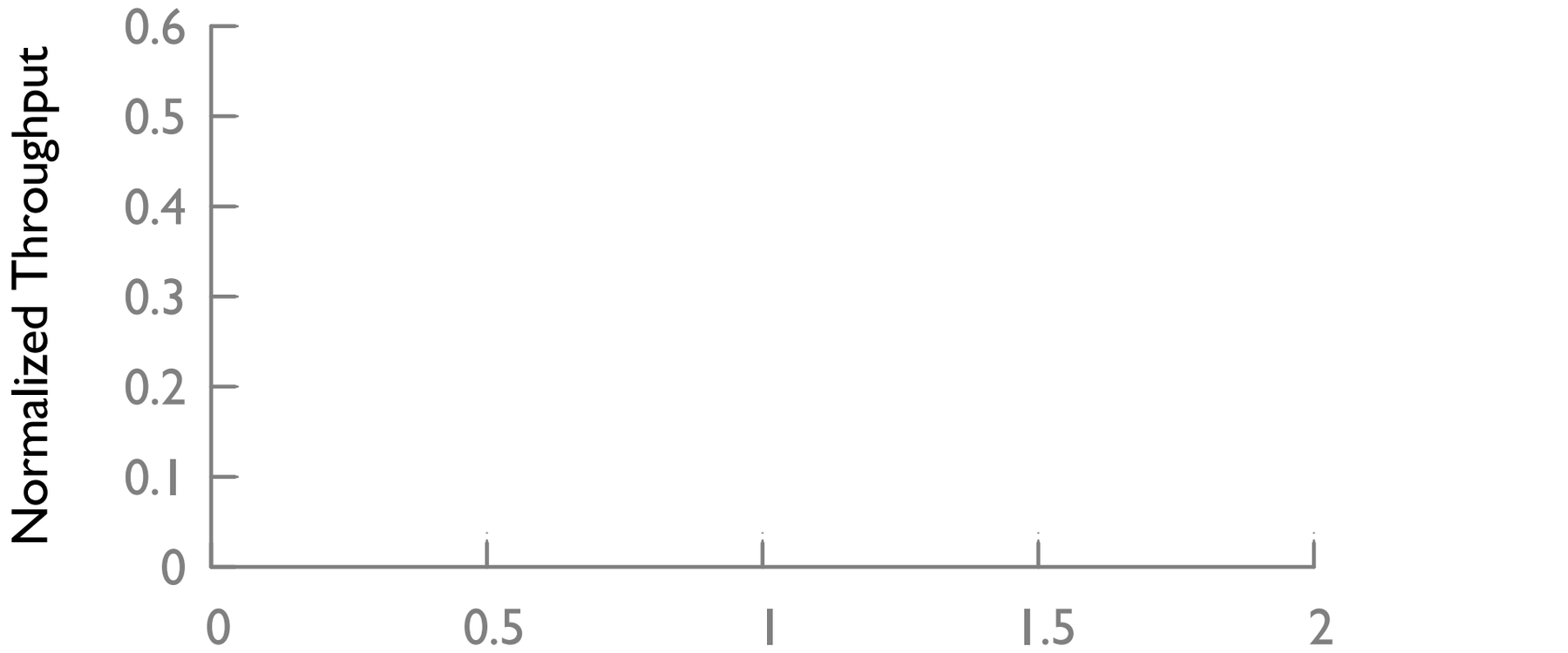
Interconnecting switches

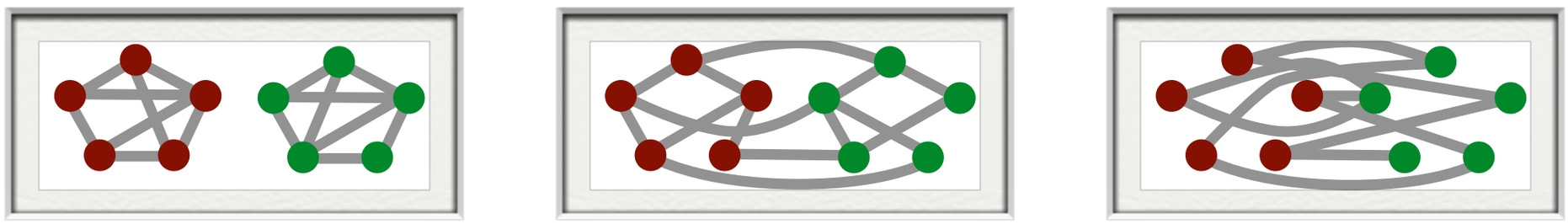




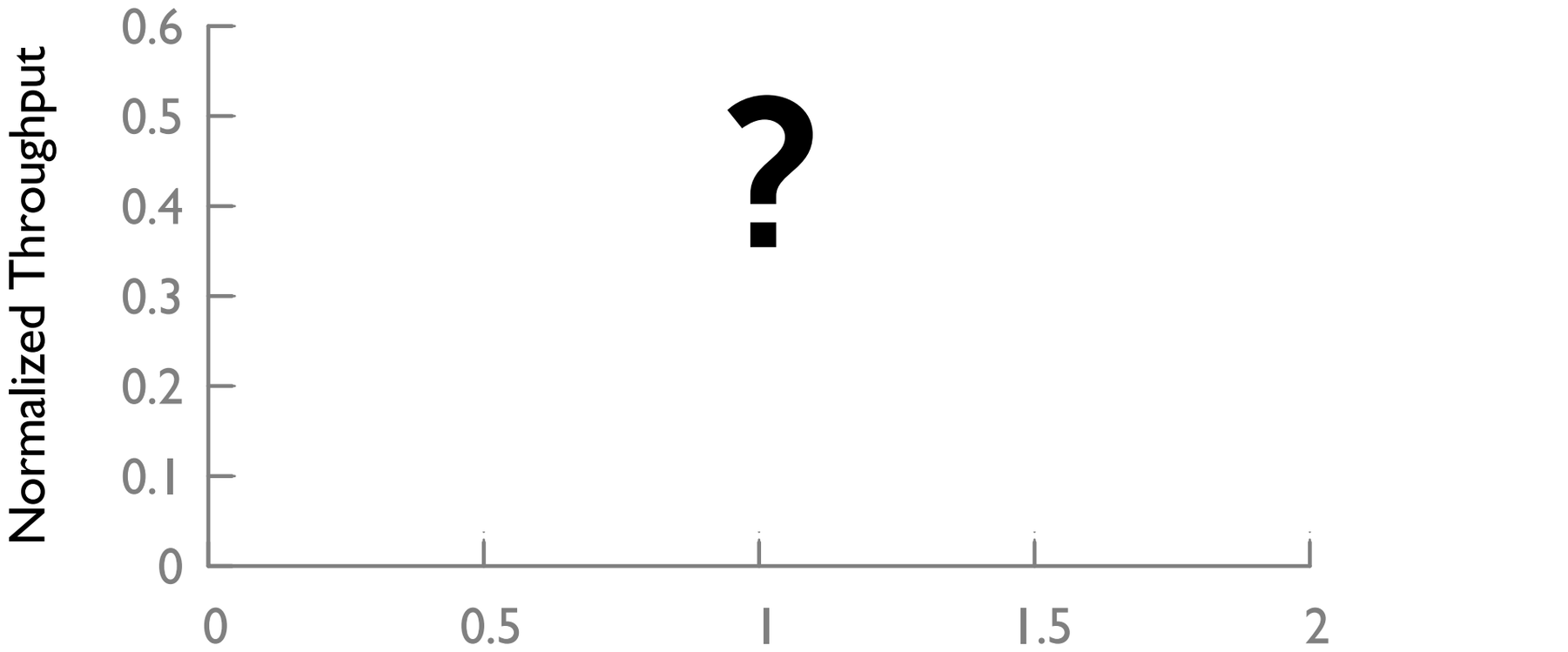


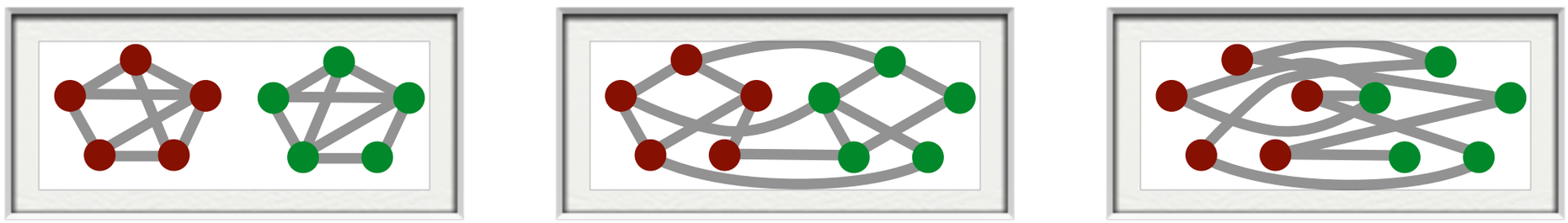
Interconnecting switches



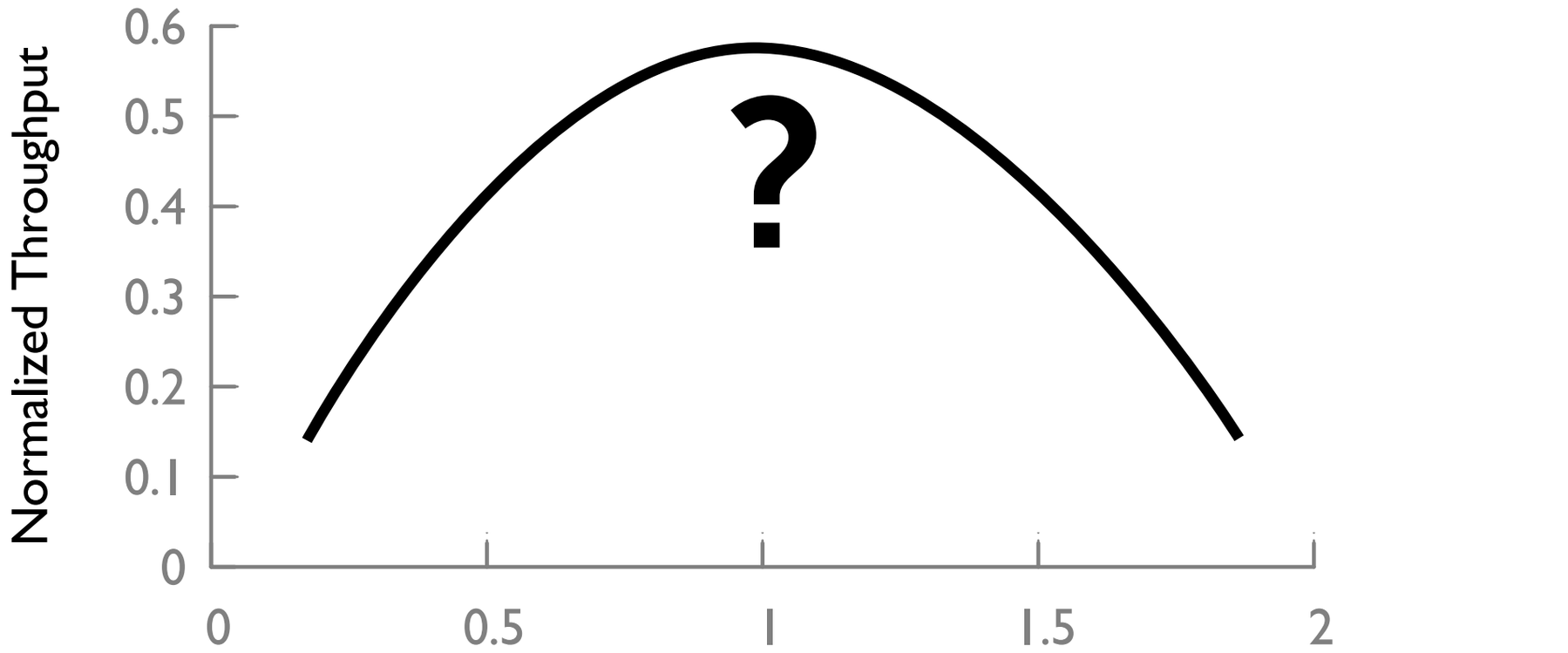


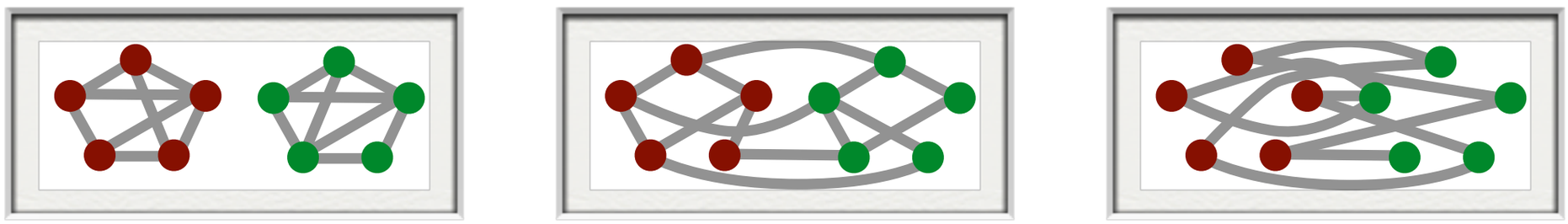
Interconnecting switches



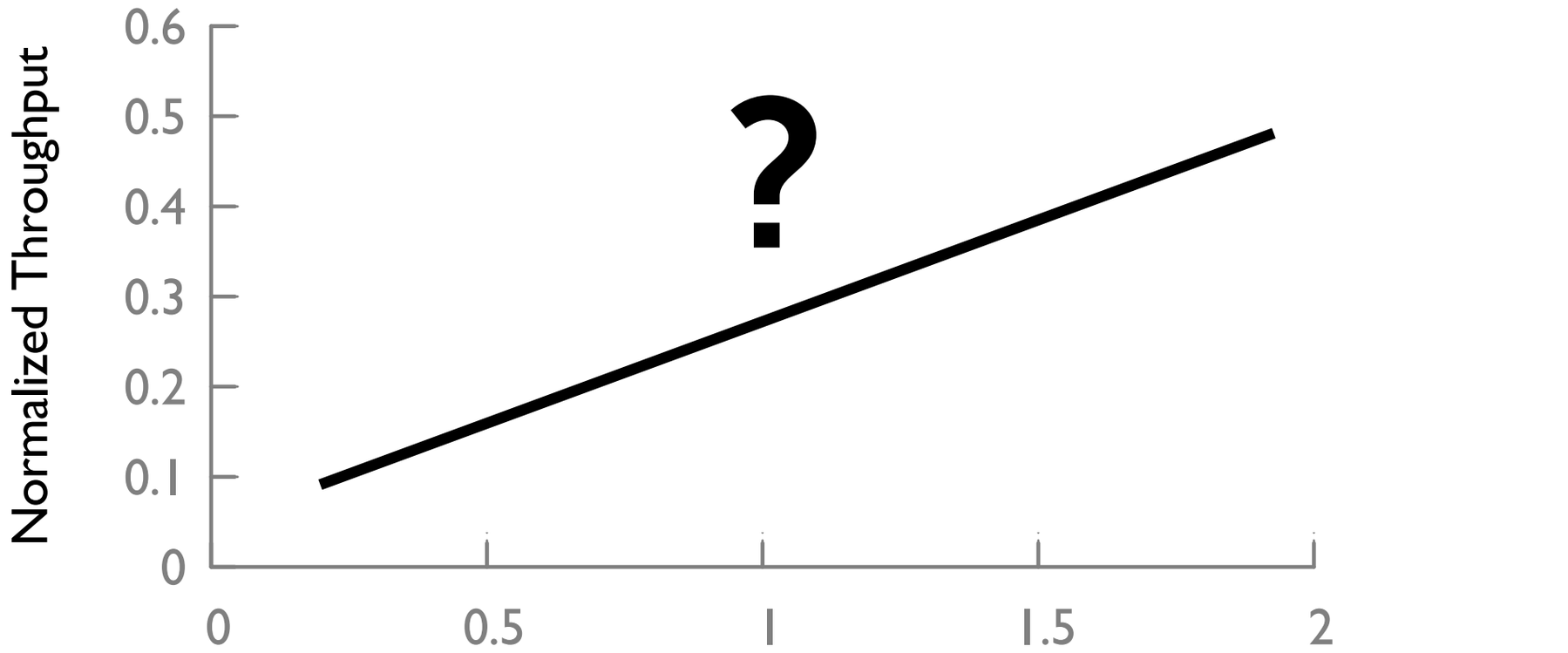


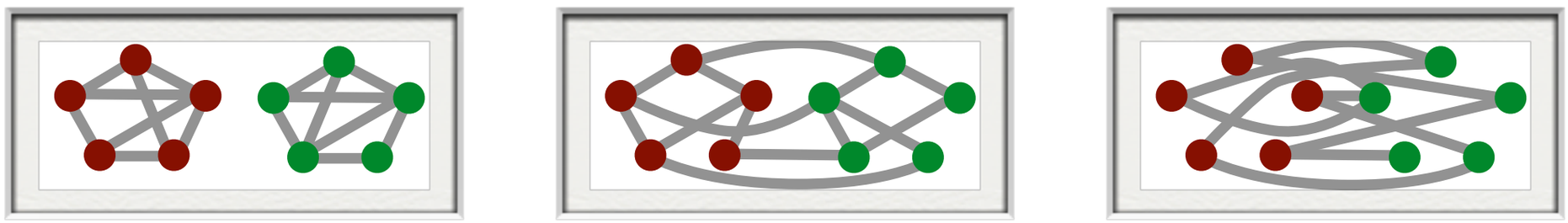
Interconnecting switches



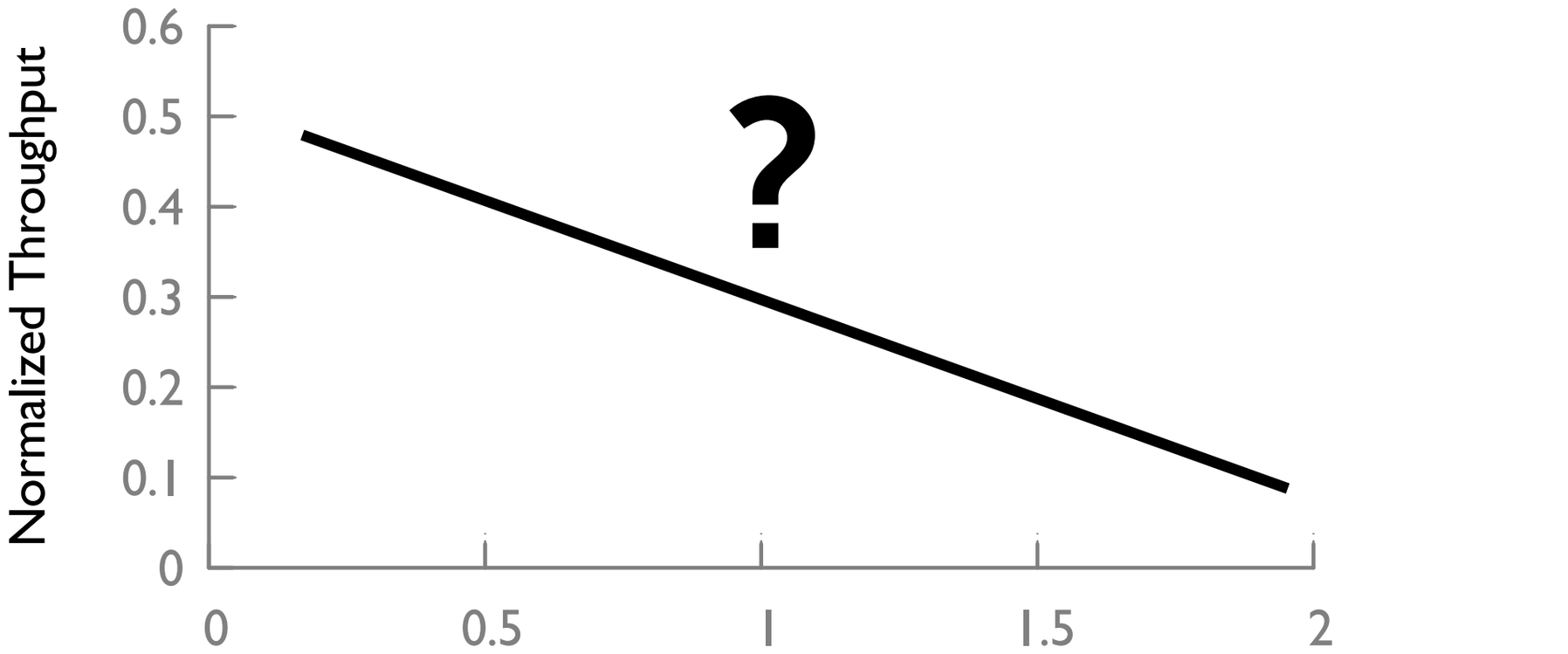


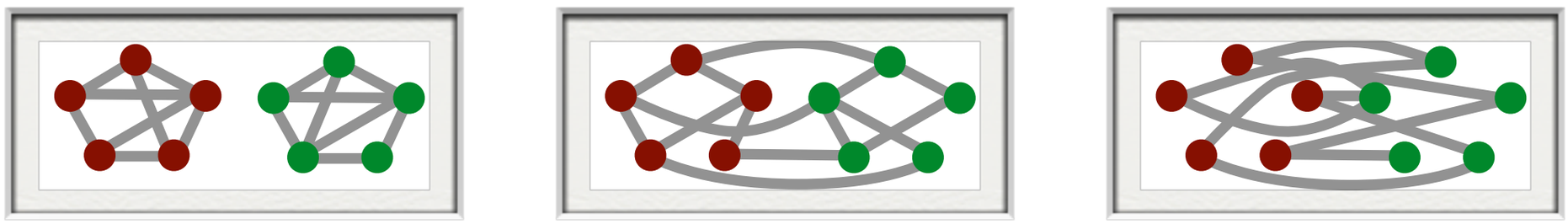
Interconnecting switches



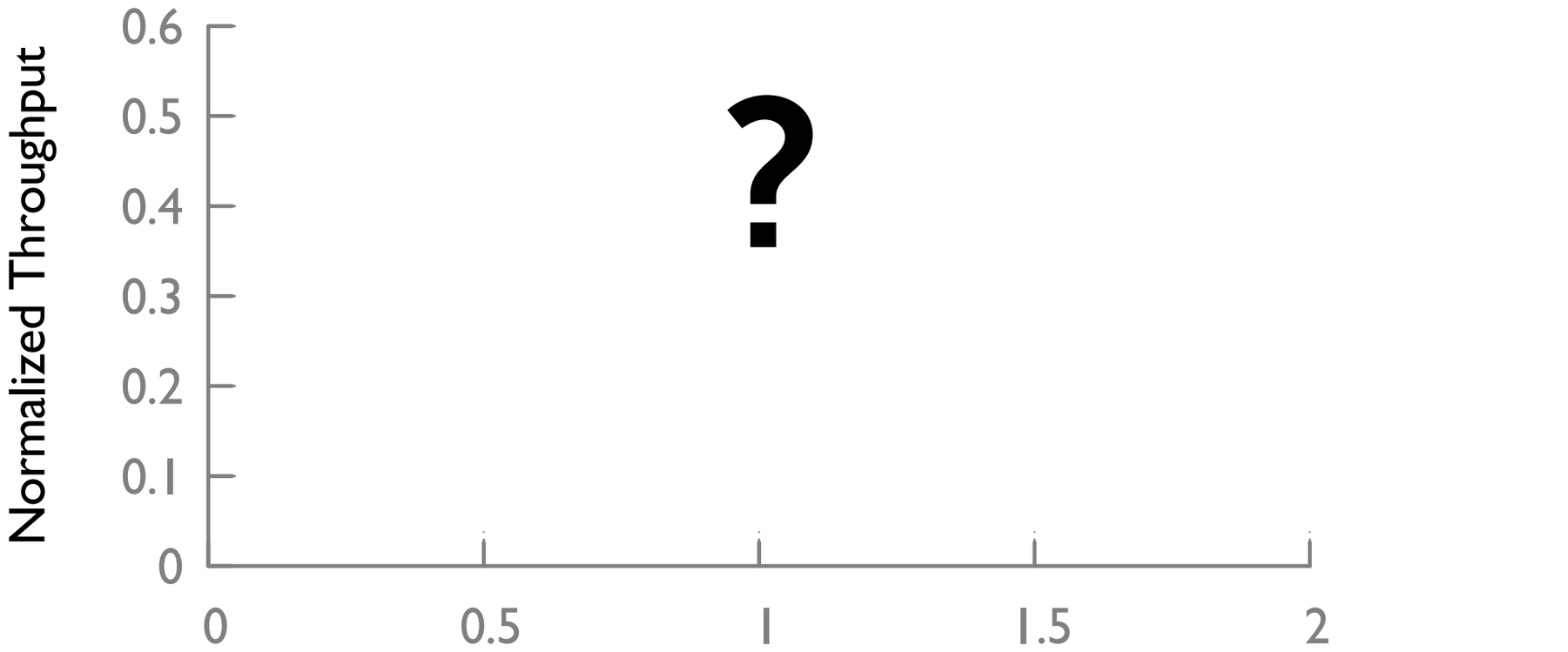


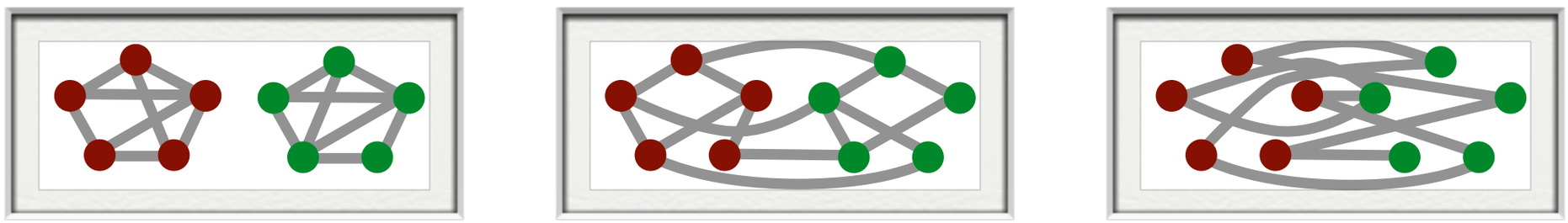
Interconnecting switches



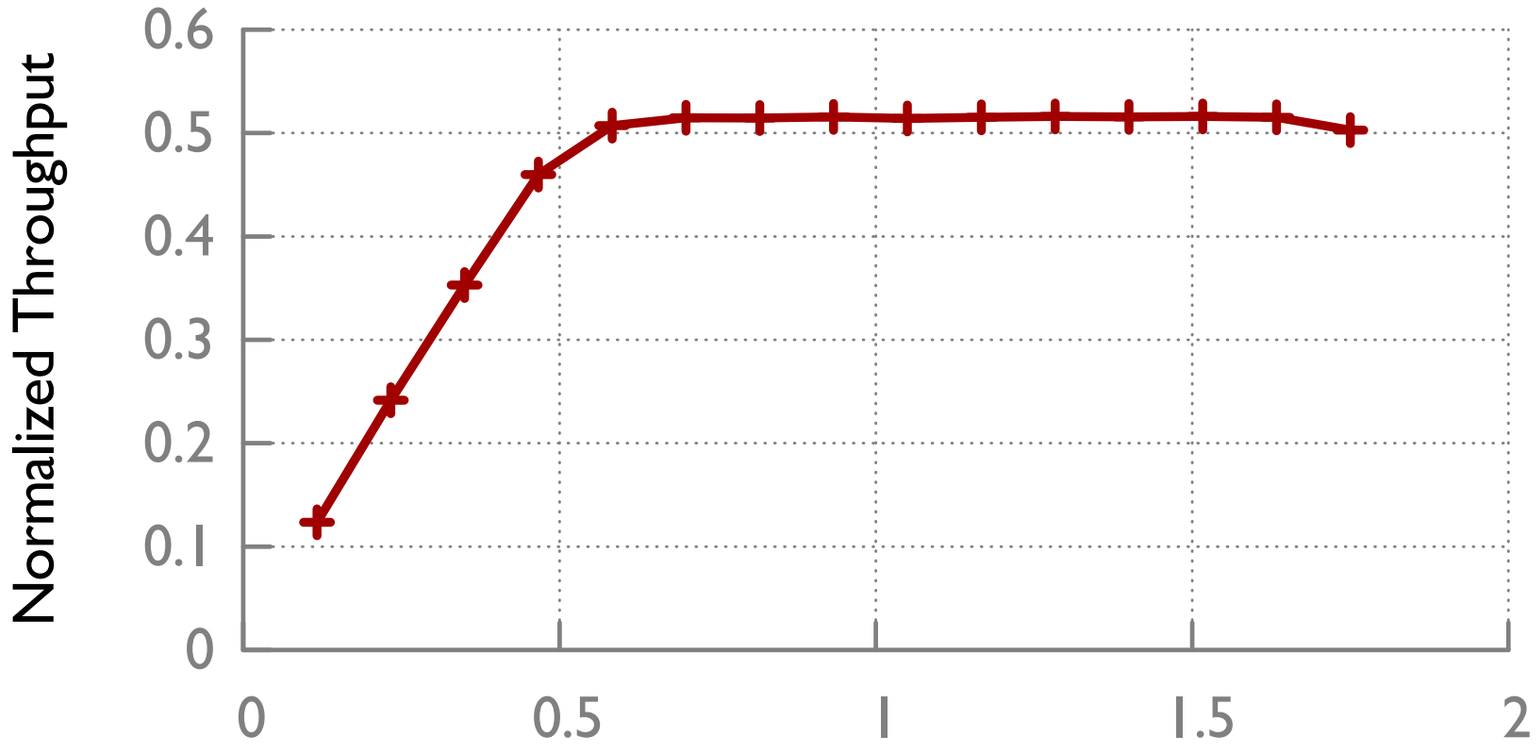


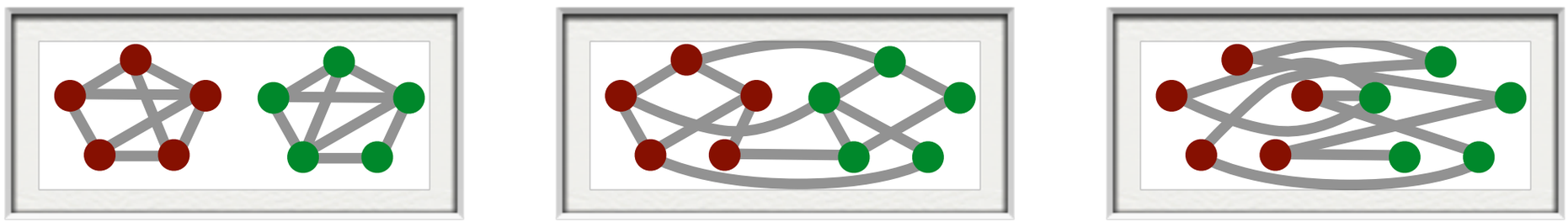
Interconnecting switches



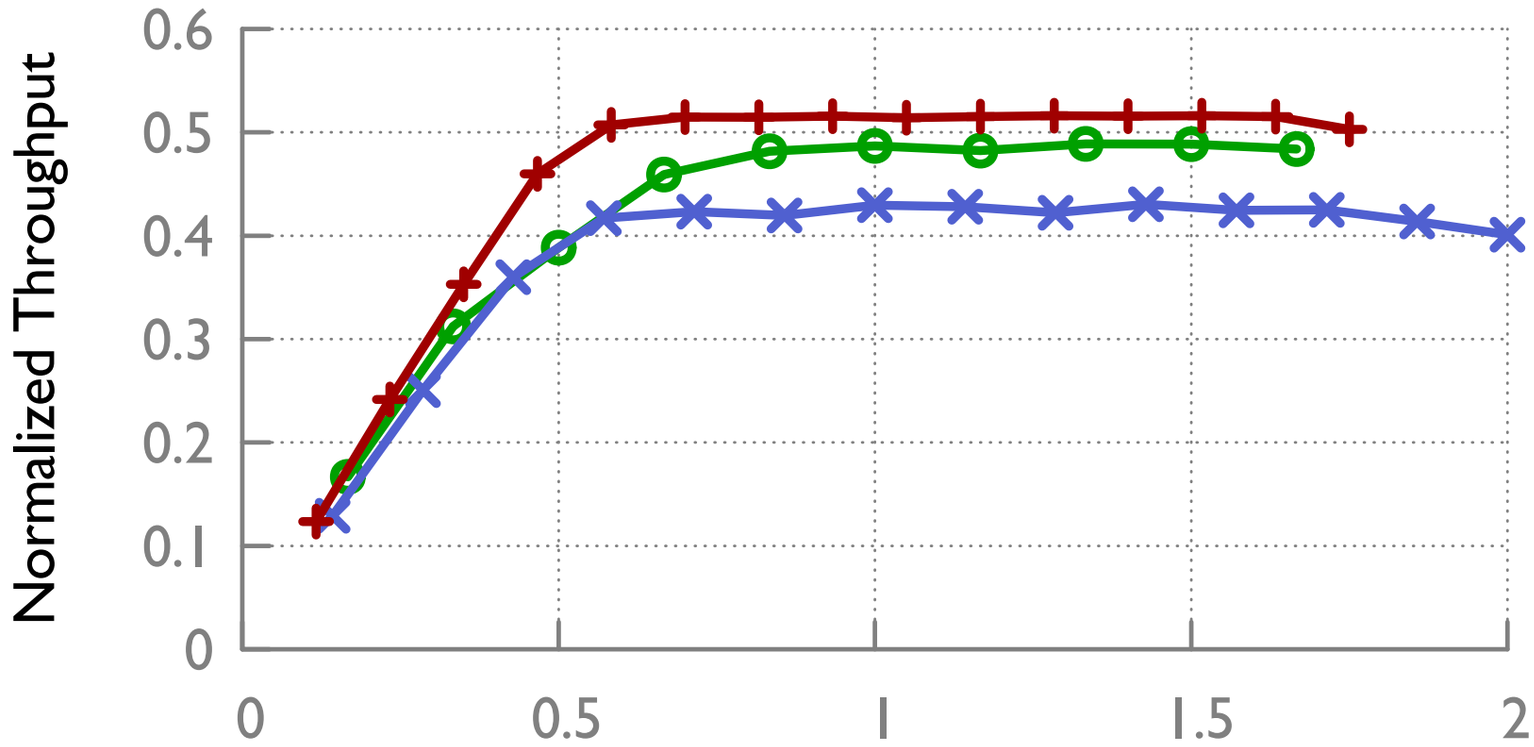


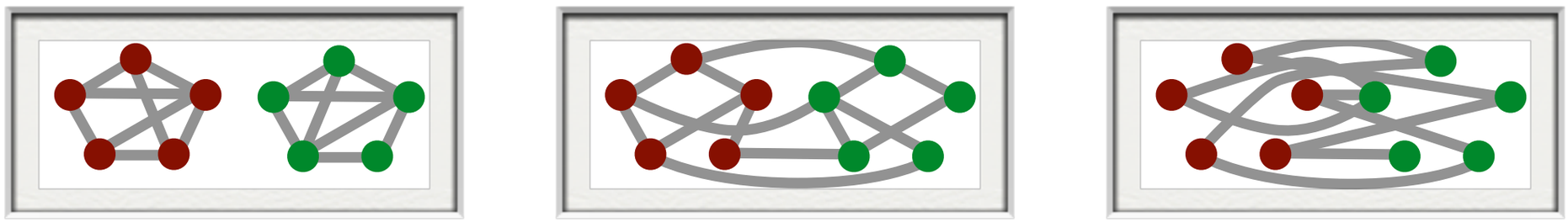
Interconnecting switches



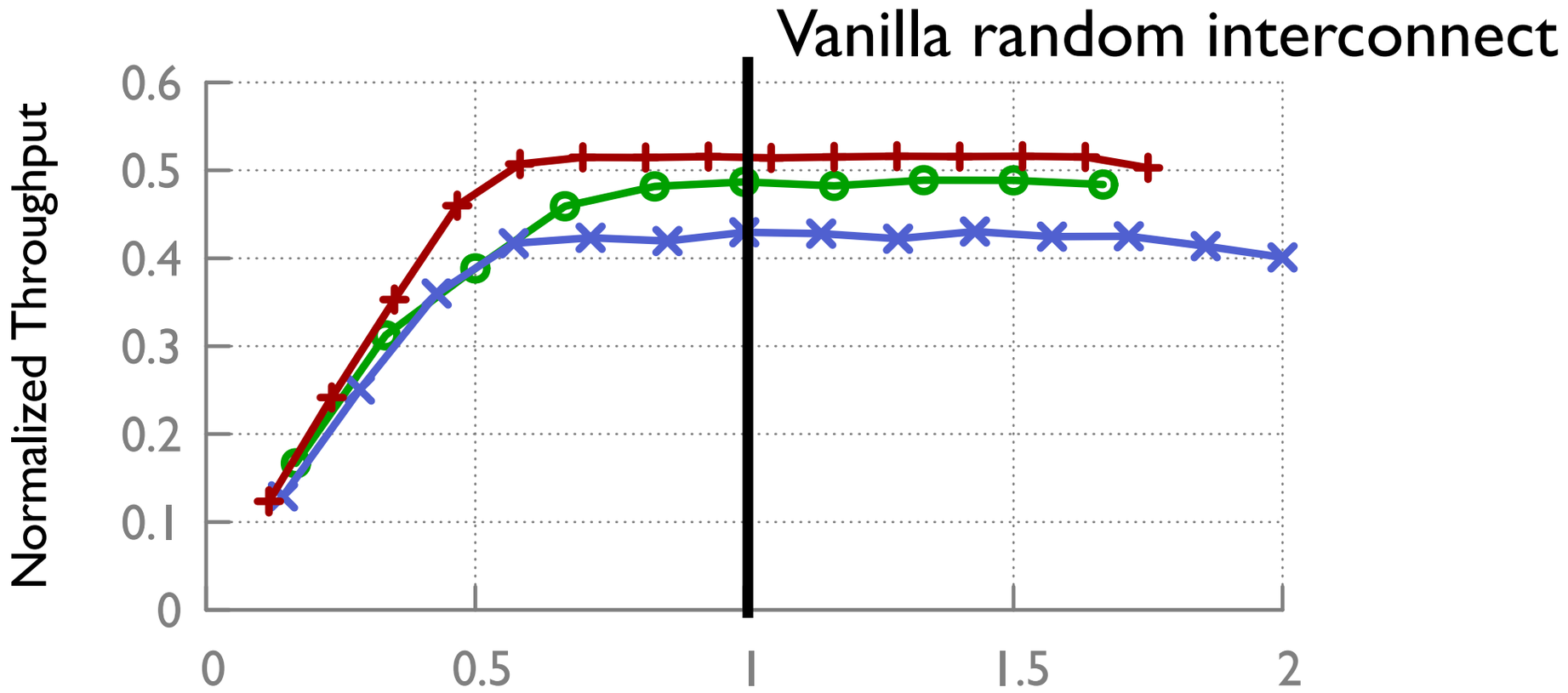


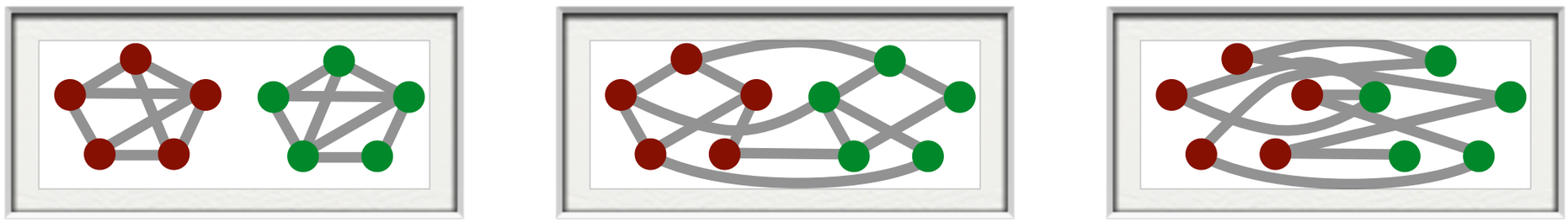
Interconnecting switches



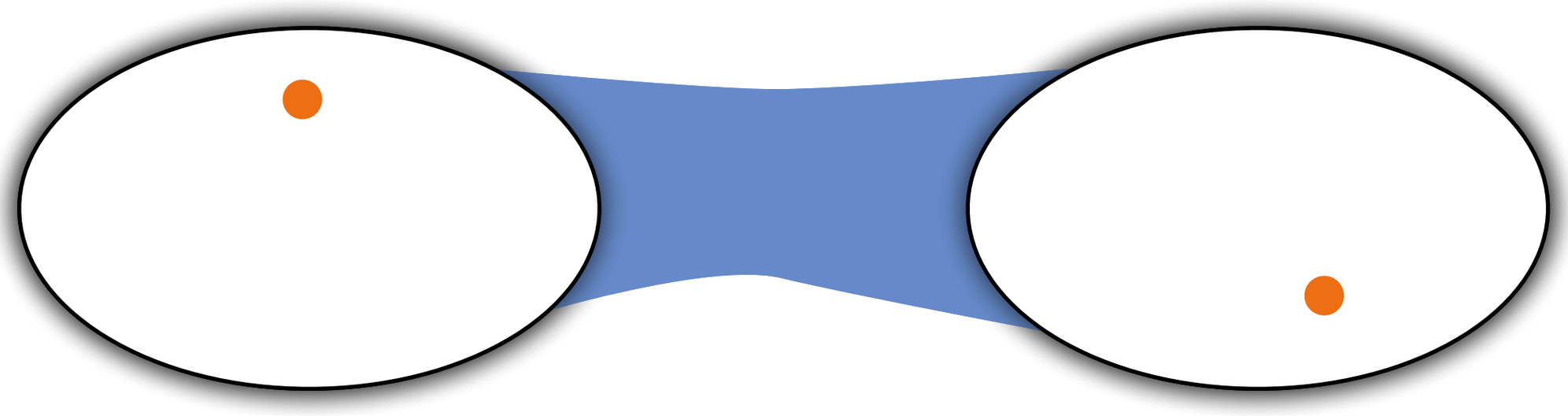


Interconnecting switches

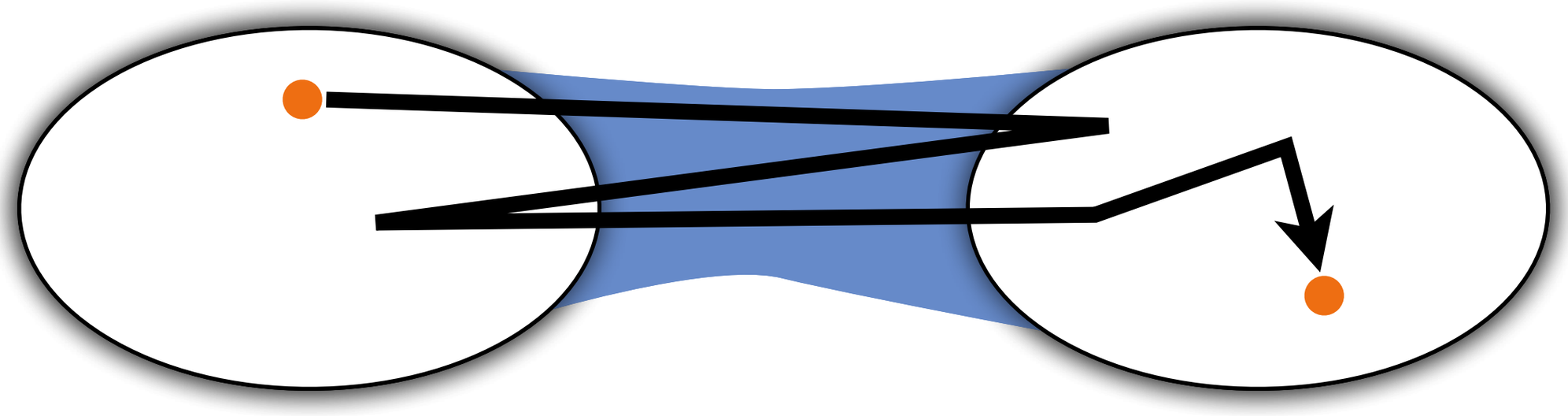




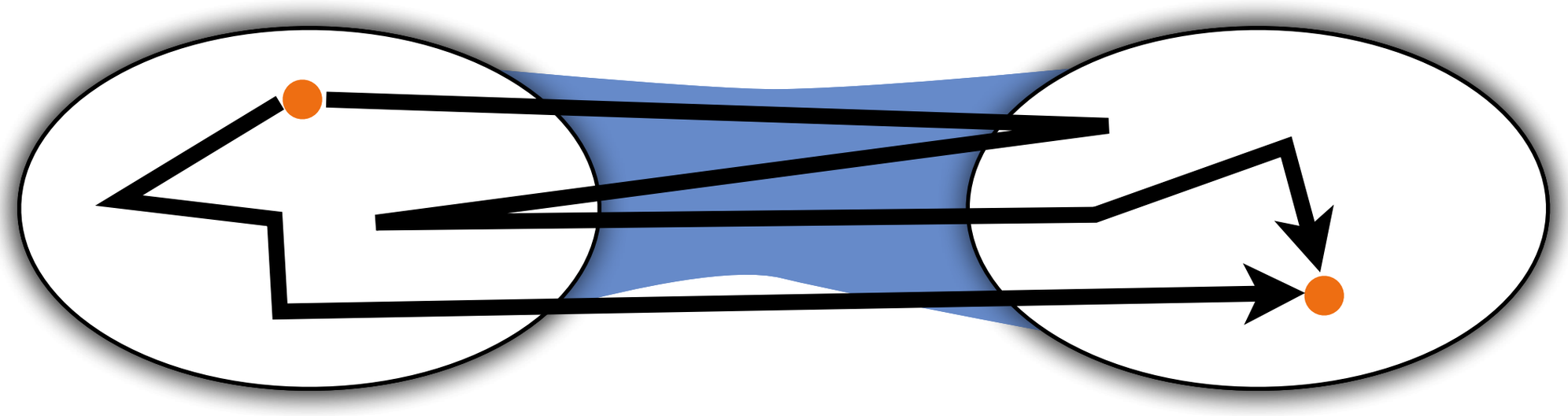
Intuition



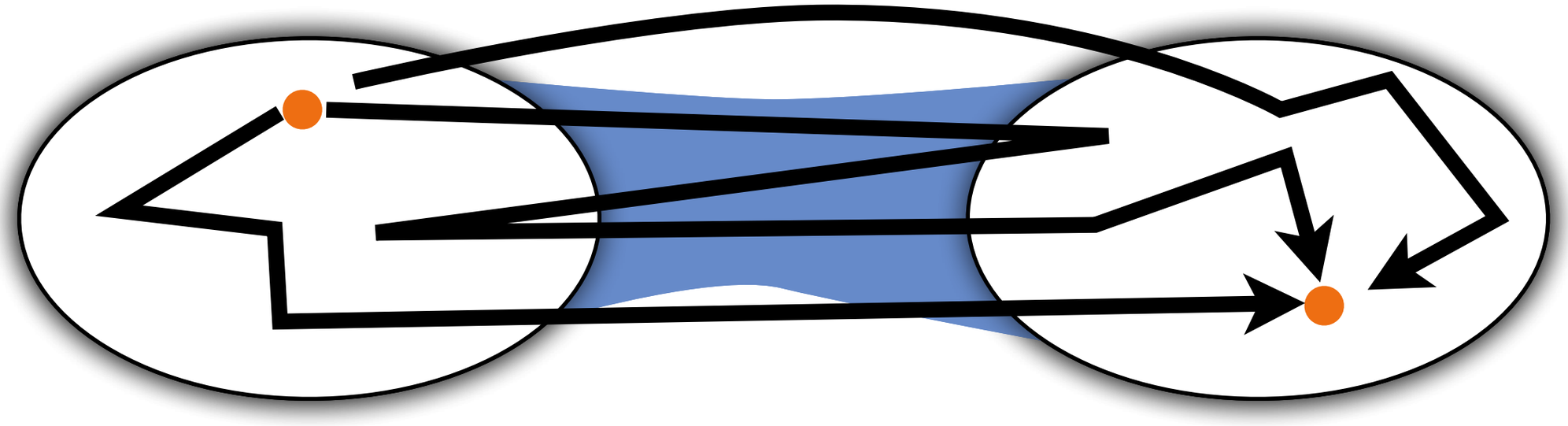
Intuition



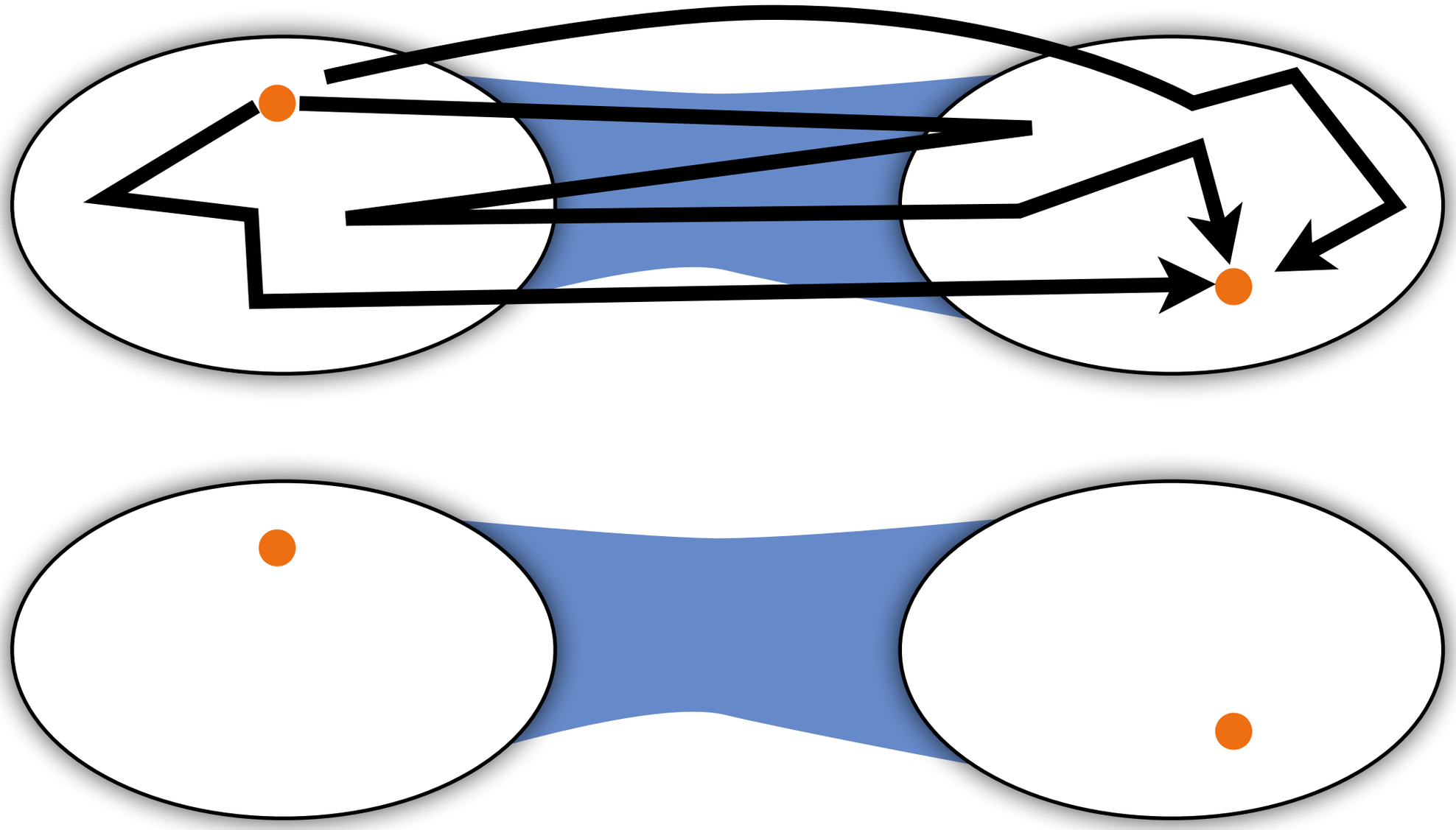
Intuition



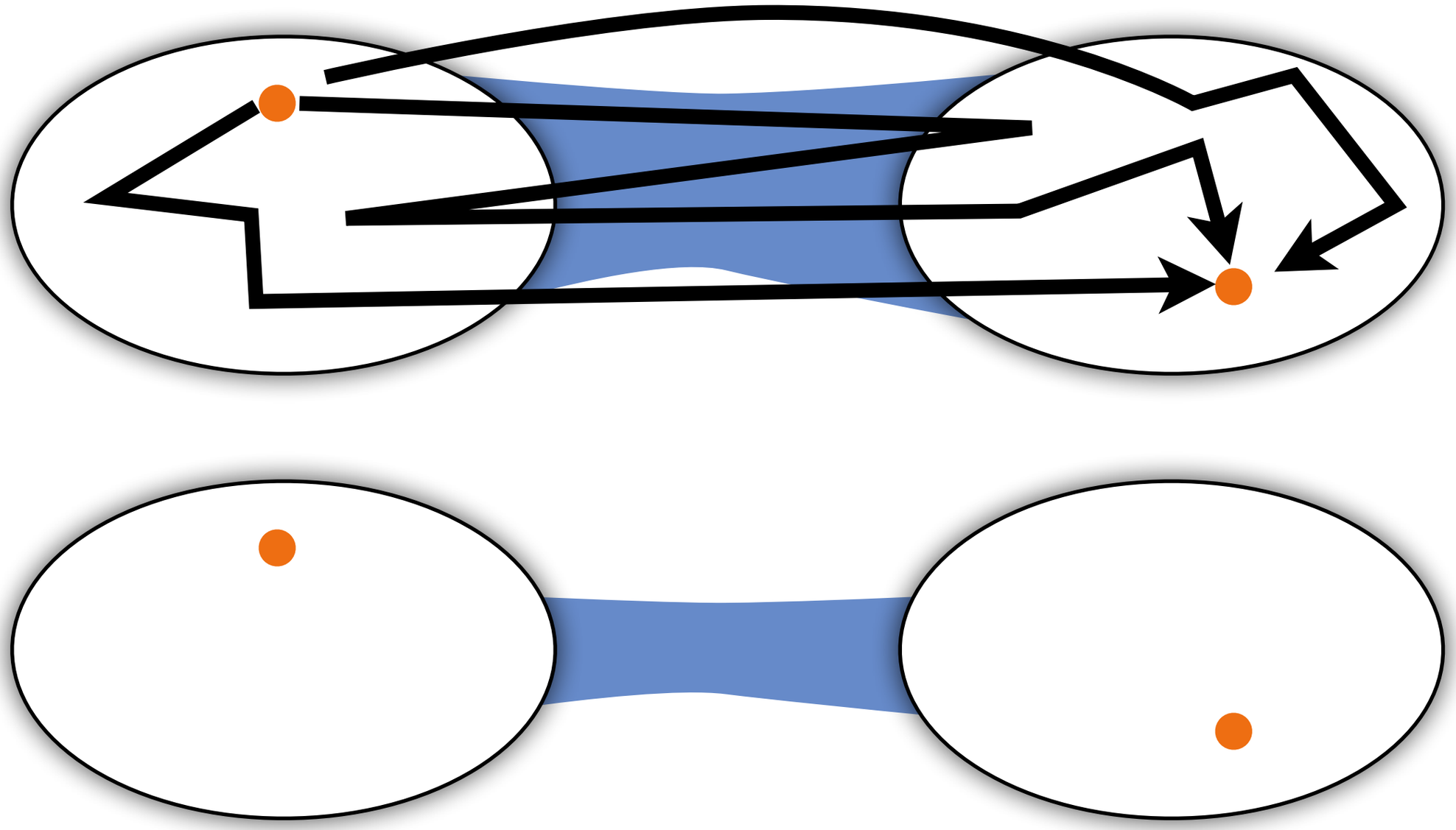
Intuition



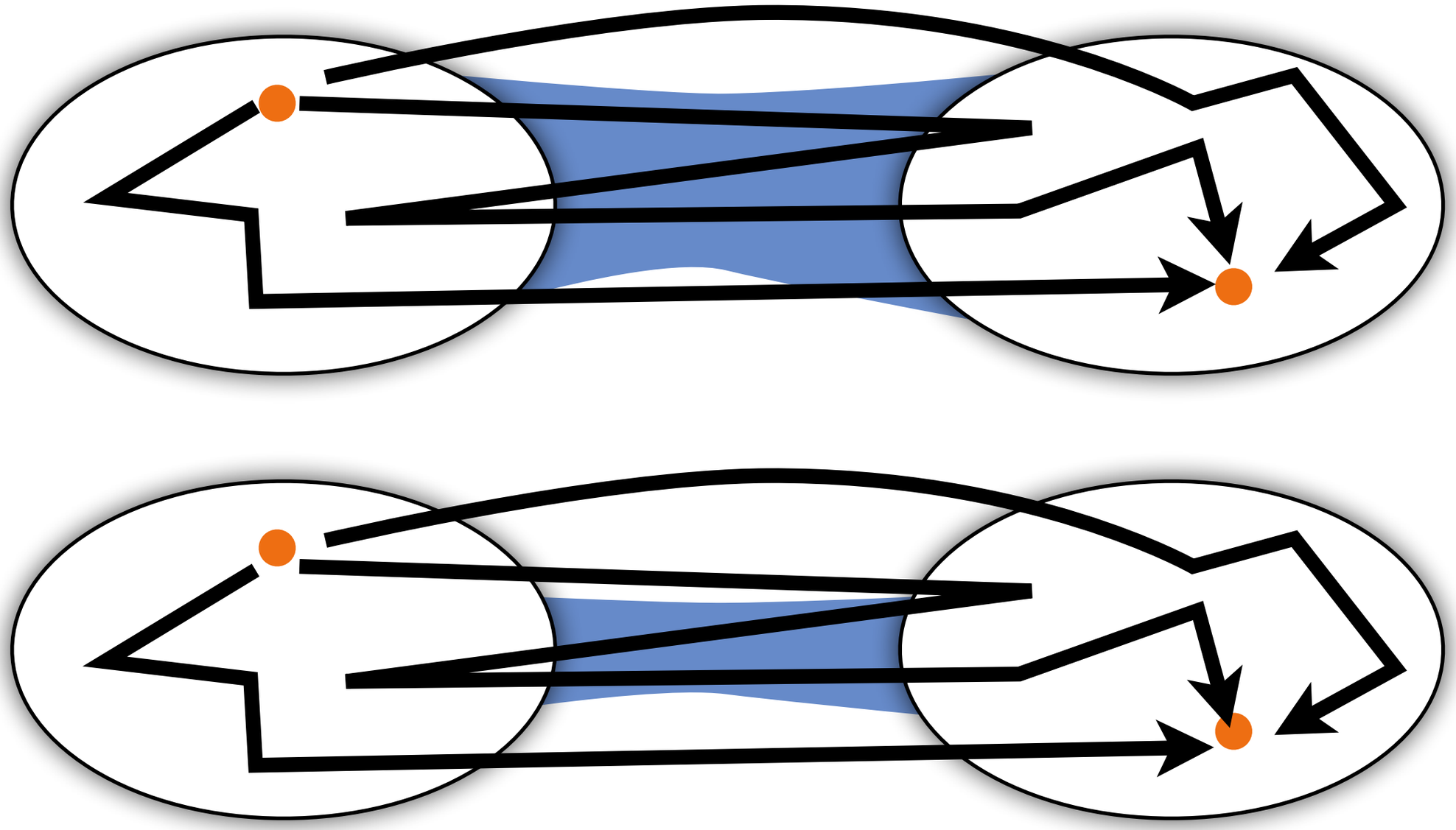
Intuition



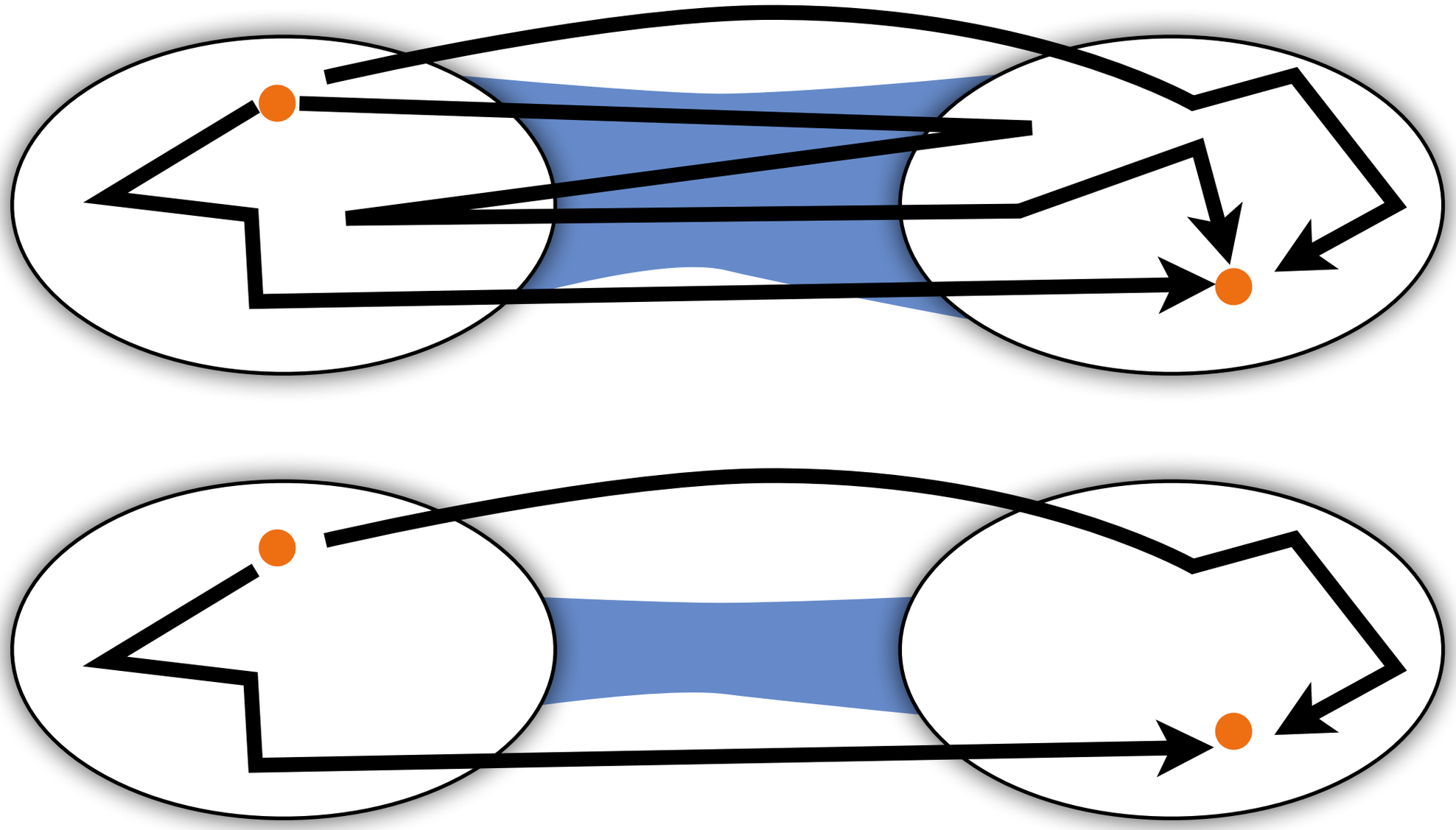
Intuition



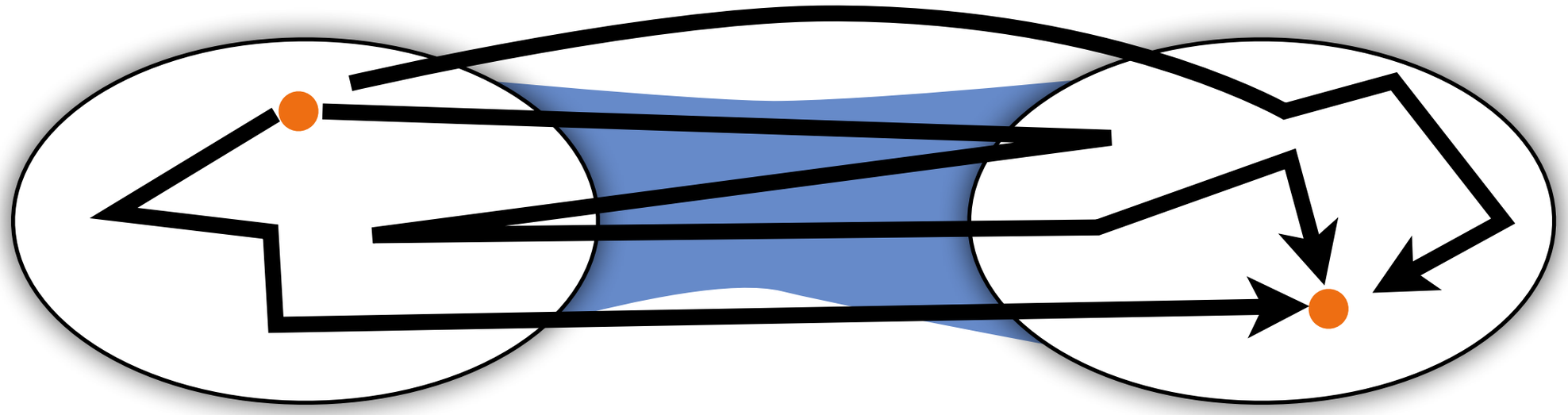
Intuition

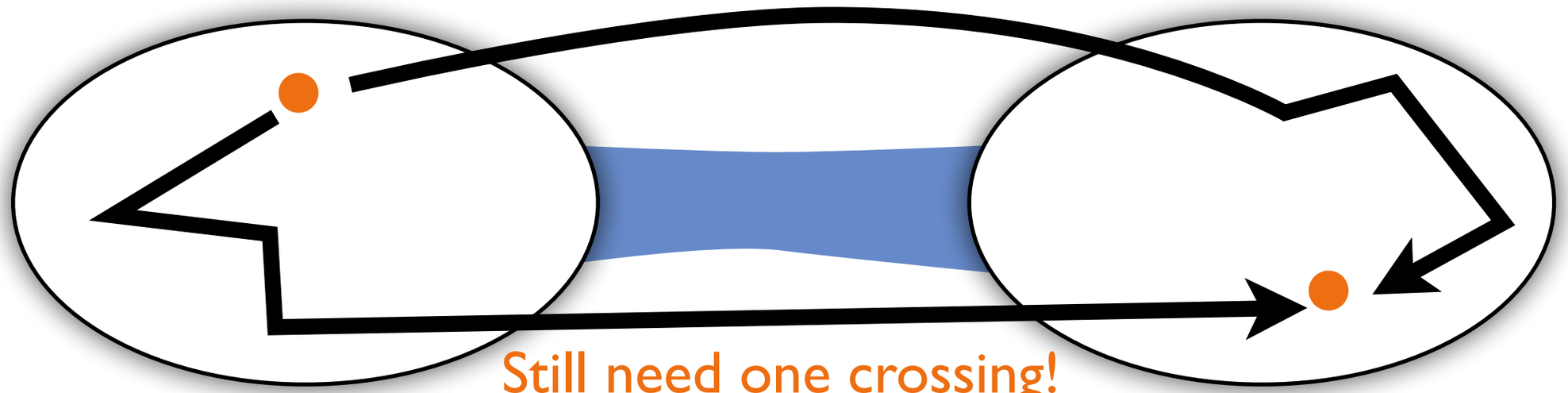


Intuition



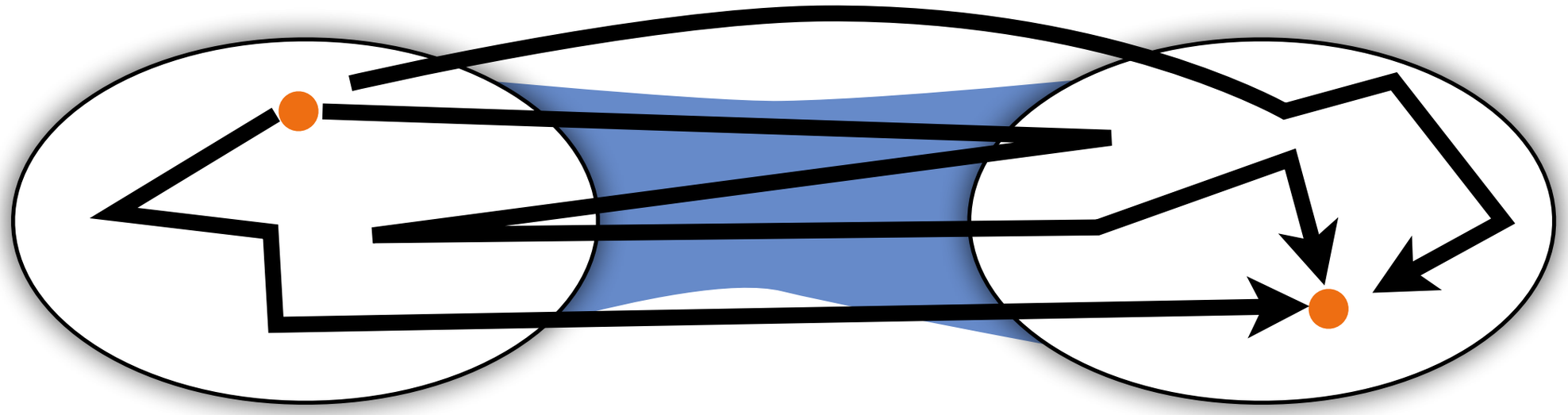
Intuition

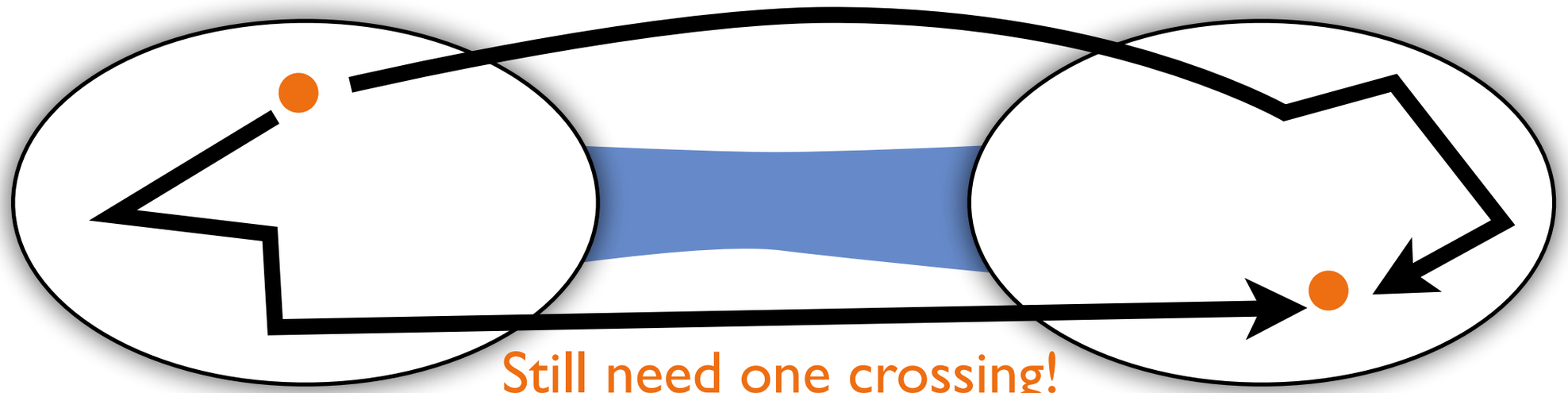




Still need one crossing!

Intuition

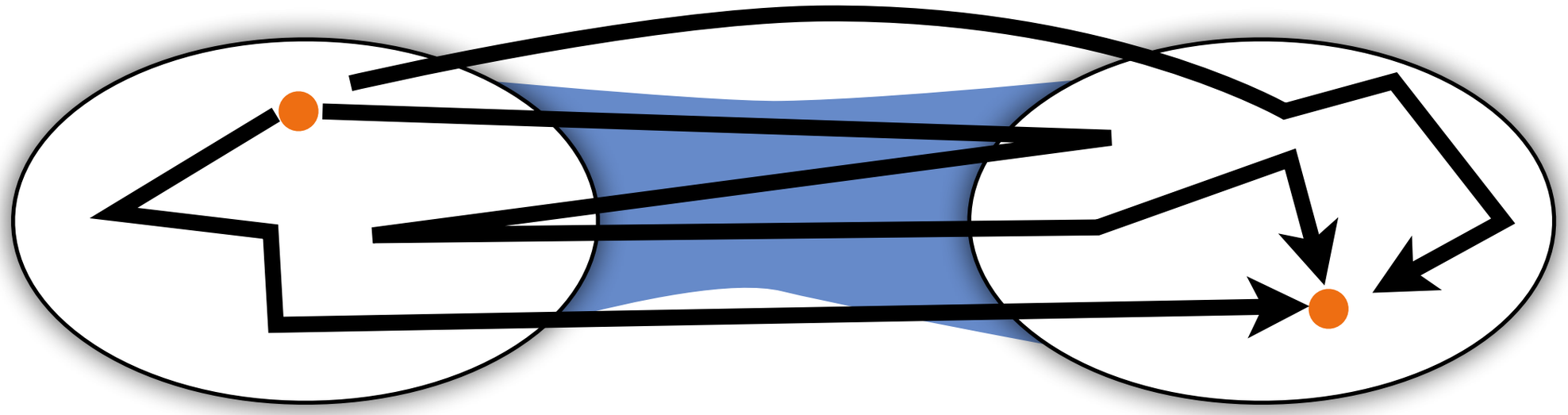


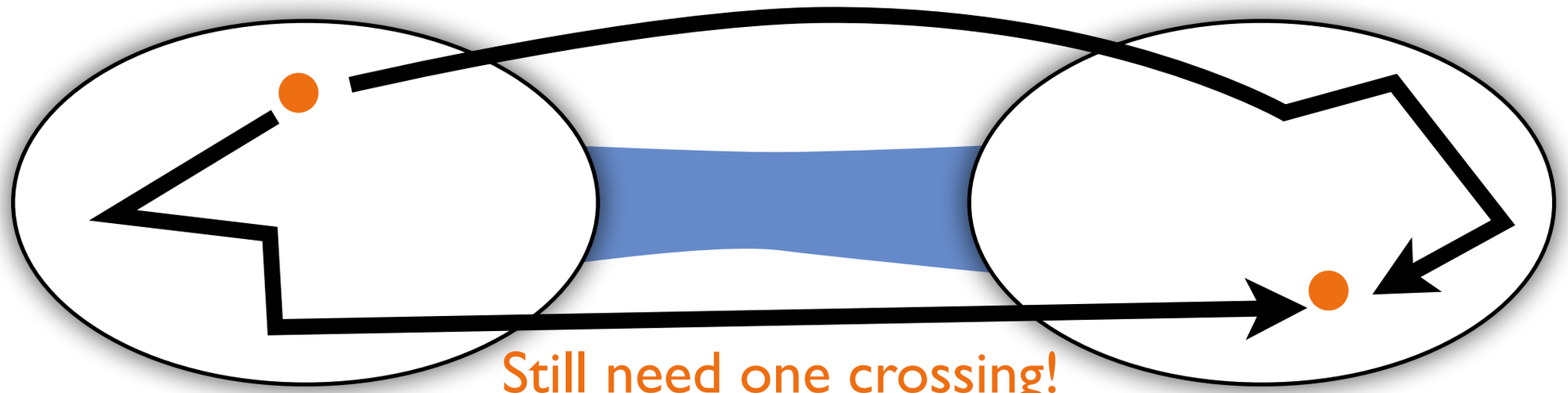


Still need one crossing!

$$\Theta\left(\frac{1}{APL}\right)$$

Intuition

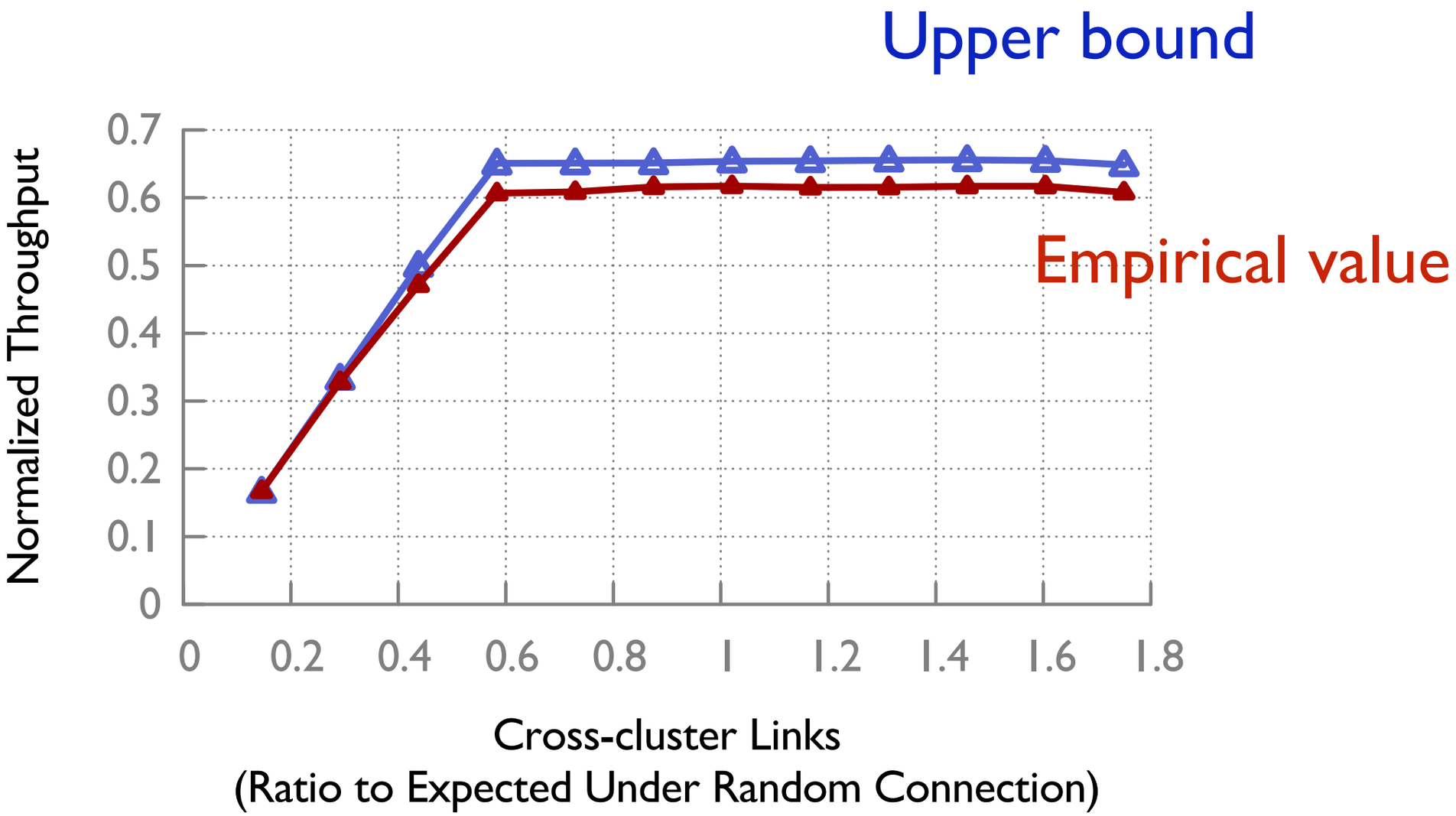




Still need one crossing!

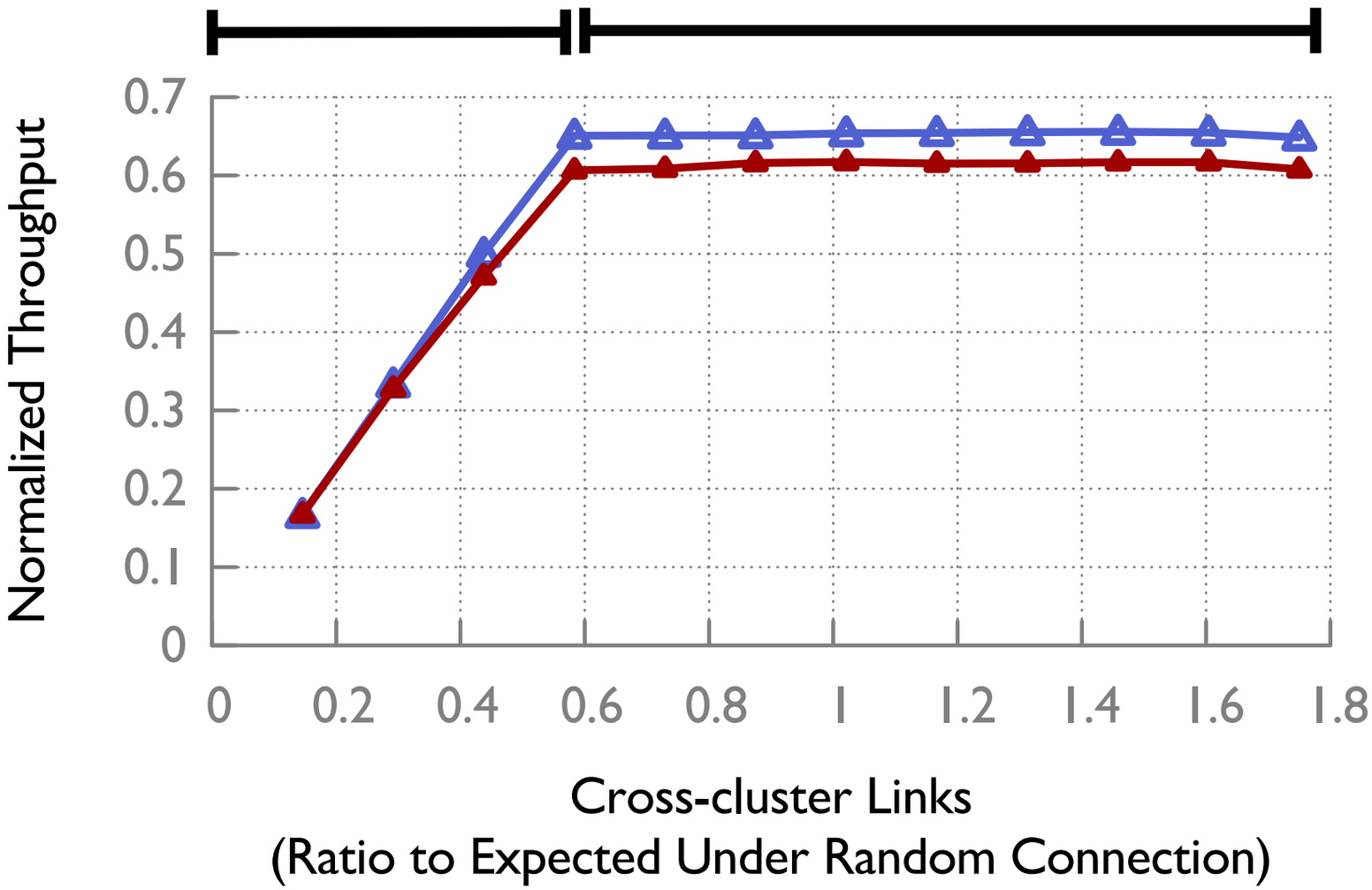
Throughput should drop when less than $\Theta\left(\frac{1}{APL}\right)$ of total capacity crosses the cut!

Explaining throughput

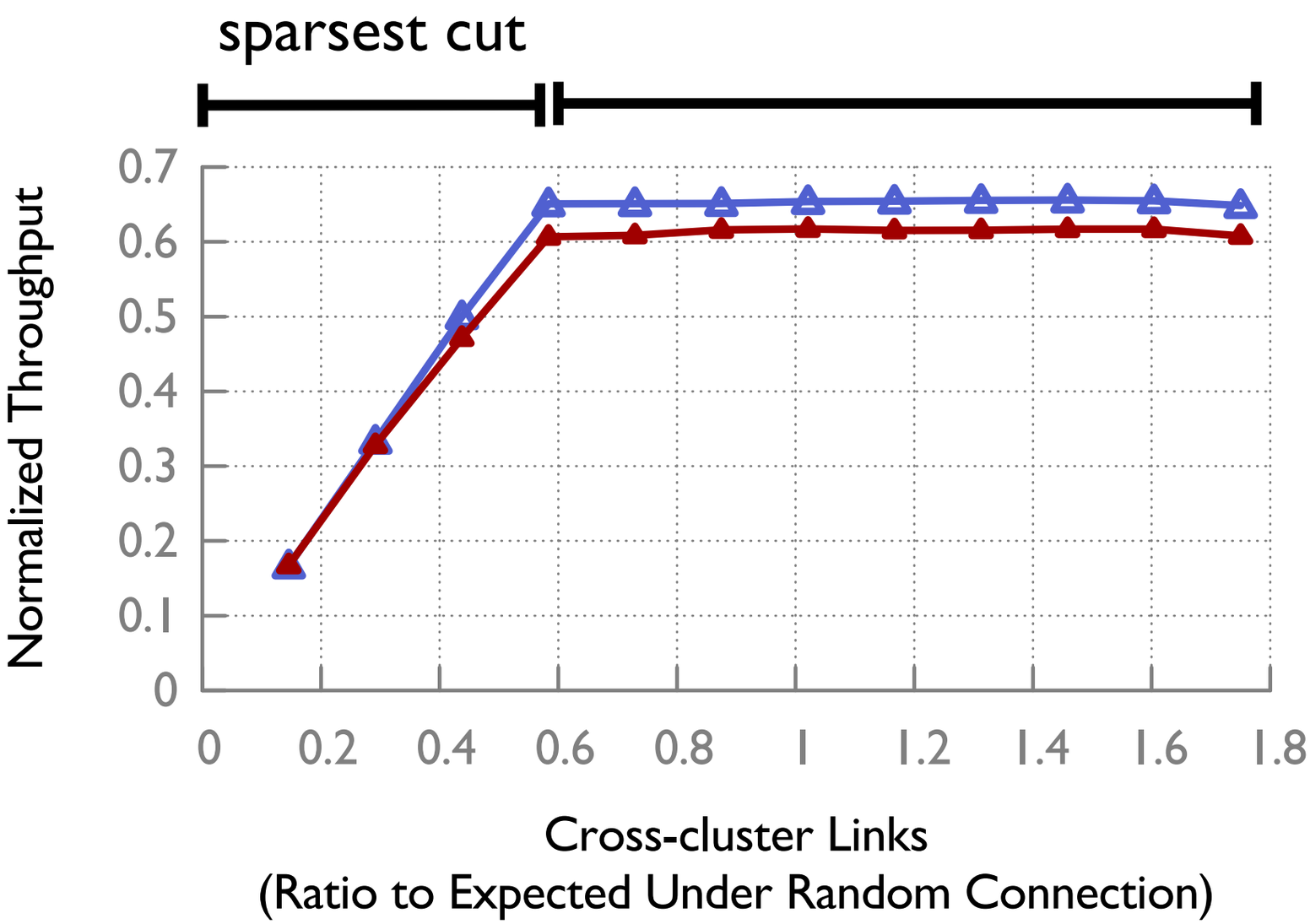


And constant-factor matching lower bounds in special case

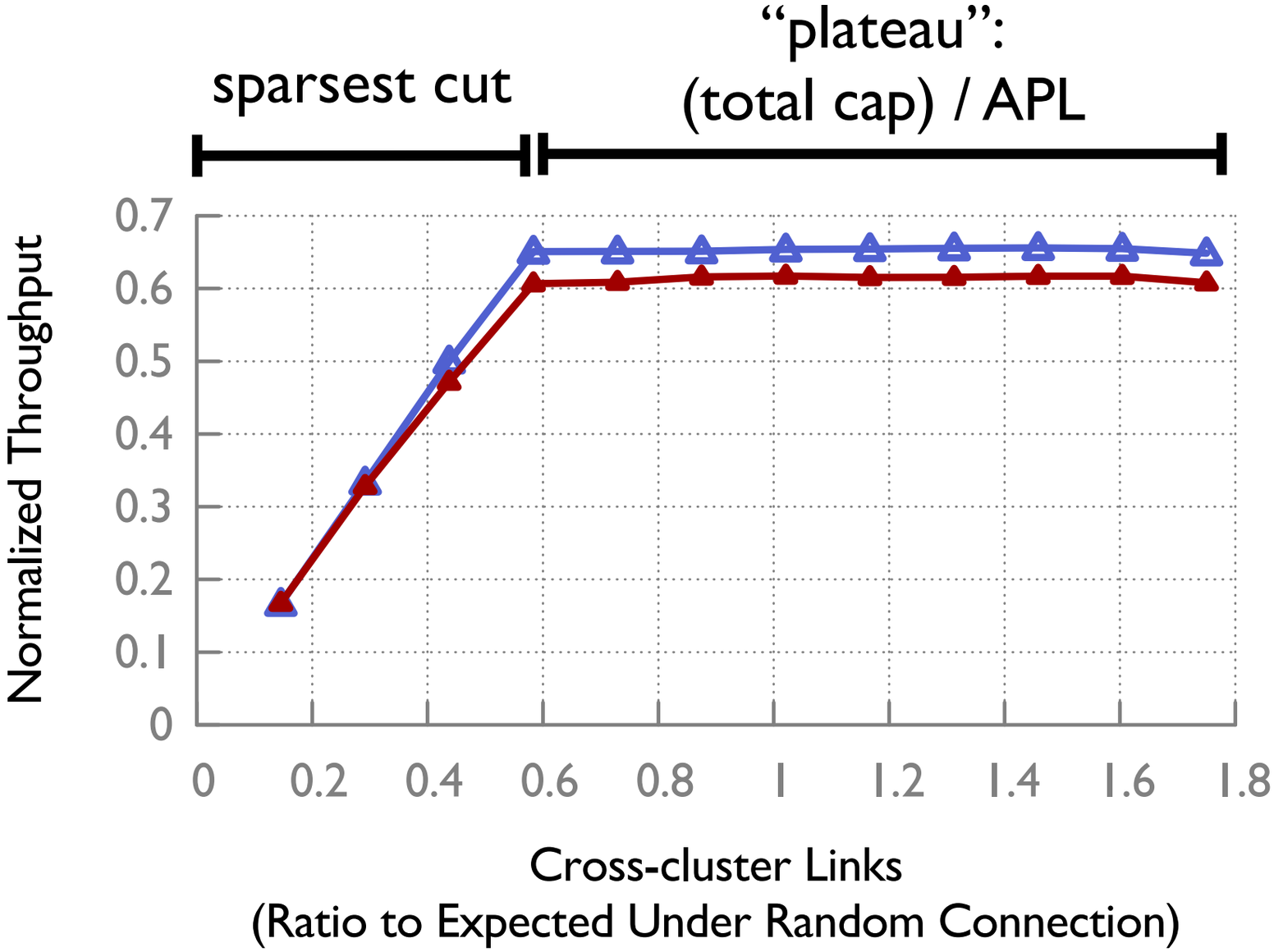
Two regimes of throughput



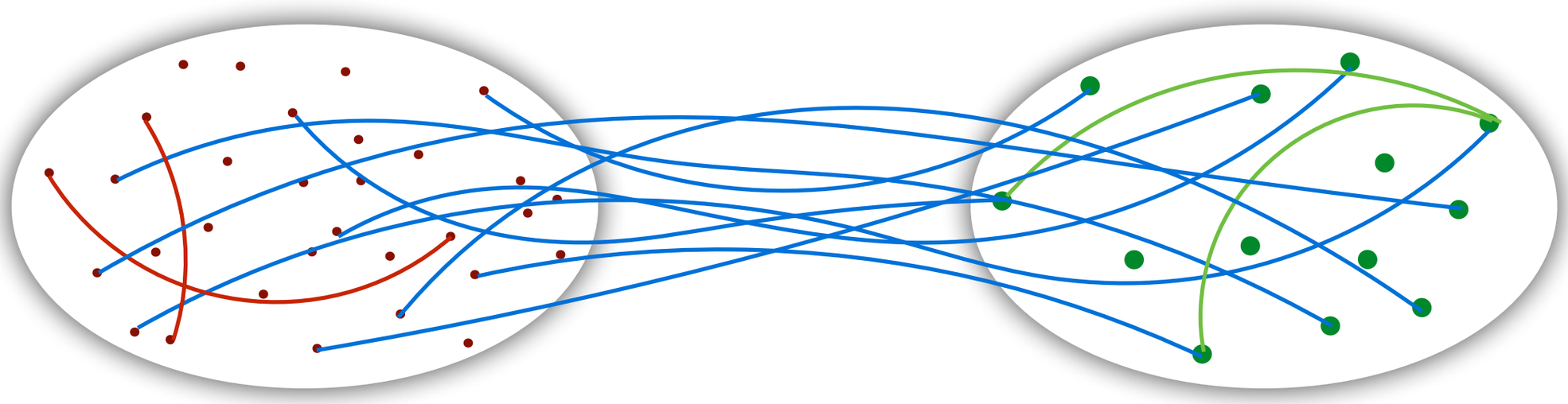
Two regimes of throughput



Two regimes of throughput

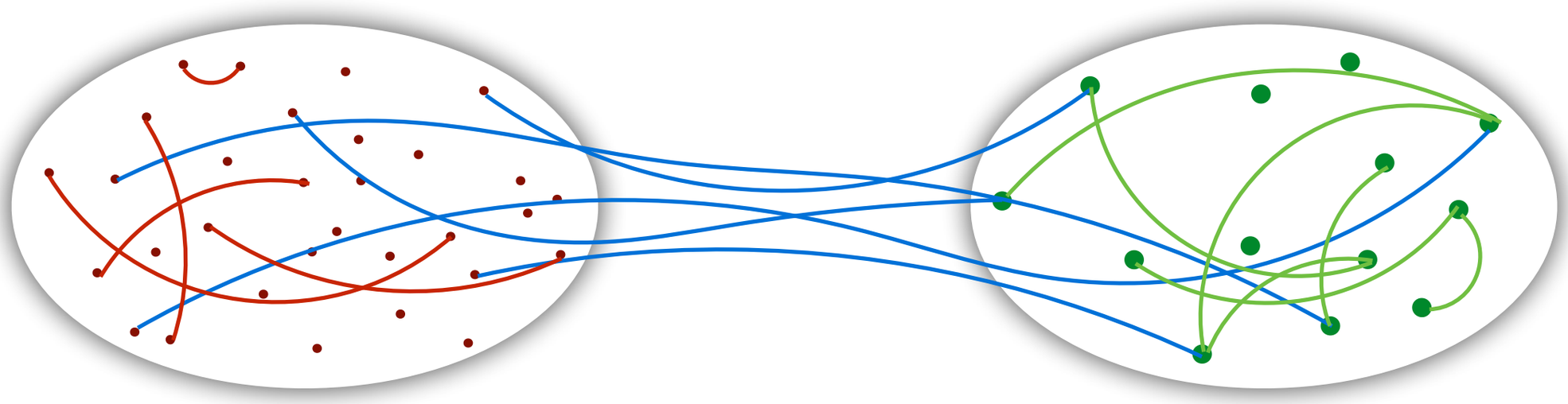


Implications



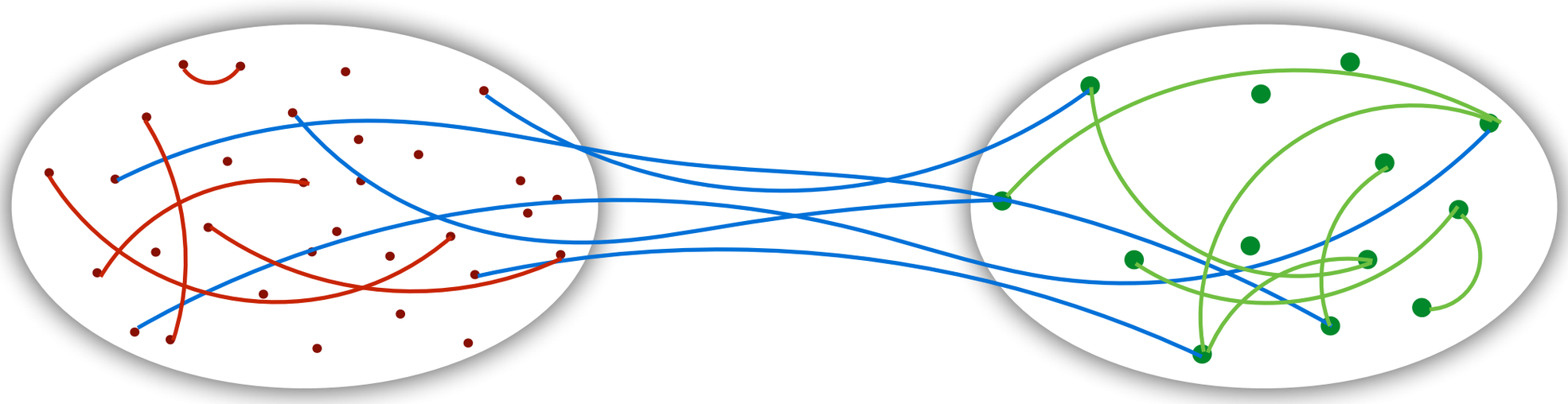
A wide range of connectivity options

Implications



A wide range of connectivity options

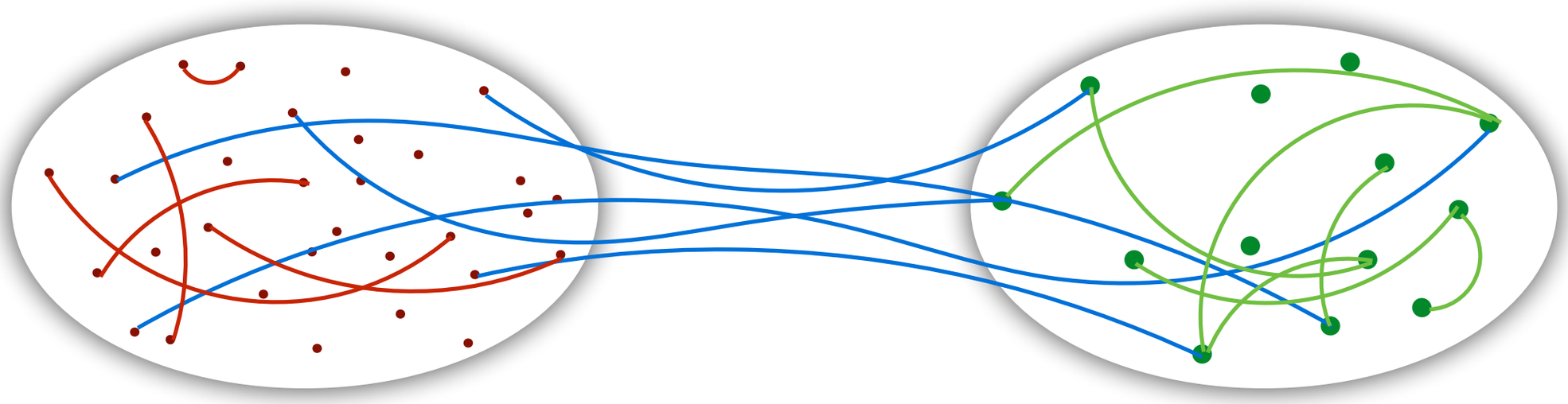
Implications



A wide range of connectivity options

Bisection bandwidth ≠ throughput

Implications



A wide range of connectivity options

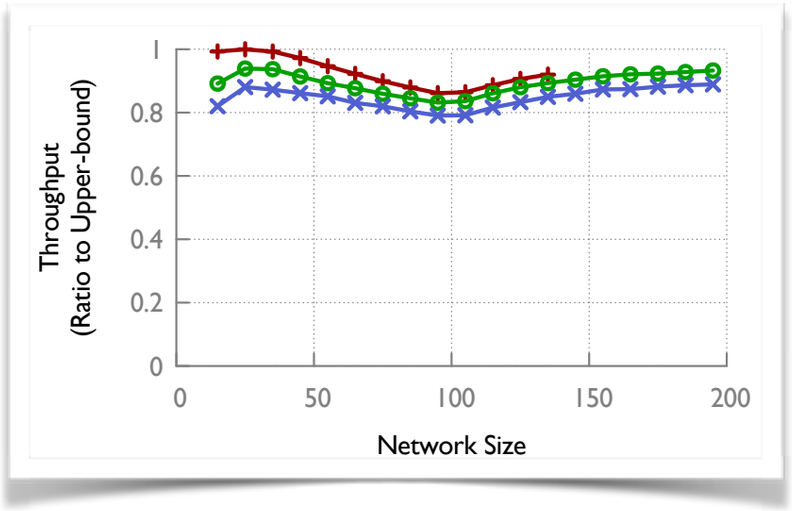
Bisection bandwidth ≠ throughput

Greater freedom in cabling

Quick recap!

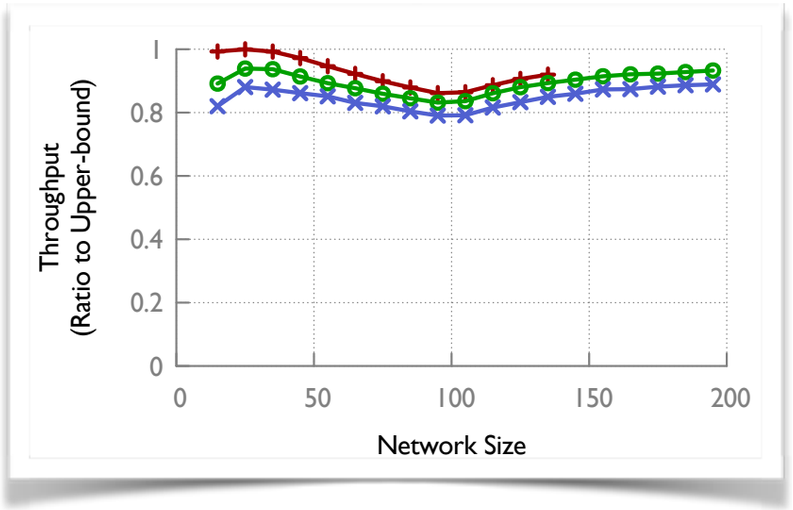
0

How close can we get to optimal network capacity?



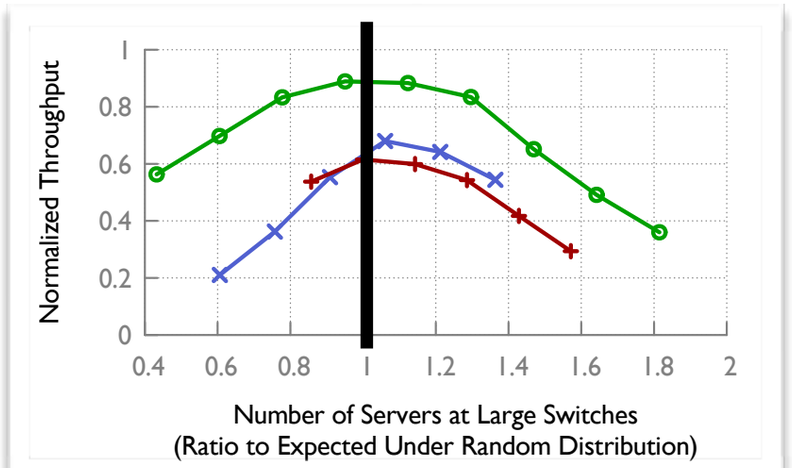
0

How close can we get to optimal network capacity?

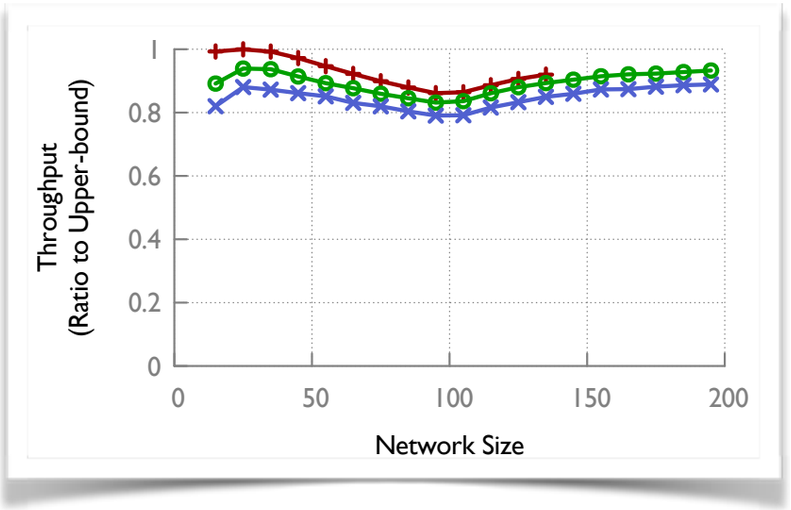


1

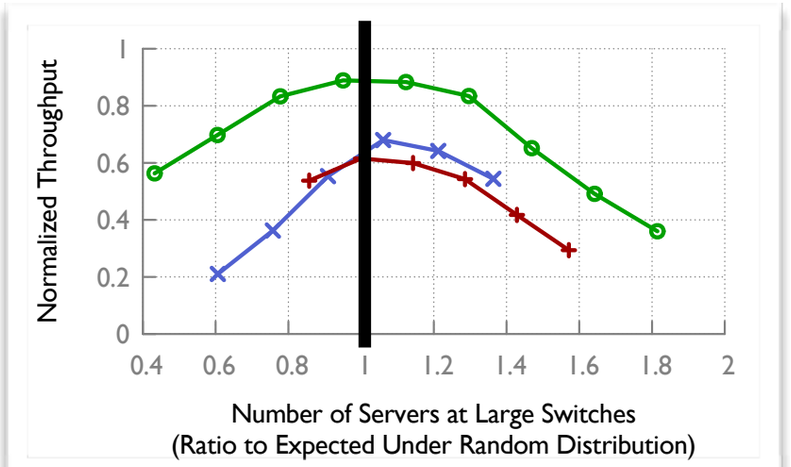
How should we distribute servers?



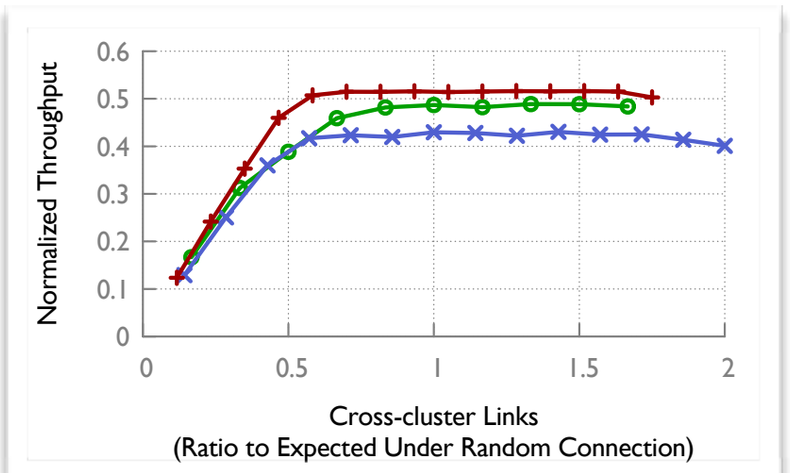
0 How close can we get to optimal network capacity?



1 How should we distribute servers?



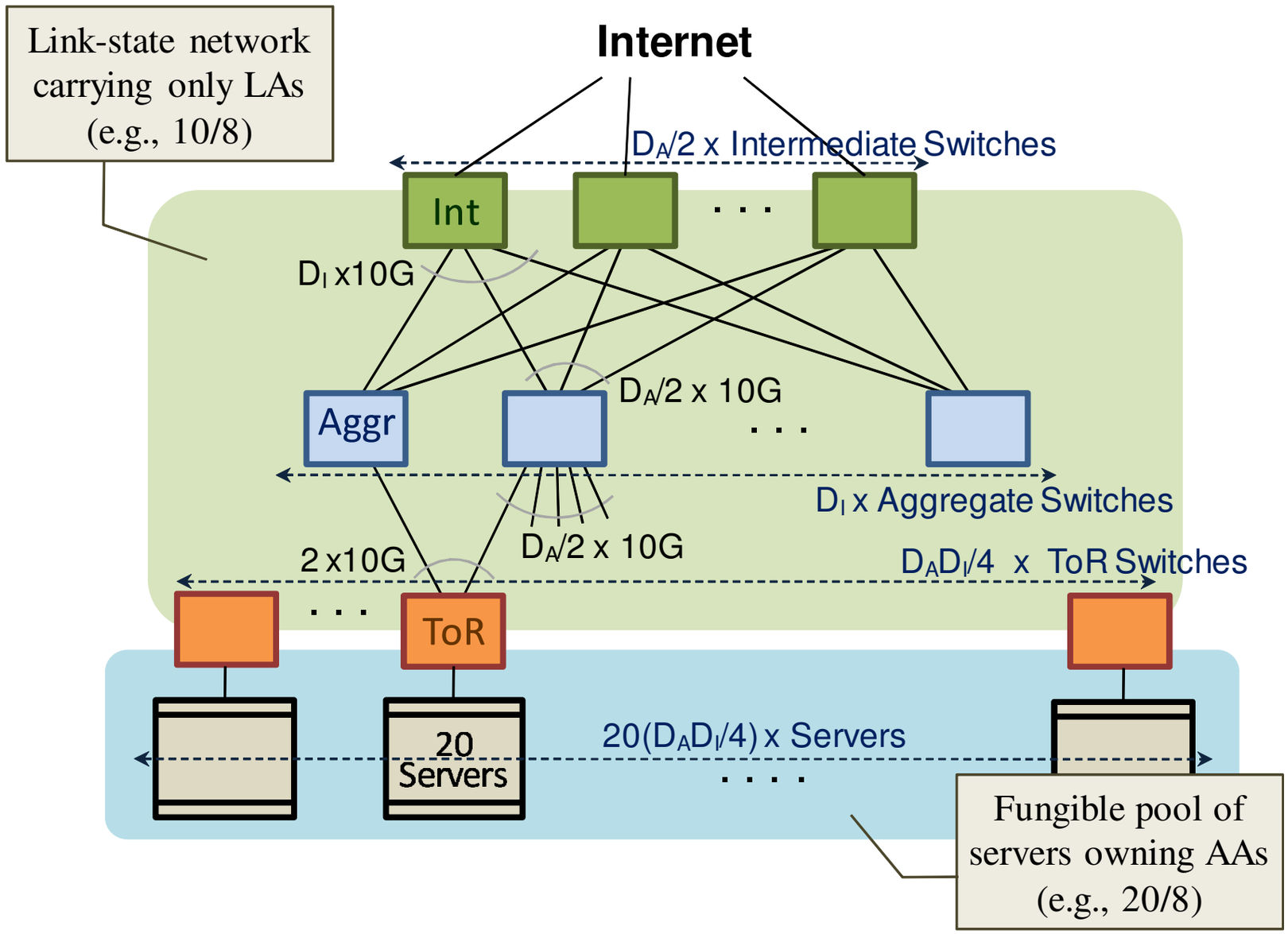
2 How should we interconnect switches?



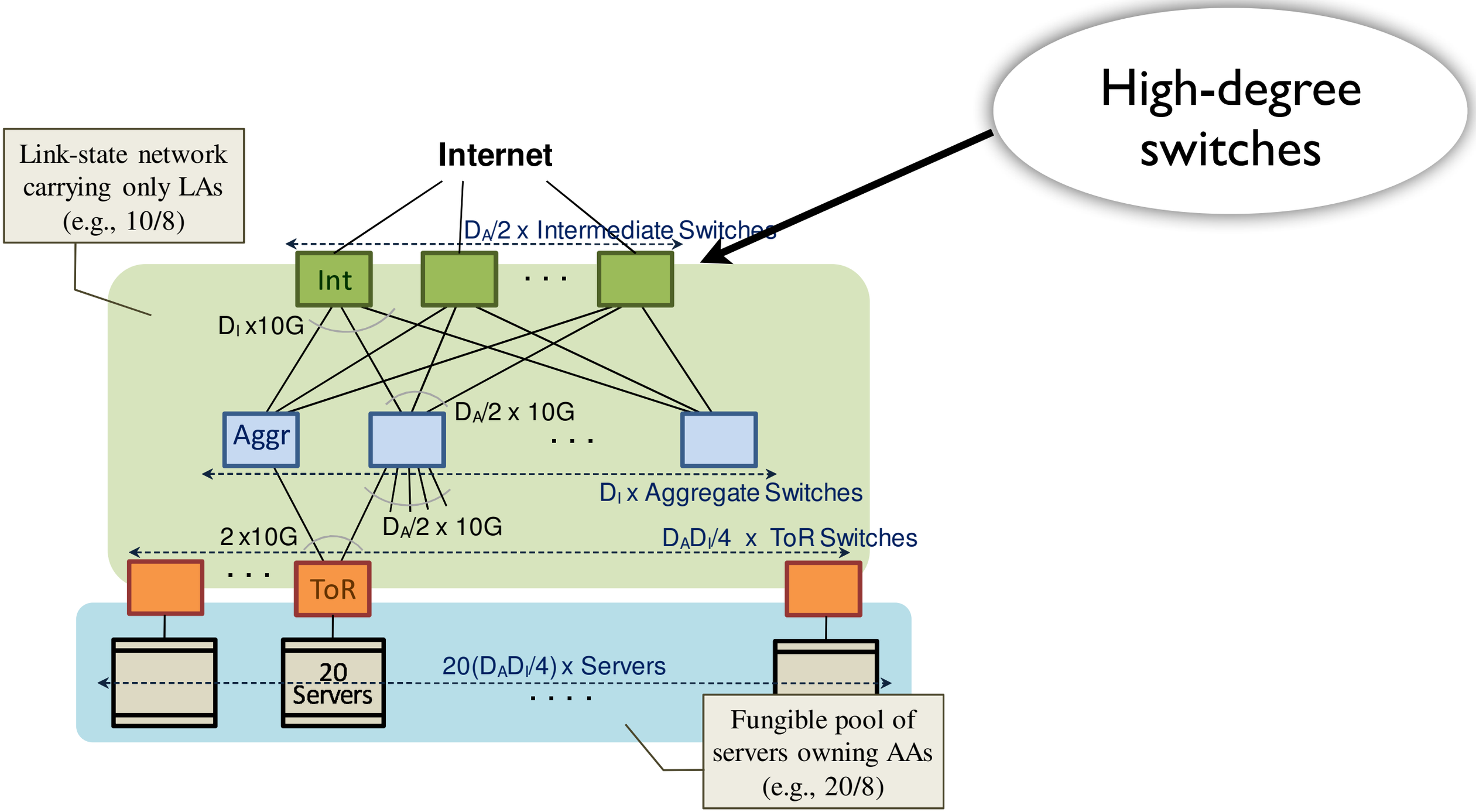
Improving a REAL heterogeneous topology

The VL2 topology

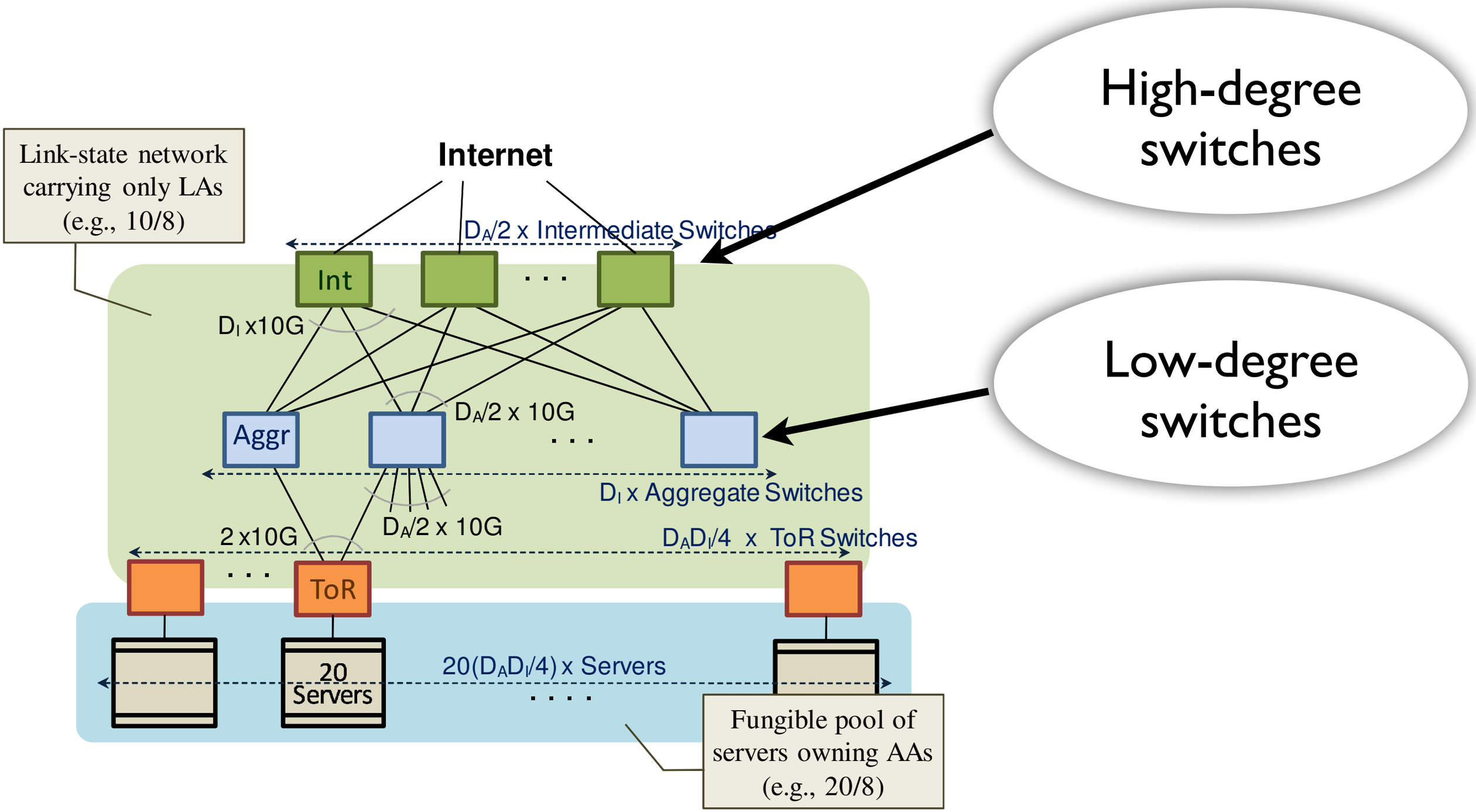
[Greenburg, Hamilton, Jain, Kandula, Kim, Lahiri, Maltz, Patel, Sengupta, SIGCOMM'09]



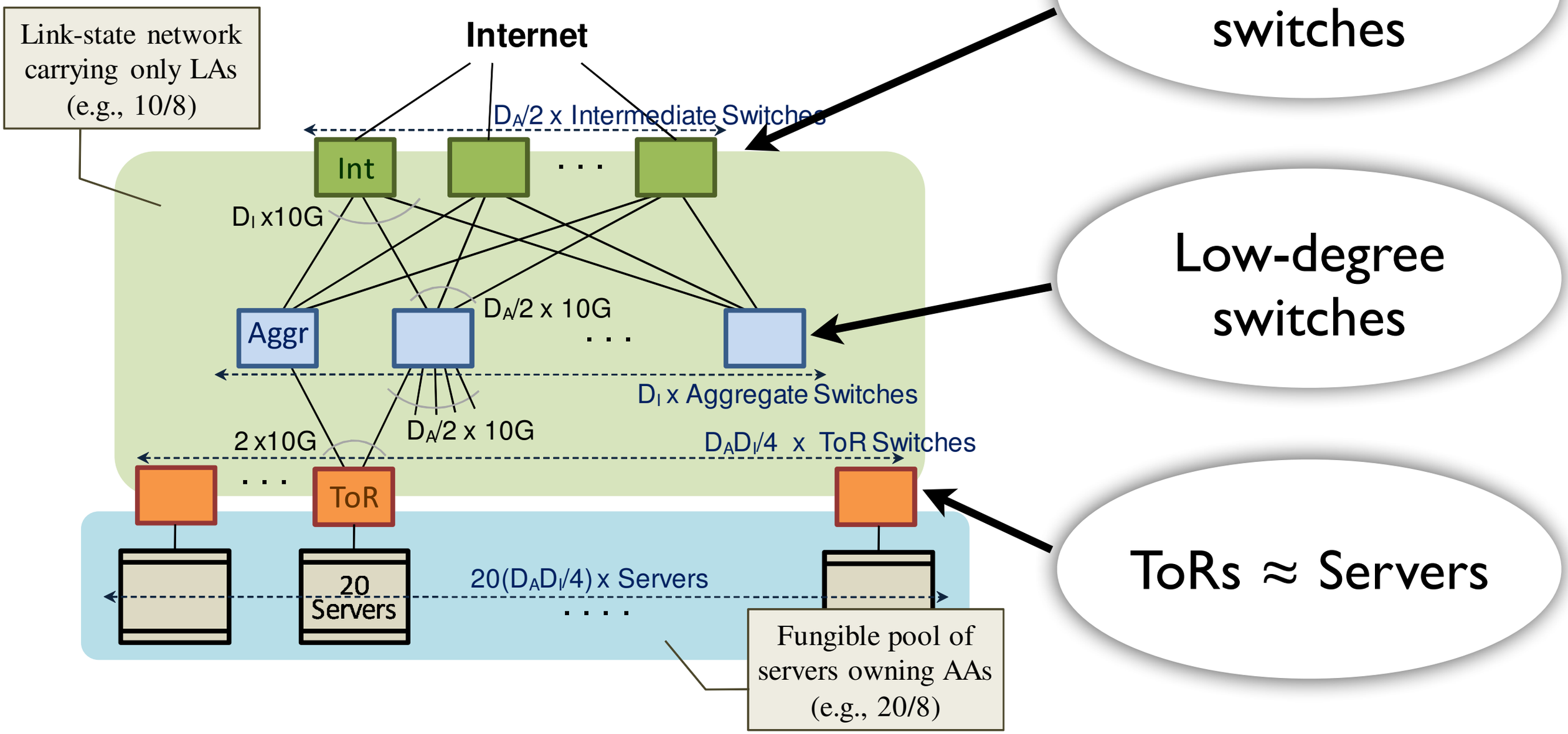
The VL2 topology



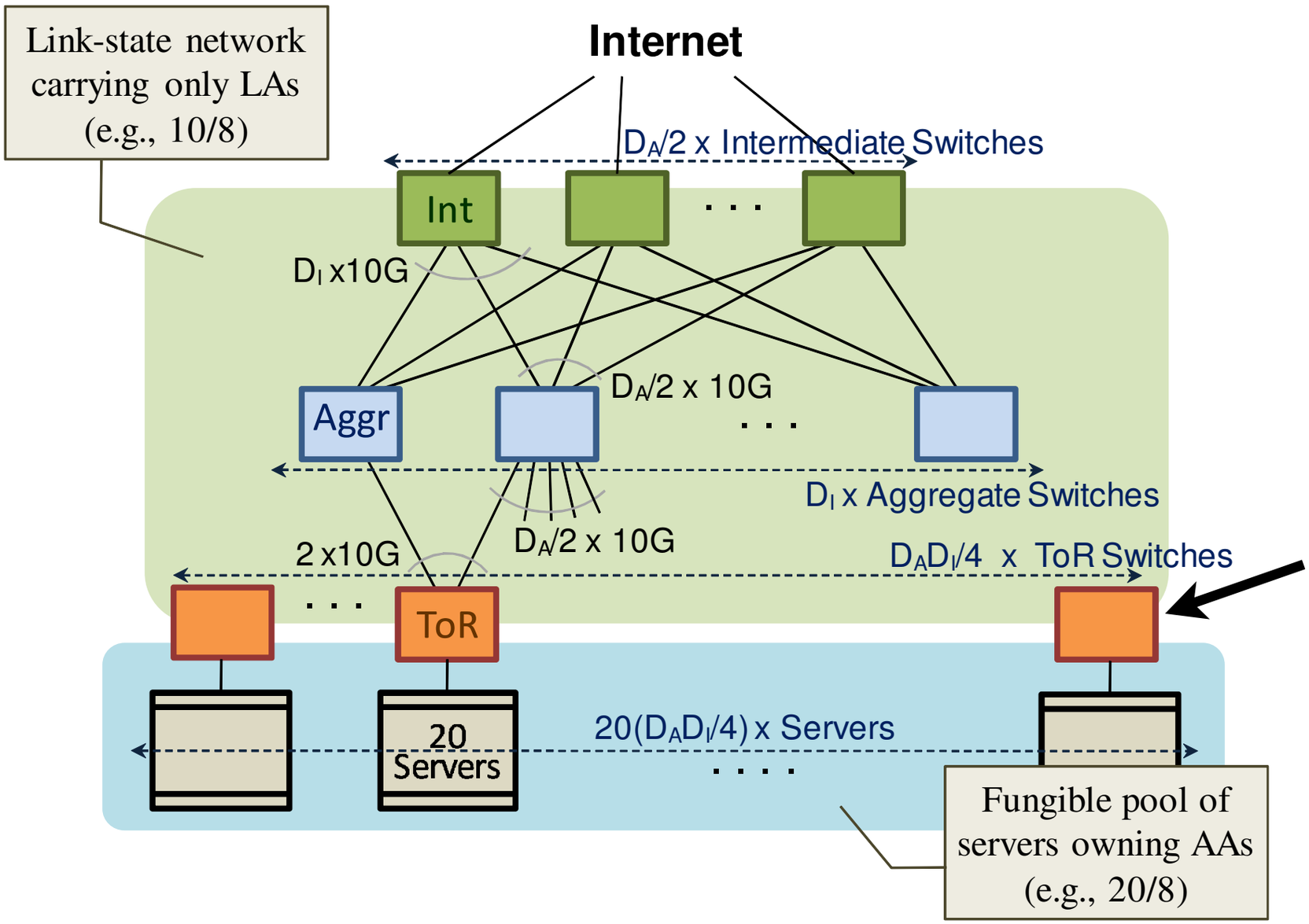
The VL2 topology



The VL2 topology

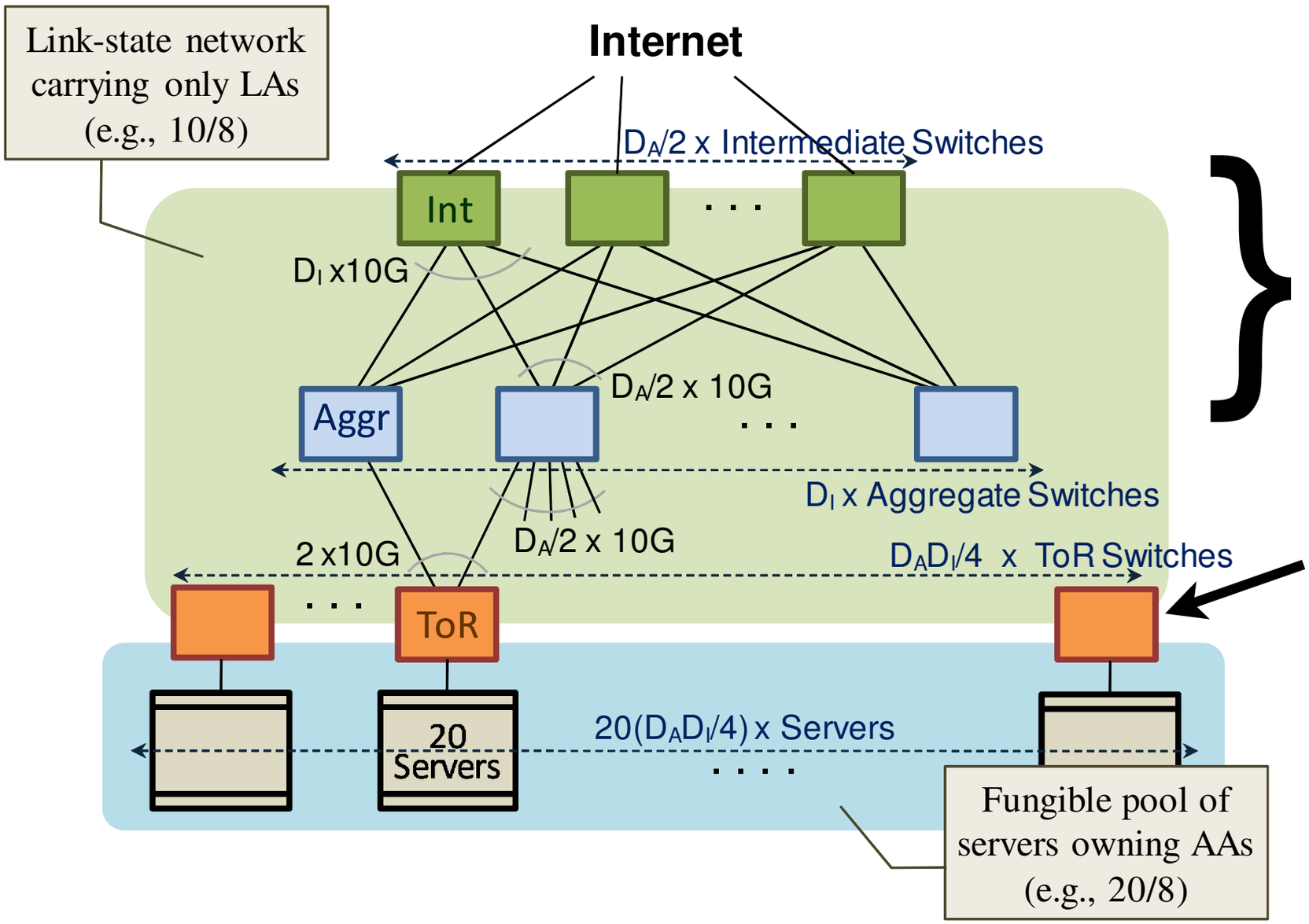


Rewiring VL2



Connect ToRs proportional to Intermediate/Agg degree

Rewiring VL2



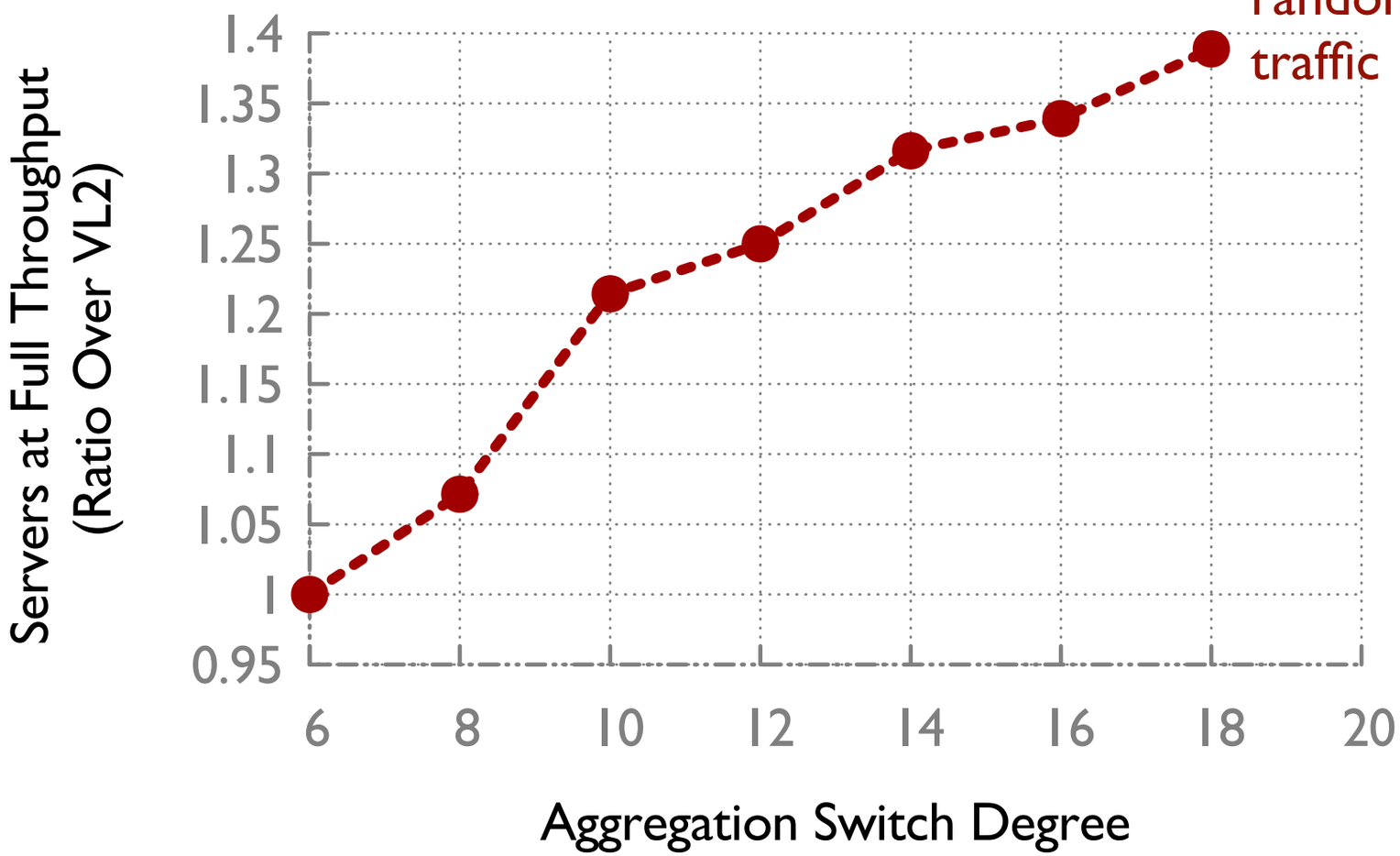
Uniform-random interconnection

Connect ToRs proportional to Intermediate/Agg degree

Rewiring VL2

40% more servers

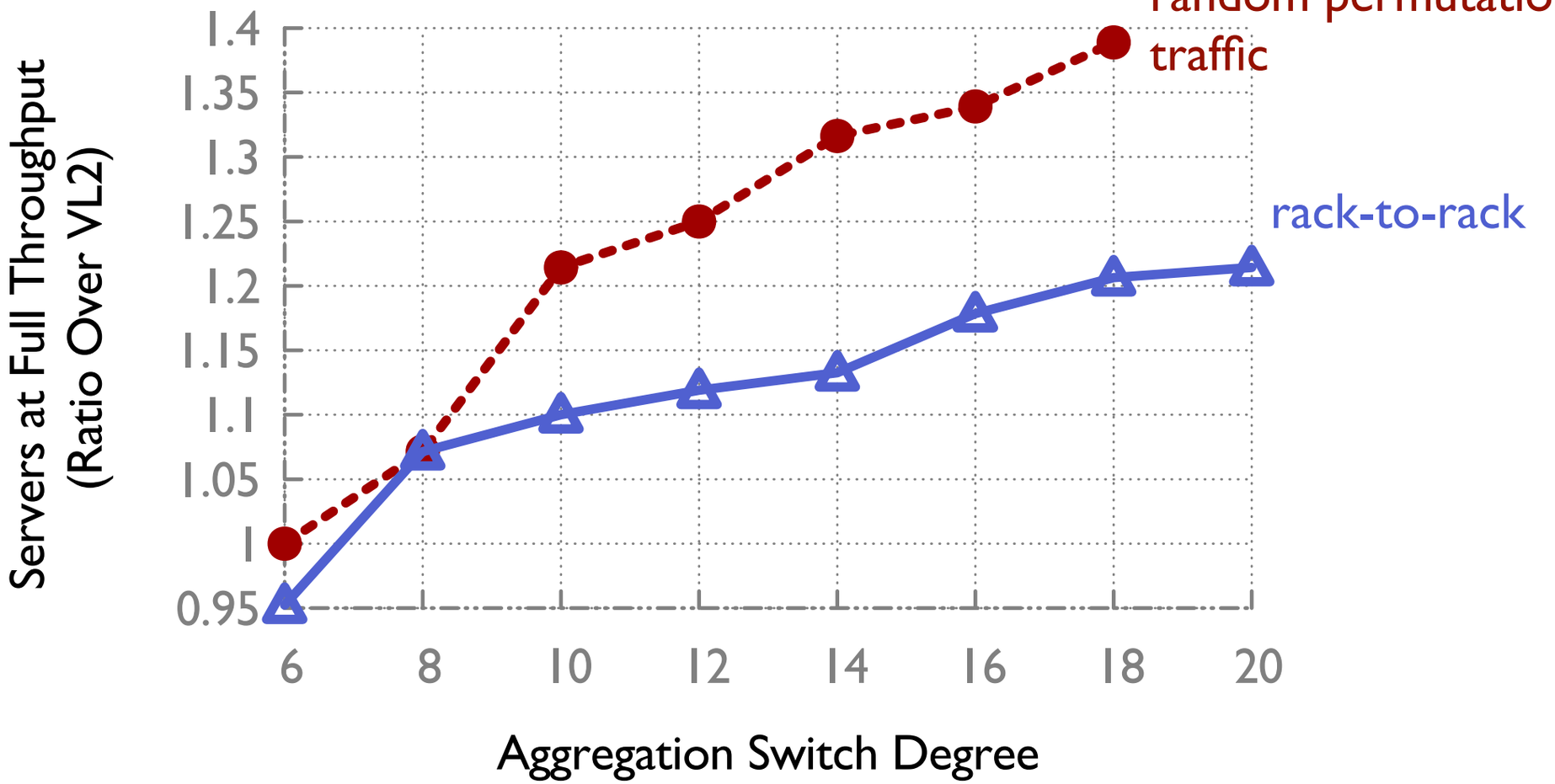
with server-to-server random permutation traffic



Rewiring VL2

40% more servers

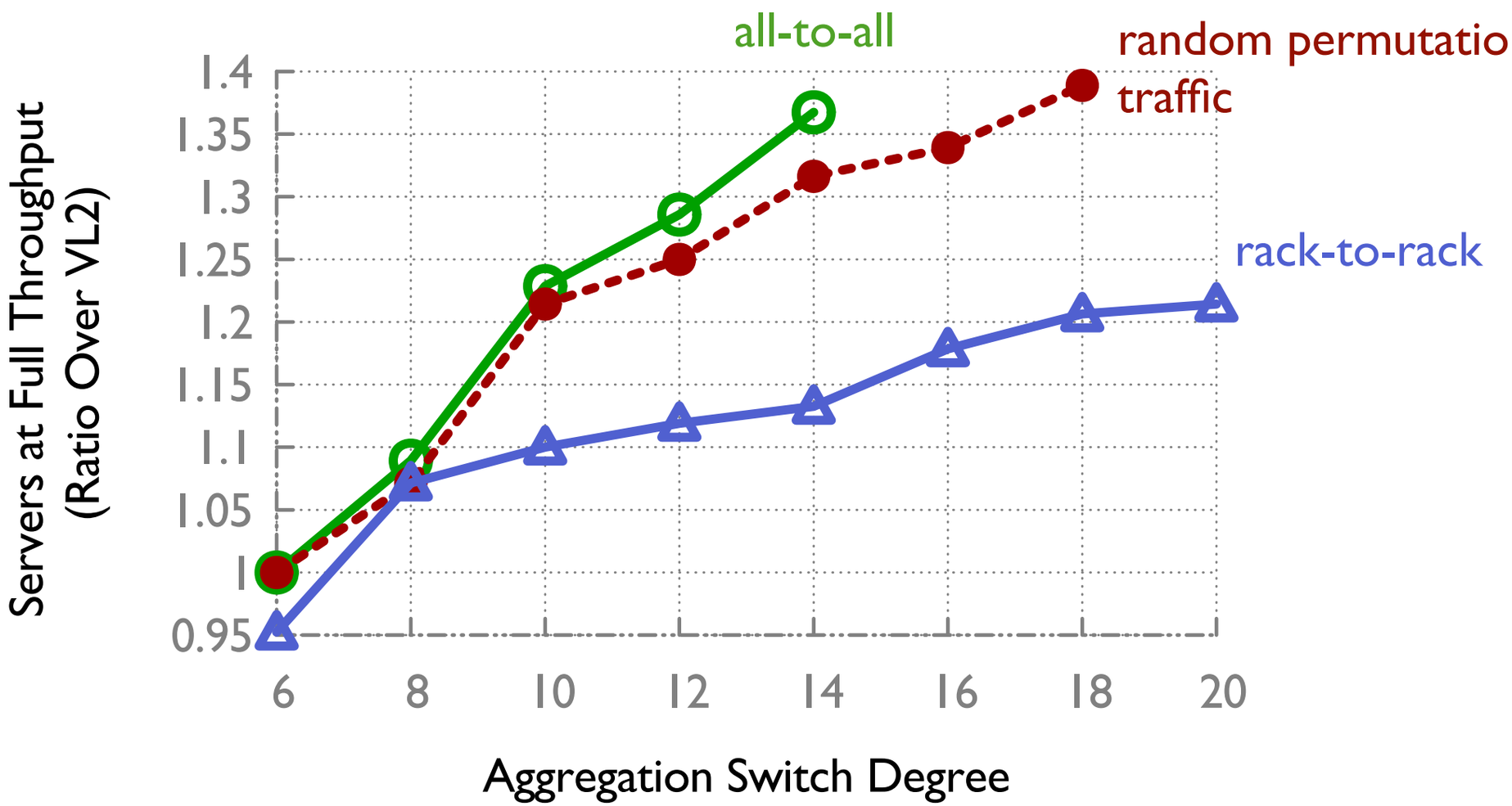
with server-to-server random permutation traffic



Rewiring VL2

40% more servers

with server-to-server random permutation traffic



How do we design throughput optimal network topologies?

https://github.com/ankitsingla/topobench