Towards Optimal-Performance Datacenters

HotNets'15 – Xpander: Unveiling the Secrets of High-Performance Datacenters
Asaf Valadarsky [3], Michael Dinitz[1], Michael Schapira[3]

CoNext'16 – Xpander: Towards Optimal-Performance Datacenters

Asaf Valadarsky [3] , Michael Dinitz[1], Gal Shahaf[3], Michael Schapira[3]

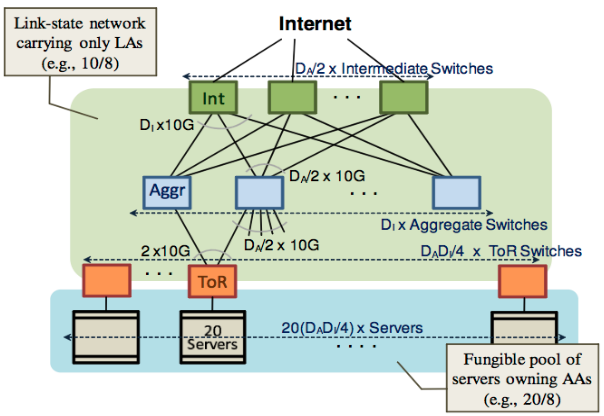
SIGCOMM'17 – Beyond Fat-Trees Without Antennae, Mirrors, and Disco-Balls
Simon Kassing[2], **Asaf Valadarsky** [3], Gal Shahaf[3], Michael Schapira[3], Ankit Singla[2]

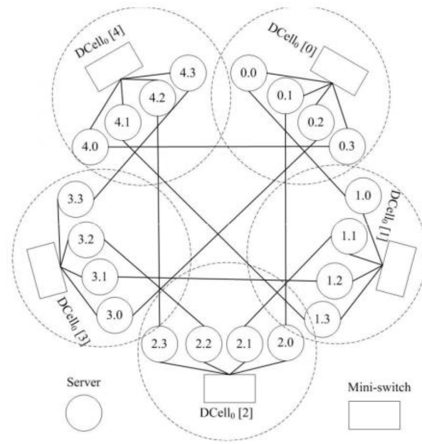
JOHNS HOPKINS UNIVERSITY
1

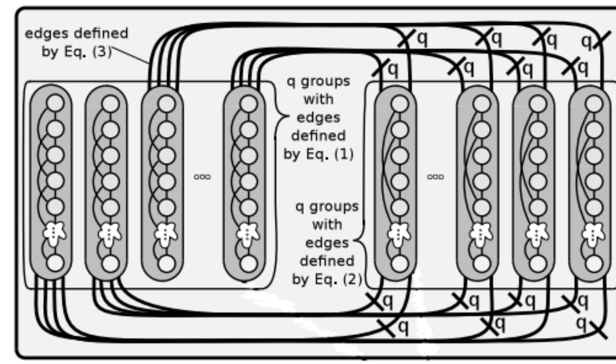
ETH Zürich
2

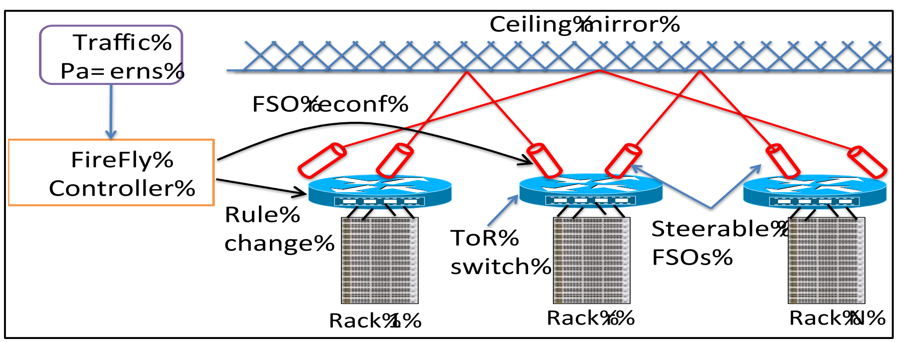
האוניברסיטה העברית בירושלים
THE HEBREW UNIVERSITY OF JERUSALEM
3

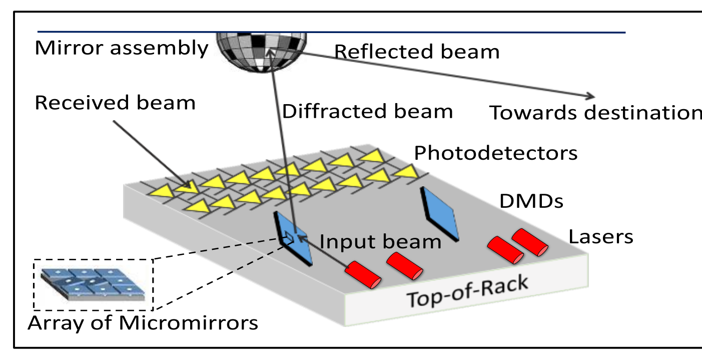
Designing A Datacenter Architecture











Network topology? Routing? Congestion Control?

Designing A Datacenter Architecture

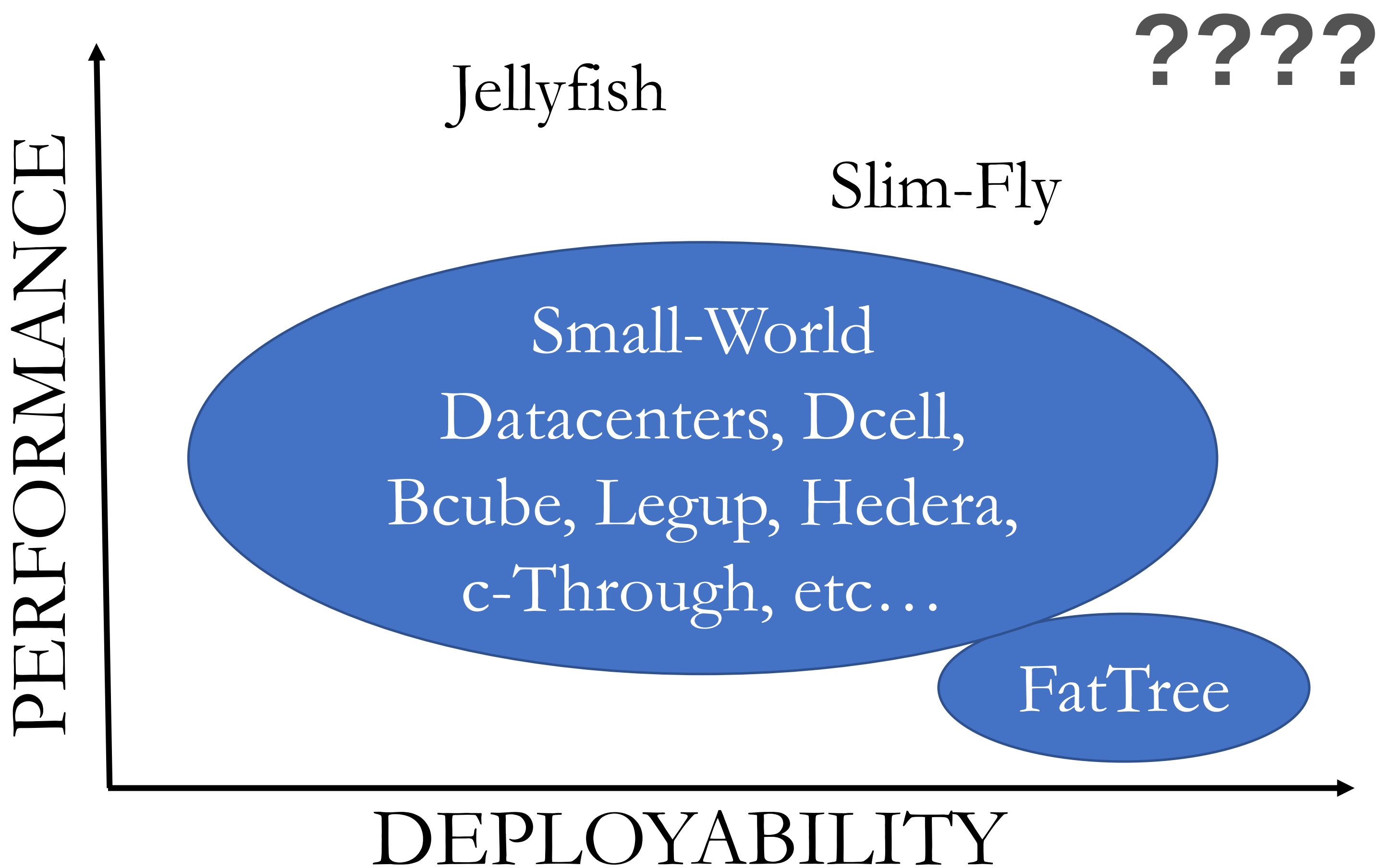
Performance

- ➡Throughput
- ➡Resiliency to failures
- ➡Path diversity
- ➡Flow completion time
- ➡…

Deployability

- ➡Cabling complexity
- ➡Operations cost
- ➡Equipment costs
- ➡"Easy to reason about"
- ➡…

What Is The "RIGHT" Datacenter Architecture?



In This (and the next) Talk

- Reaching that upper-right corner entails designing "expander datacenters"

- **Xpander:** a tangible and near-optimal datacenter design

- **Next talk:** Theoretical advances in the field of expander datacenters

Expander Datacenters

- An expander datacenter architecture:

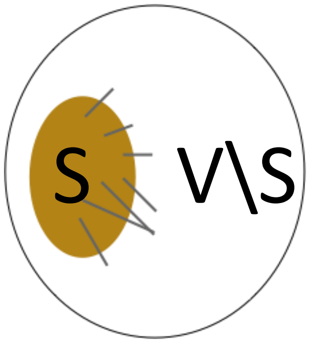
  ➡Utilizes an expander graph as its network topology (*see next slide + Michael's talk*)

  ➡Employs multi-path routing to exploit path diversity

Expander Graphs: Intuition

- A graph is called an "expander graph" if it has "good" edge expansion

$$\min_{S \subset V, 0 < |S| \leq \frac{n}{2}} \frac{EdgesBetween(S, V \setminus S)}{|S|}$$



- **Intuition:** In a d-regular graph, with constant edge expansion *c*, there are at least |S|*c* links crossing any cut (S,V\S)
  - ➡ We want high values of *c* (ideally ~d/2)
  - ➡ Traffic is never bottlenecked at small set of links
  - ➡ Many paths between any source/destination pairs

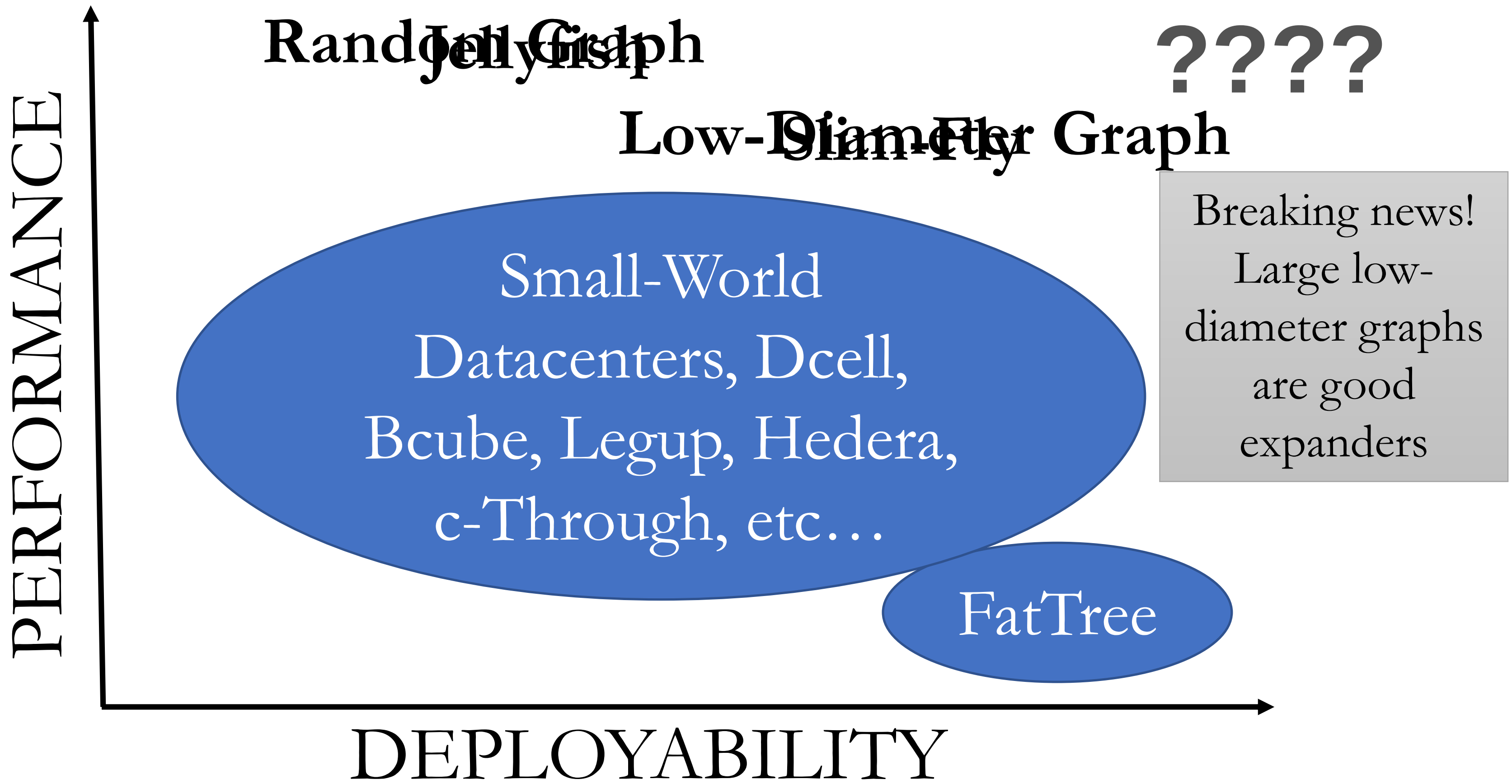
Expander Datacenters Achieve Near-Optimal Performance

- ➞ Support higher traffic loads
- ➞ More resilient to failures
- ➞ Support more servers with less network devices
- ➞ Multiple short-paths between hosts
- ➞ Incrementally expandable

Our Evaluation

- Theoretical analyses
- Flow- and packet-level simulations
- Experiments on a network emulator
- Experiments on an SDN-capable network

Expander Datacenters **ARE** The State-Of-The-Art Datacenters



CAN WE HAVE IT ALL?

A well structured design + Near optimal performance

YES! :)

Designing A Datacenter Architecture

<u>Performance</u>

- Throughput
- Re… …ilures
- F…
- Flow… …ion time
- …

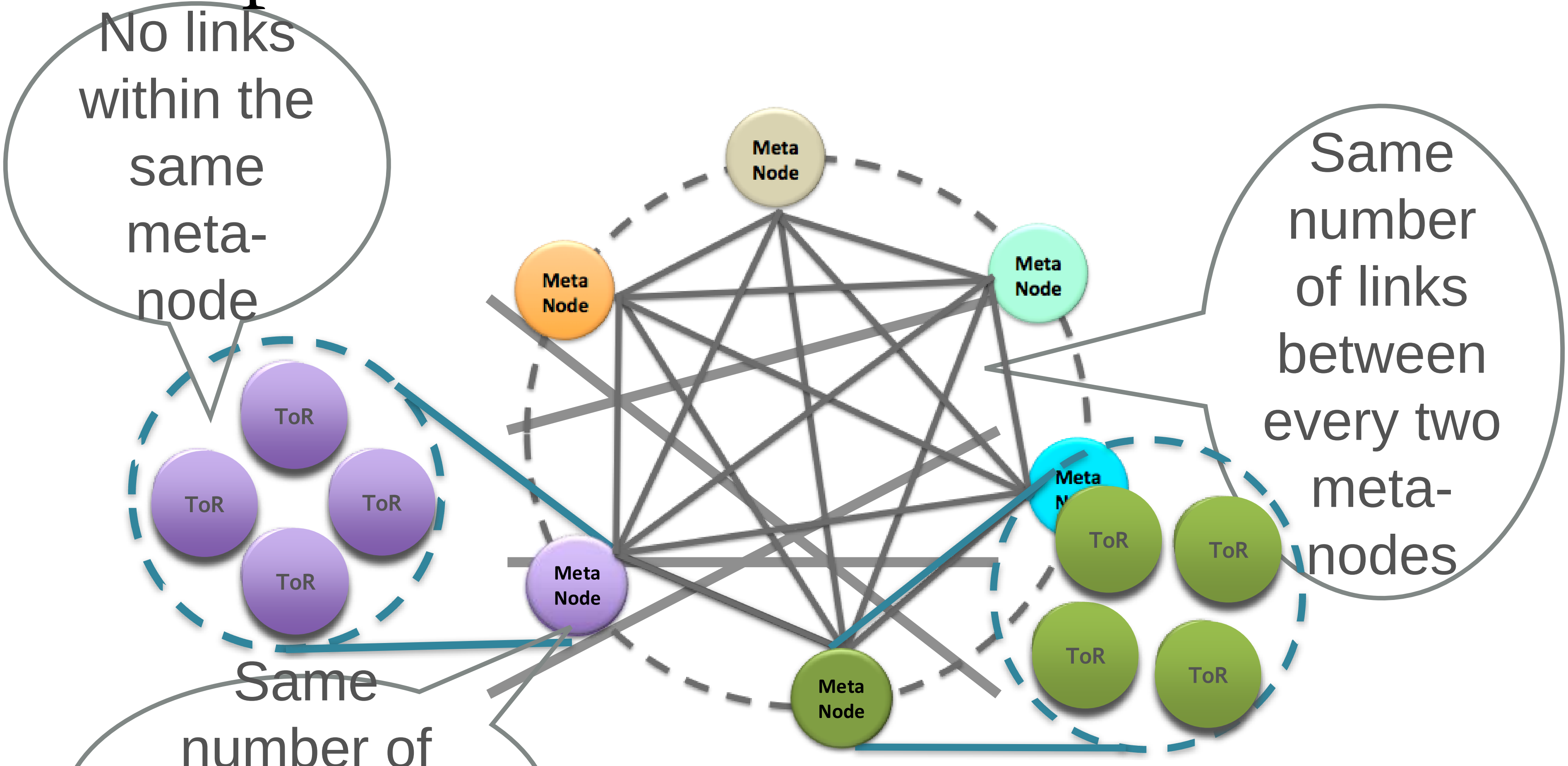
Expander Datacenter

<u>Deployability</u>

- Cabling complexity
- Op…
- E…
- "Cou… …about"
- …

Deployment-Oriented Construction

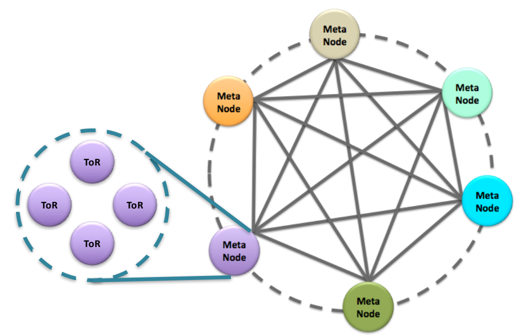
Xpander Datacenter Architecture



Leverages a **deterministic** graph-theoretic construction of expanders [BL ’06]

Xpander Datacenter Architecture

Topology



Routing

K-Shortest Paths

Congestion Control

DCTCP [SIGCOMM'10]

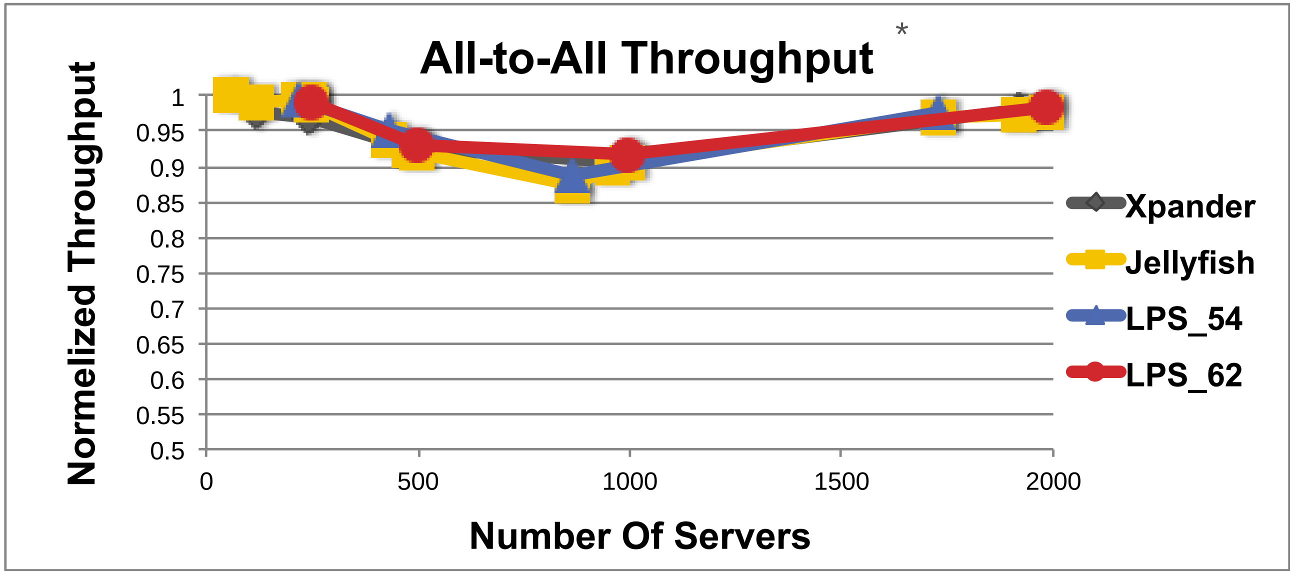
Expander datacenters Achieve Near-Optimal performance

- ➞ **Support higher traffic loads**
- ➞ **More resilient to failures**
- ➞ Support more servers with less network devices
- ➞ Multiple short-paths between hosts
- ➞ Incrementally expandable

Datacenter Throughput

- How much traffic can a datacenter network support?
  - The network is modelled as a capacitated graph $G=(V,E,c)$ coupled with a demand matrix D
  - The *maximum-concurrent-flow* α_D is the maximum α such that each commodity in D sends exactly an α of its demand
  - Common selections of D: All-to-All, Permutation, Many-to-One, and One-to-Many

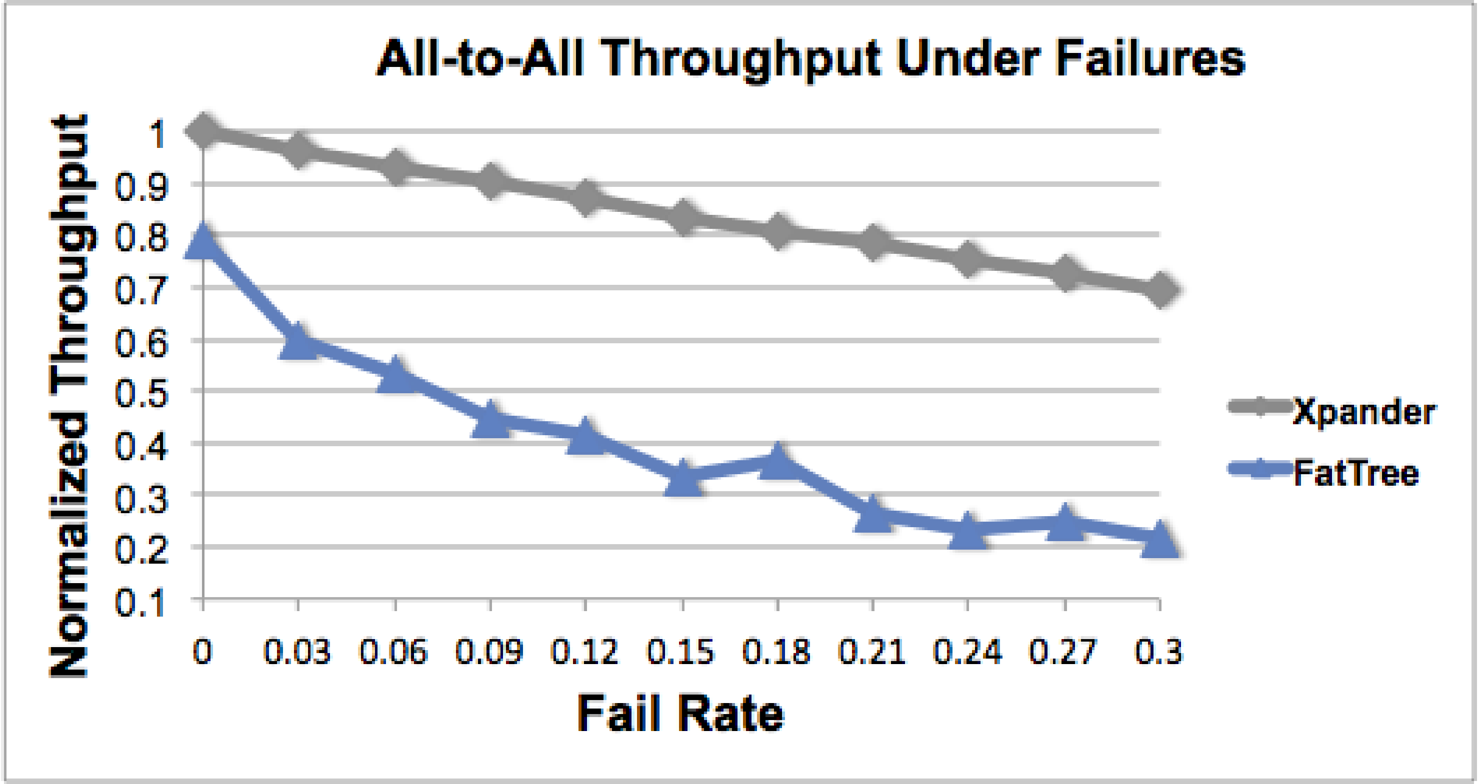
Near Optimal All-To-All Throughput



* 18-port switches

<u>Theorem:</u> In the all-to-all setting, the throughout of any *d*-regular expander *G* on *n* vertices is within a factor of O(log*d*) of that of the throughput-optimal *d*-regular graph on n vertices

Resilience To Failures



Observation: In any d-regular expander (with edge expansion >=1), any two vertices are connected by exactly d edge-disjoint paths

Datacenter Traffic

- Datacenter traffic is unpredictable
  - Different tenants want different things
  - Varying degree of mixture between long and short flows
- With different types of skewness (i.e., percentage of chatty servers)
  - Could range between a uniform to highly skewed distributions

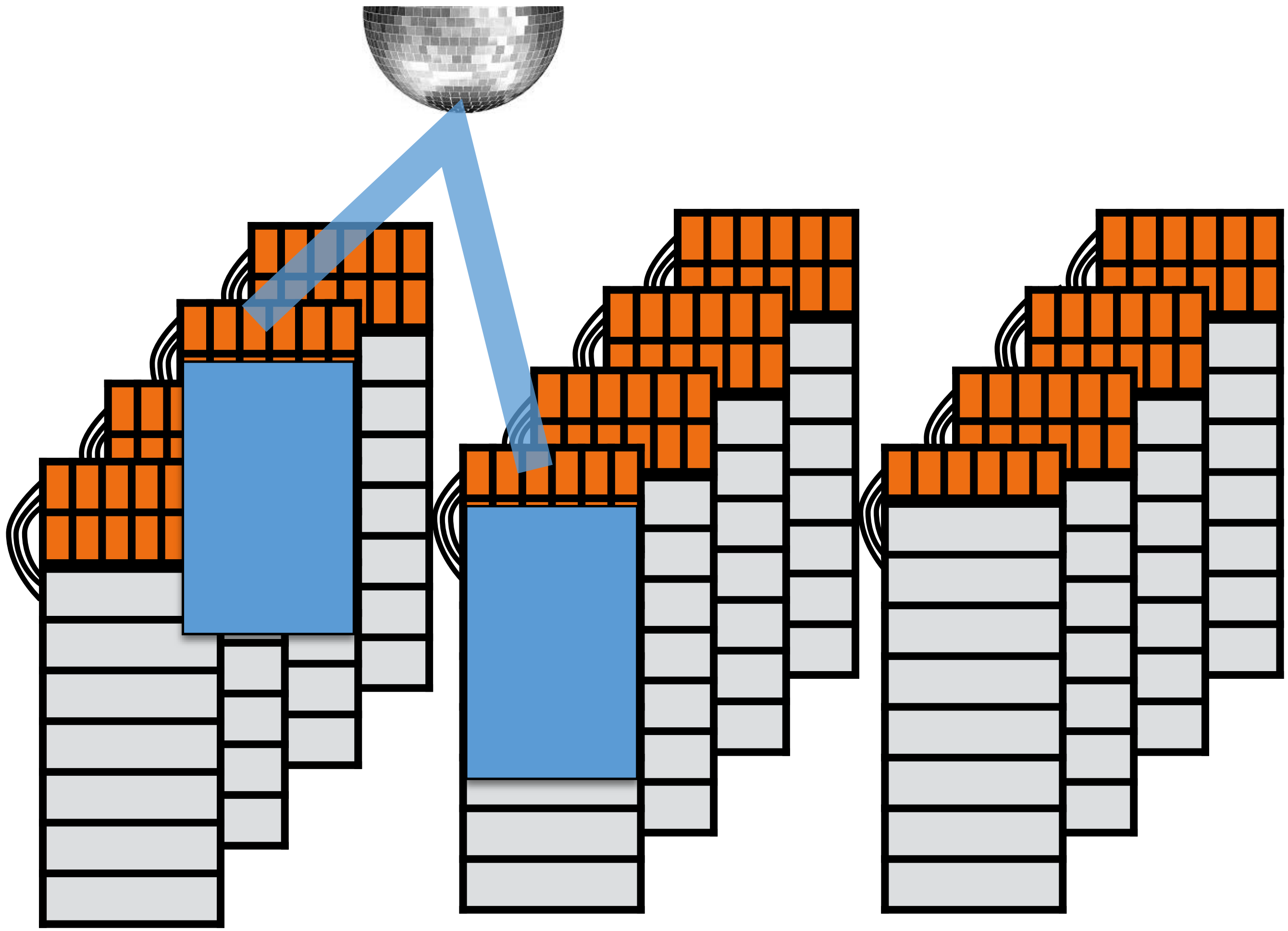
Near-Optimal Throughput Even Against Adversarial Traffic!

Theorem 1: Throughput of any expander on *n* vertices is a logarithmic (in *n*) factor away from the optimum with respect to any traffic pattern

Theorem 2: For any *d*-regular graph *G* on *n* vertices there is some traffic matrix under which the throughput of *G* is a logarithmic (in *n*) factor away from the optimum

Distance from Optimum	Xpander
throughput<80%	<1%
80% ≤ throughput <85%	2.3%
85% ≤ throughput <90%	16.14%
90% ≤ throughput <95%	44.48%
95% ≤ throughput	36.61%

Dynamic Networks: Set Up Network Connections On The Fly



Are Static Networks Irrelevant?

- Are fewer but flexible ports better than many cheaper static ones?

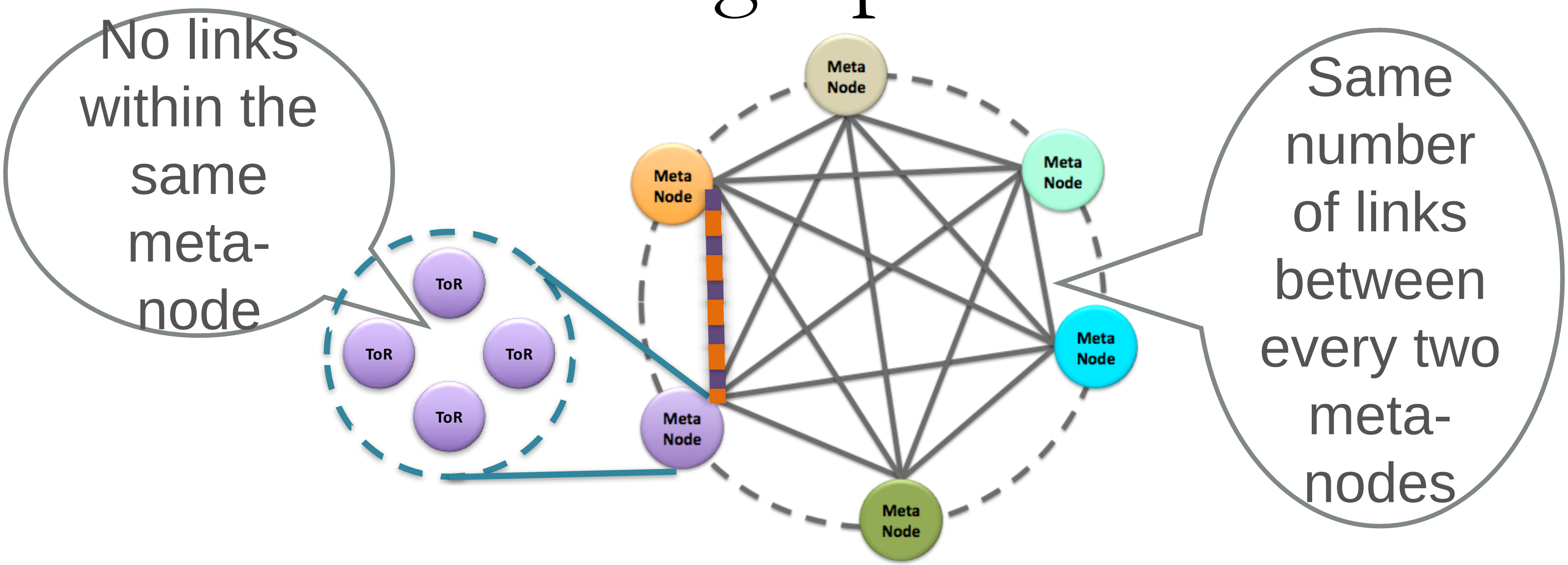
We show that Xpander attains performance comparable to state-of-the-art dynamic networks at a comparable cost!

This and more in our new SIGCOMM paper ☺

Deploying A New Datacenter Architecture

- Need to address the concerns of IT managing the datacenter, mainly:
  - Keeping changes to the protocol stack to a minimum: DCTCP as the congestion control mechanism and K-Shortest paths routing
  - Minimize cabling complexity *(see next slide)*
  - Have the ability to increase the datacenter size *More on this in Michael's talk (coming up next)*

Cabling Xpander



- Place ToRs of each meta-node in close proximity
- Bundle cables between two meta-nodes
- Use color-coding to distinguish between different meta-nodes and bundles of cables

Conclusion

- We show that expander datacenters outperform traditional datacenters
  - ✓ Sheds light on past results about random and low-diameter datacenter networks
- We present **Xpander**, a novel datacenter architecture
  - ✓ Suggests a **tangible** alternative to today's datacenter architectures
  - ✓ Achieves **near-optimal** performance

Thank you!

Questions?