

Stitching Inter-Domain Paths over IXPs

Vasileios Kotronis¹, Rowan Klöti¹, **Matthias Rost**², Panagiotis Georgopoulos¹,
Bernhard Ager¹, Stefan Schmid³, Xenofontas Dimitropoulos^{1,4}

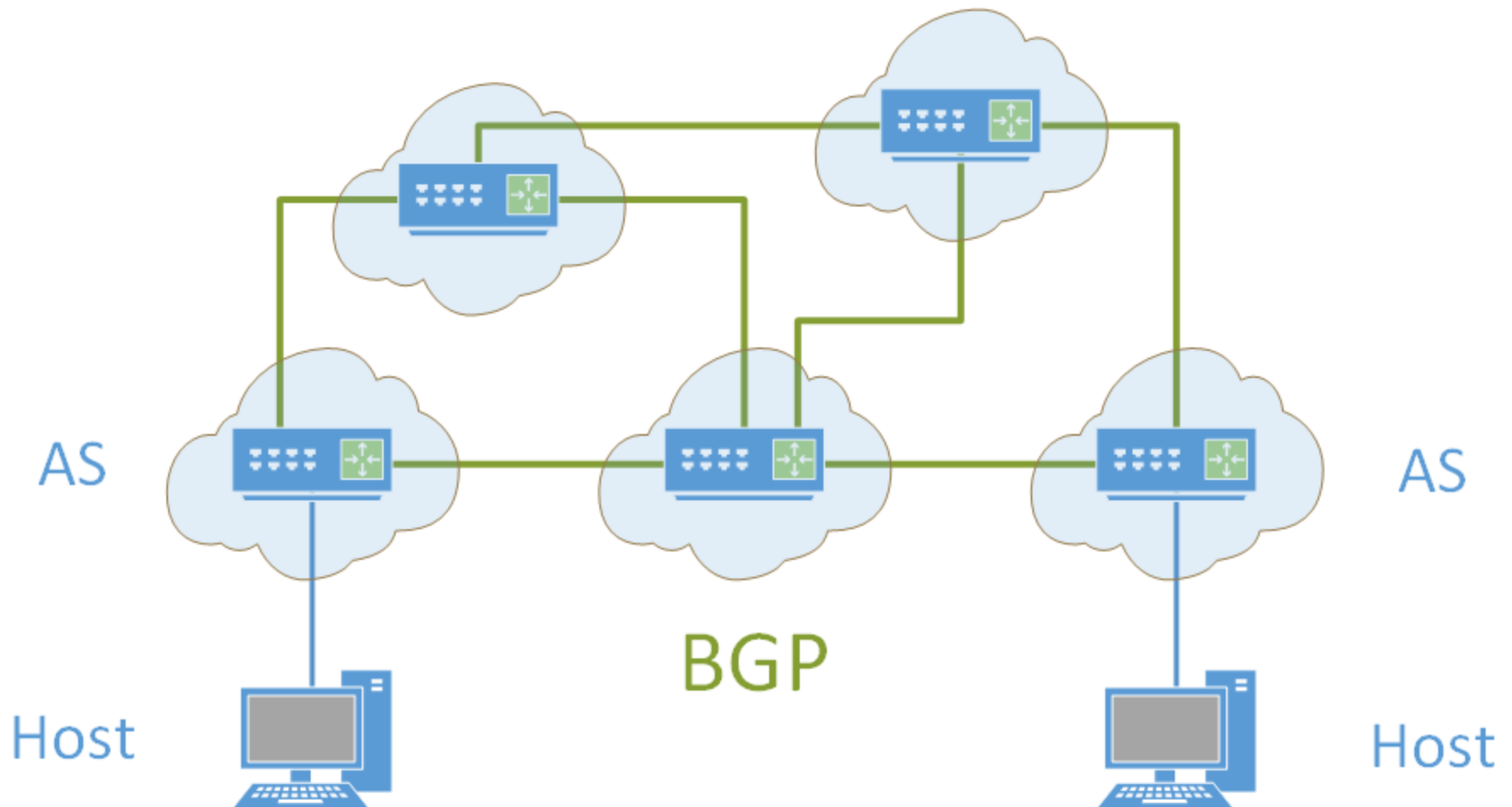
1 ETH Zurich
Switzerland

2 TU Berlin
Germany

3 Aalborg University
Denmark

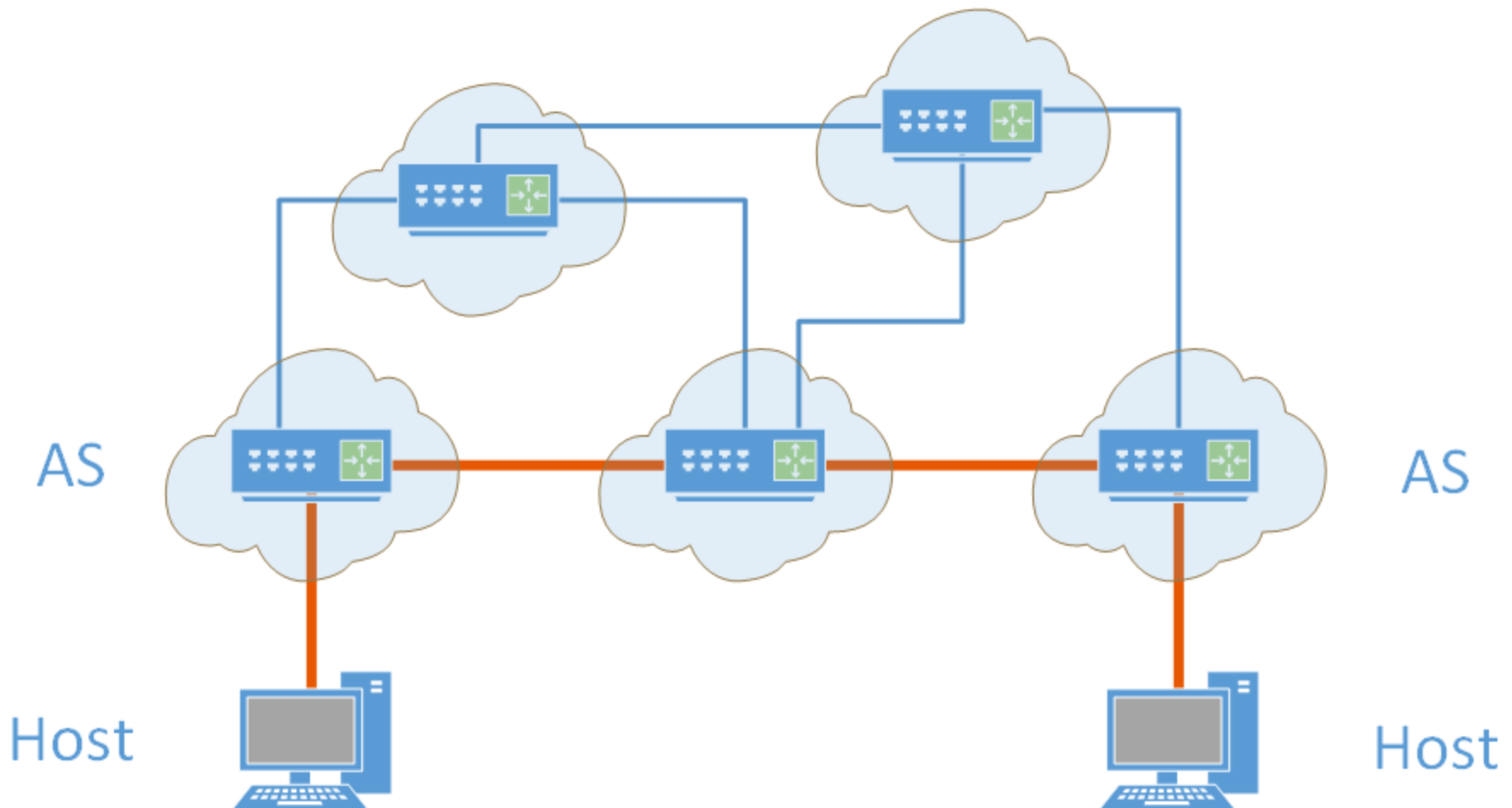
4 FORTH
Greece

Inter-Domain Routing: Status Quo



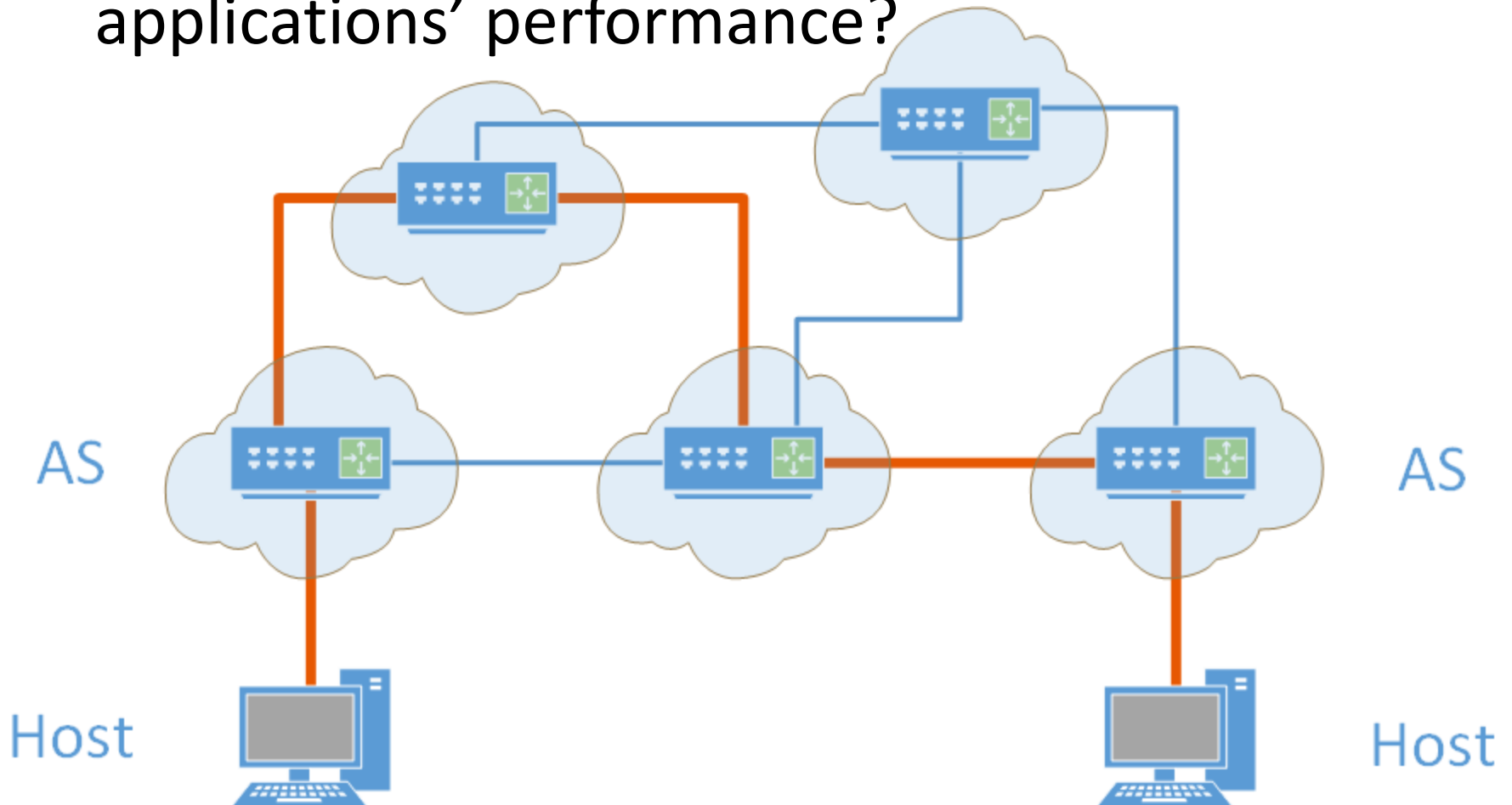
Inter-Domain Routing: Status Quo

- BGP selects *single* policy-compliant path



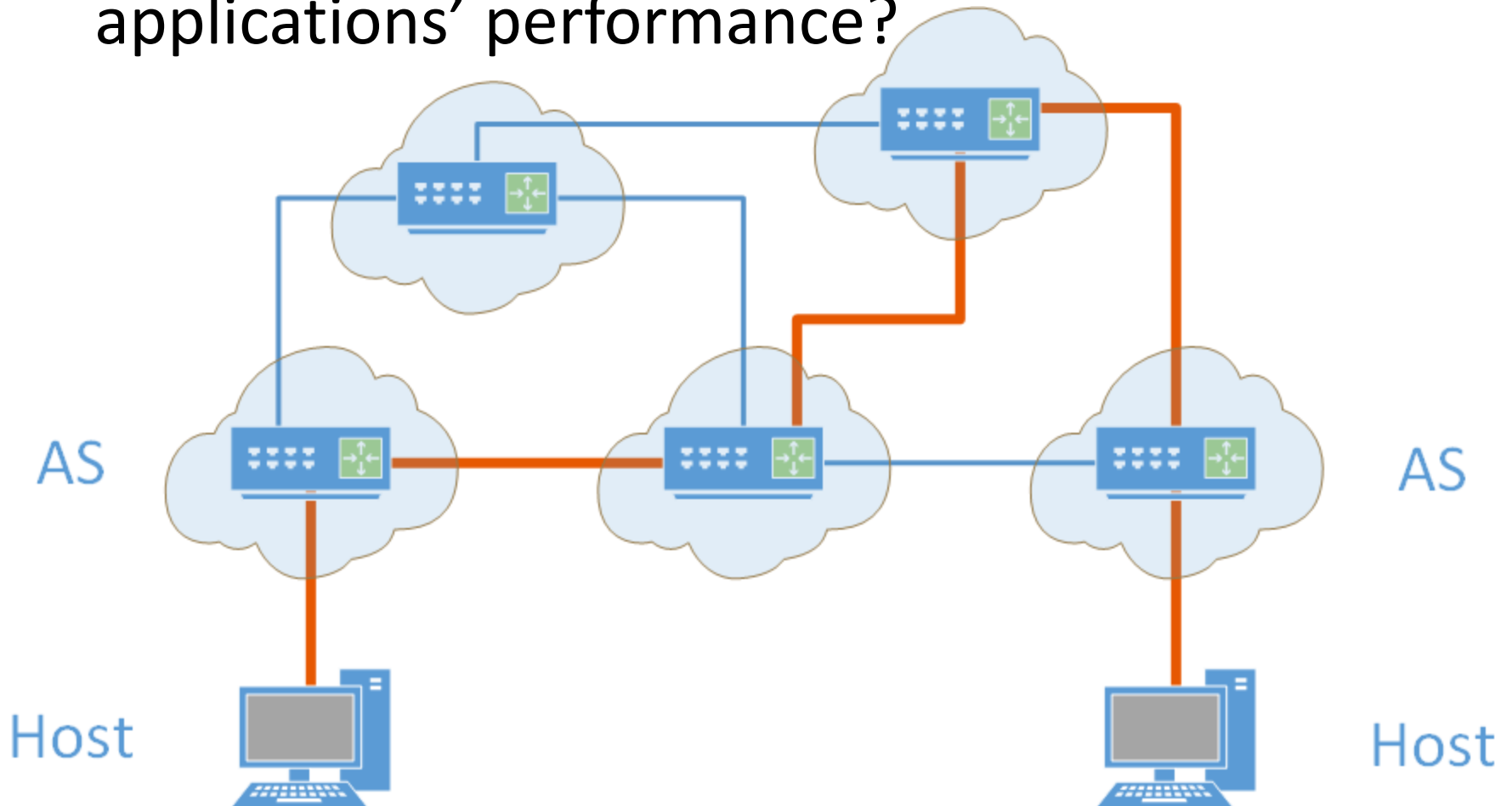
Inter-Domain Routing: Problem Statement

- How can we harness path diversity to improve applications' performance?



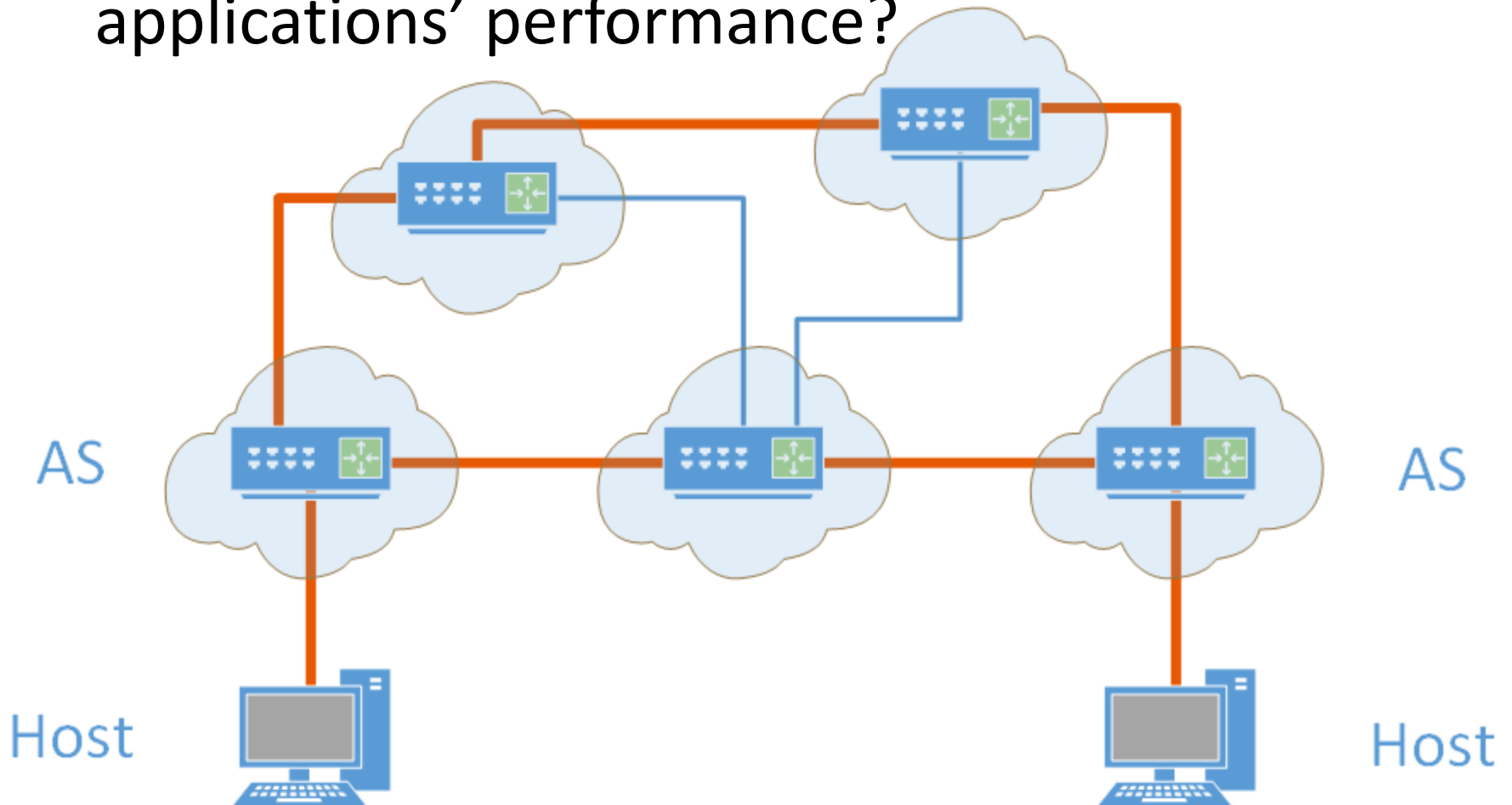
Inter-Domain Routing: Problem Statement

- How can we harness path diversity to improve applications' performance?



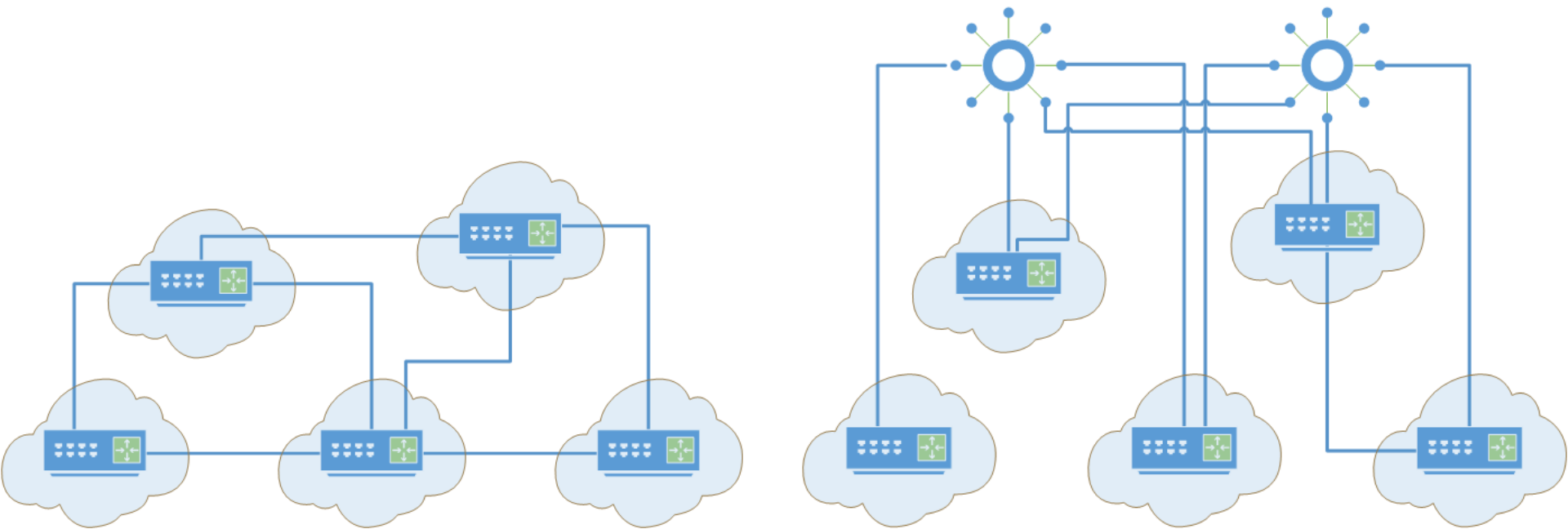
Inter-Domain Routing: Problem Statement

- How can we harness path diversity to improve applications' performance?



Our Proposal: Stitching Paths at IXPs

- ASes connect at Internet Exchange Points (IXPs)

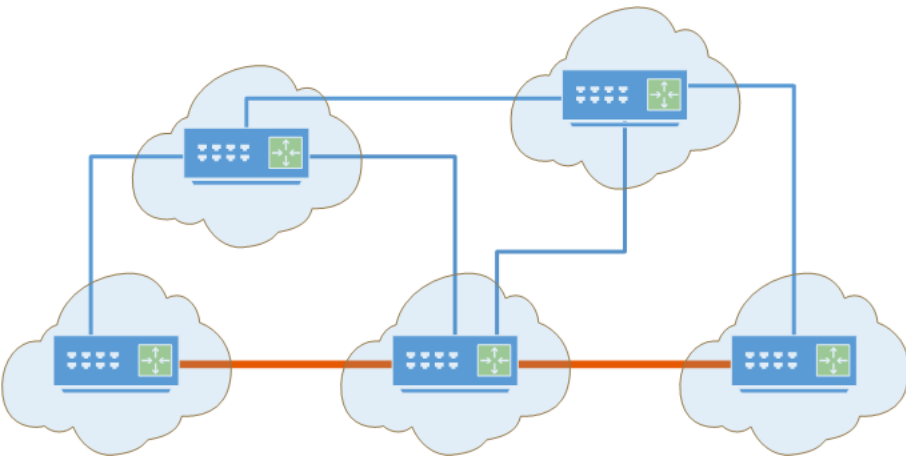


AS-centric view

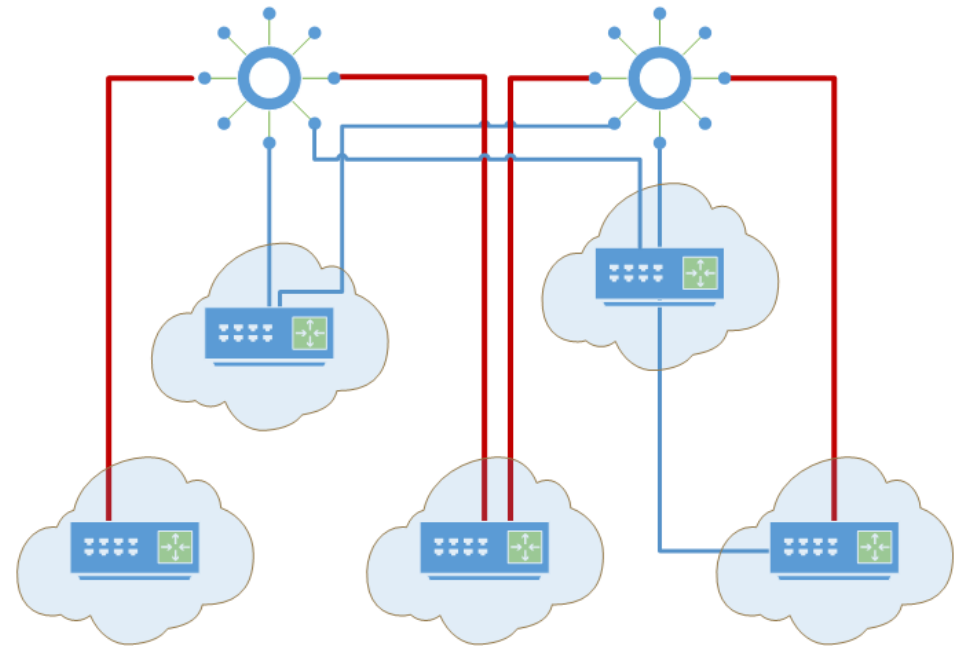
IXP-centric view

Our Proposal: Stitching Paths at IXPs

- ASes connect at Internet Exchange Points (IXPs)
- Many paths cross IXPs



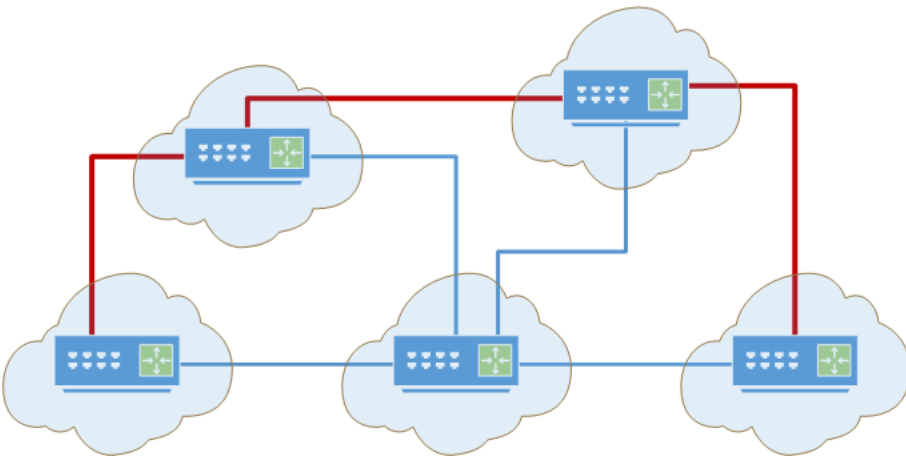
AS-centric view



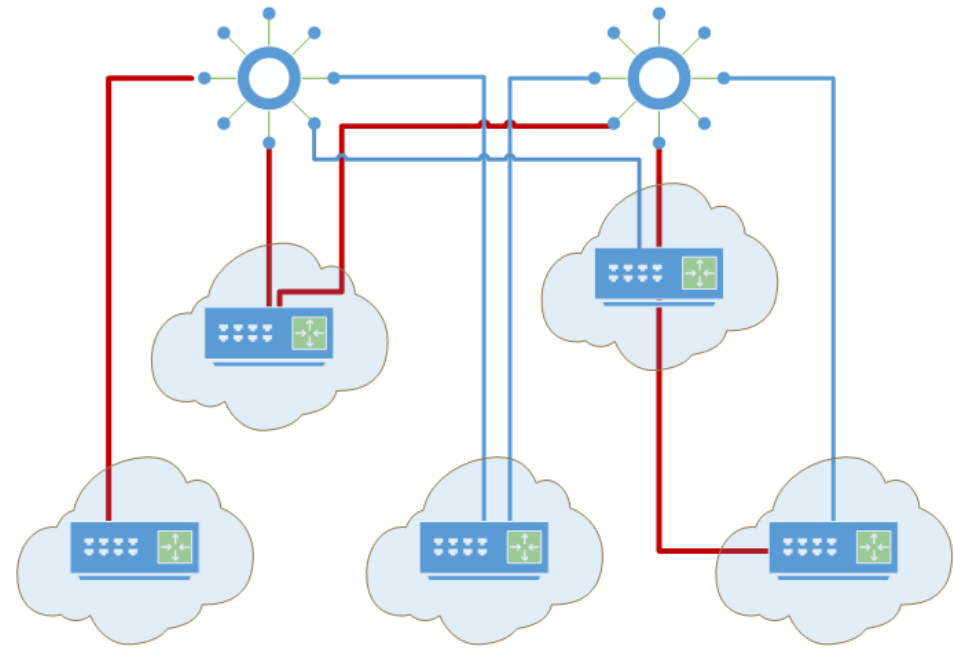
IXP-centric view

Our Proposal: Stitching Paths at IXPs

- ASes connect at Internet Exchange Points (IXPs)
- Many paths cross IXPs



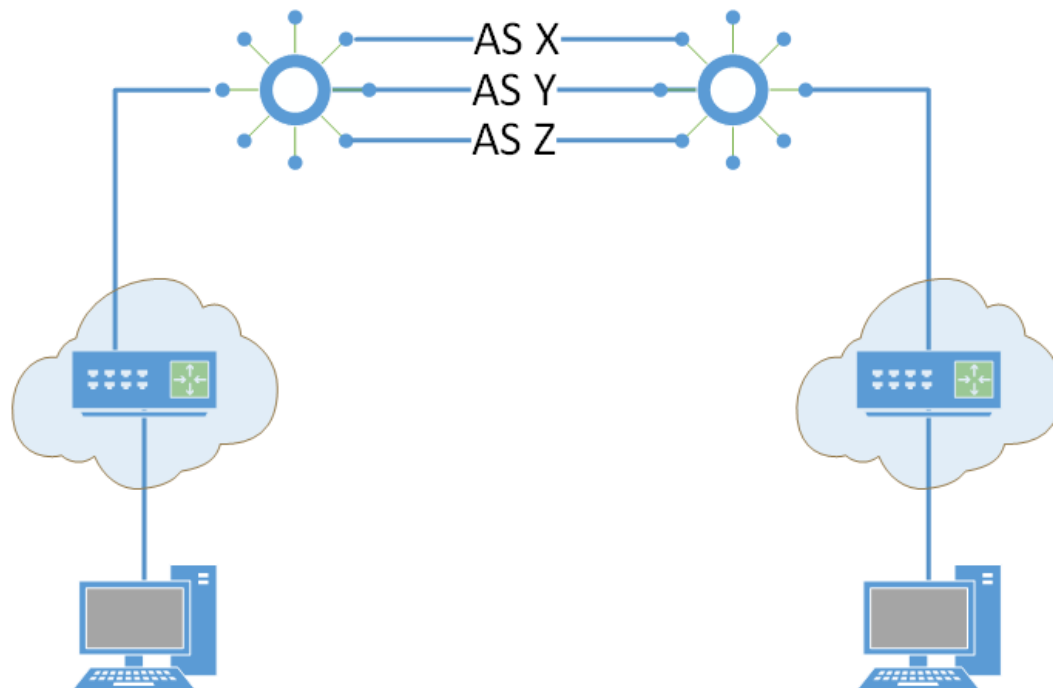
AS-centric view



IXP-centric view

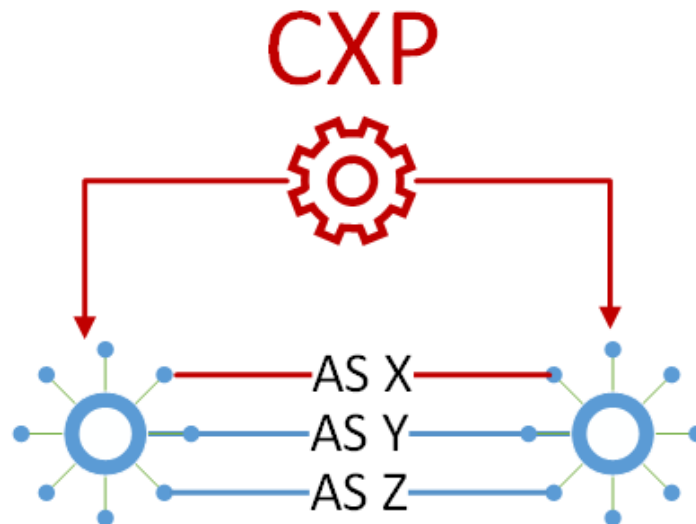
Our Proposal: Stitching Paths at IXPs

- ASes connect at Internet Exchange Points (IXPs)
- Idea: use ASes for inter-IXP transit



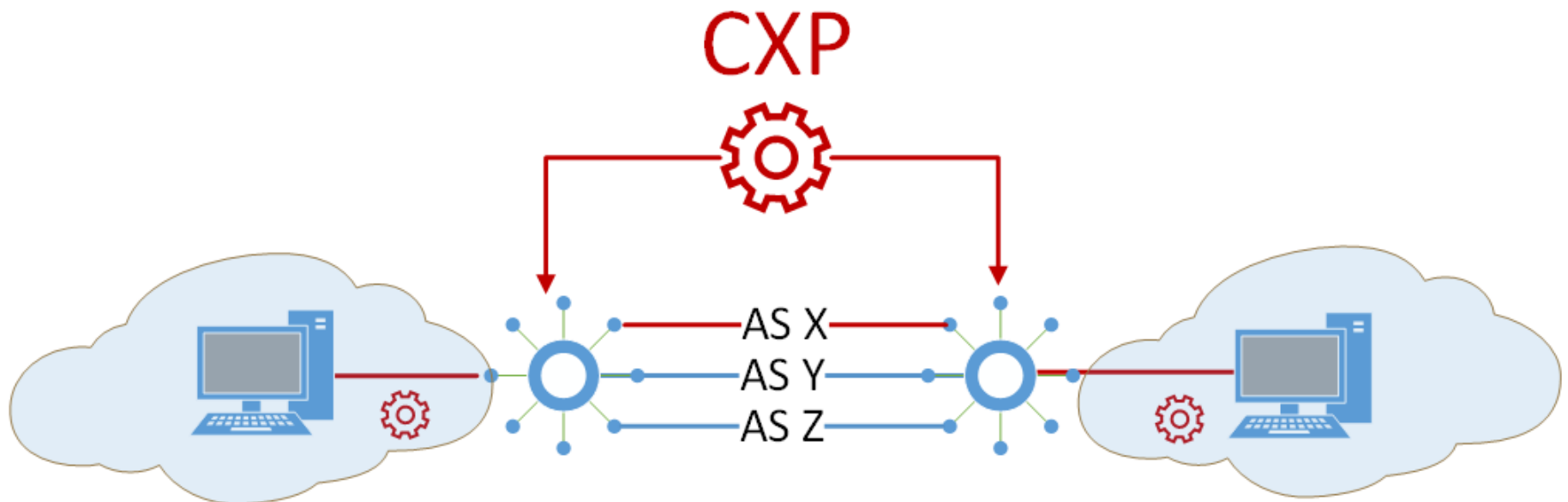
Our Proposal: Control Exchange Points

- **Centrally** stitch inter-IXP paths at IXPs using CXPs



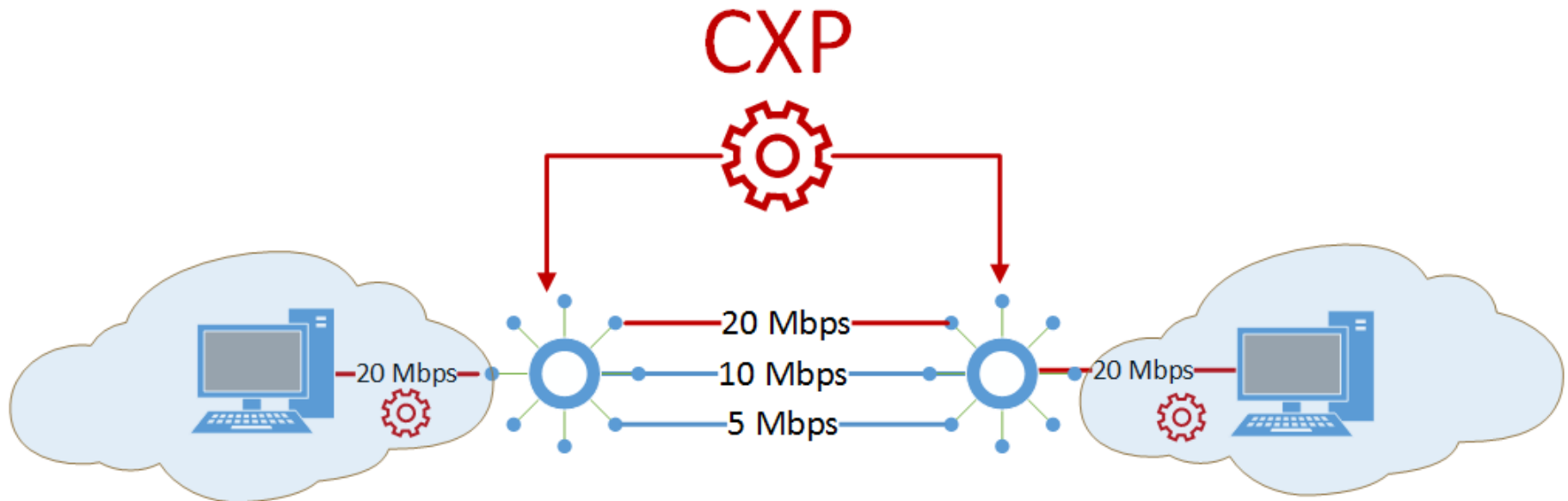
Our Proposal: Control Exchange Points

- **Centrally** stitch inter-IXP paths at IXPs using CXPs
- ASes responsible to connect hosts to IXPs / CXP



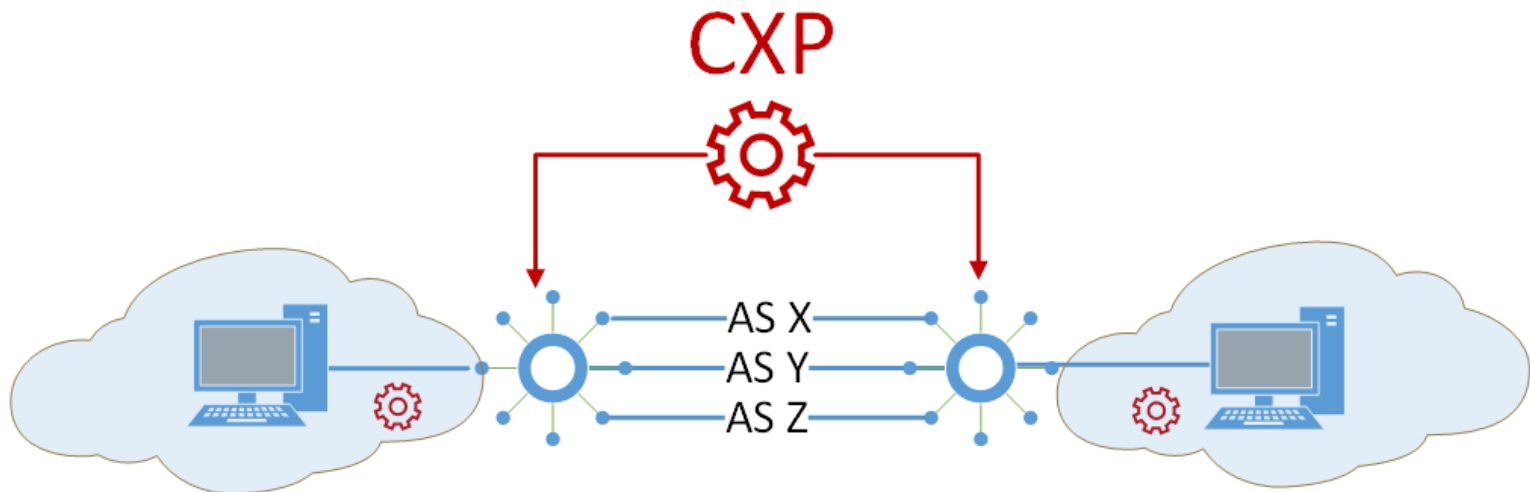
Our Proposal: Control Exchange Points

- **Centrally** stitch inter-IXP paths at IXPs using CXPs
- ASes responsible to connect hosts to IXPs / CXP
- CXP may measure inter-IXP link performance and/or ASes might provide guarantees



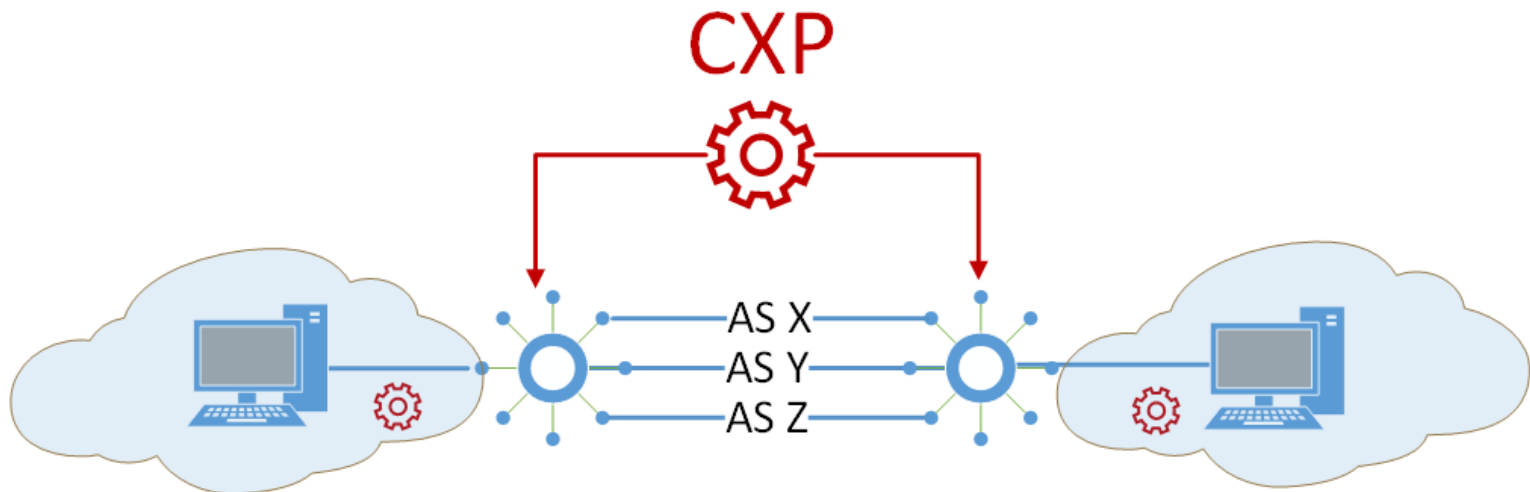
Main Questions

- Measuring the IXP multigraph
- Algorithms for embedding end-to-end paths



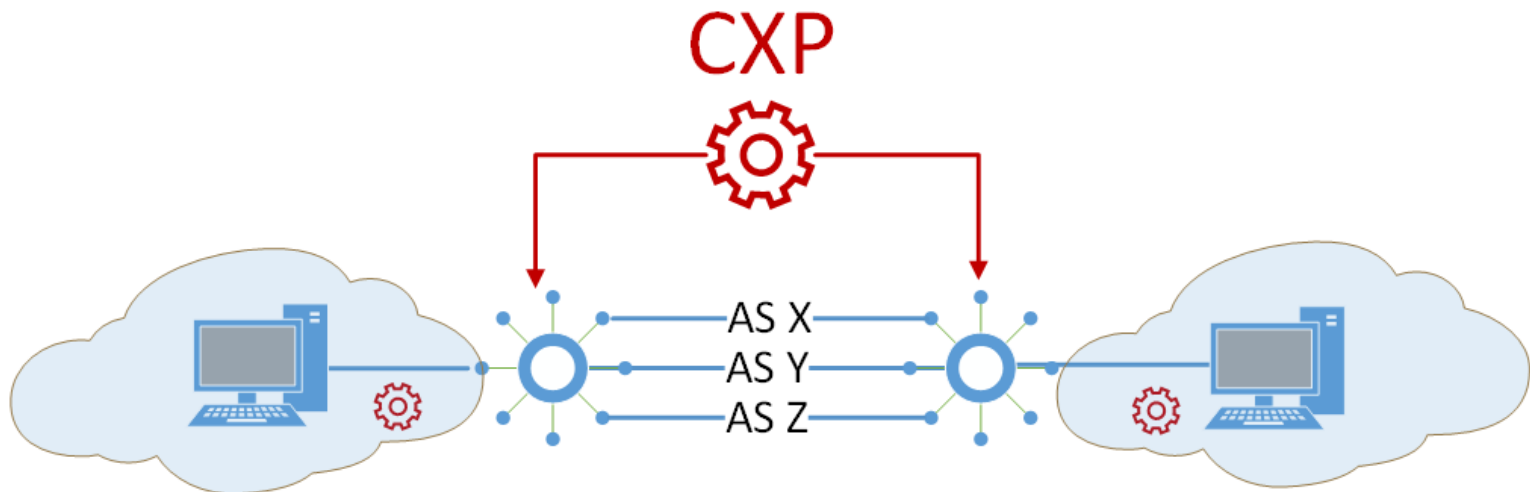
Main Questions

- Measuring the IXP multigraph
 - What is the gain in path diversity?
 - Which IXPs should be controlled by CXPs?
 - How many customers can we reach?
- Algorithms for embedding end-to-end paths



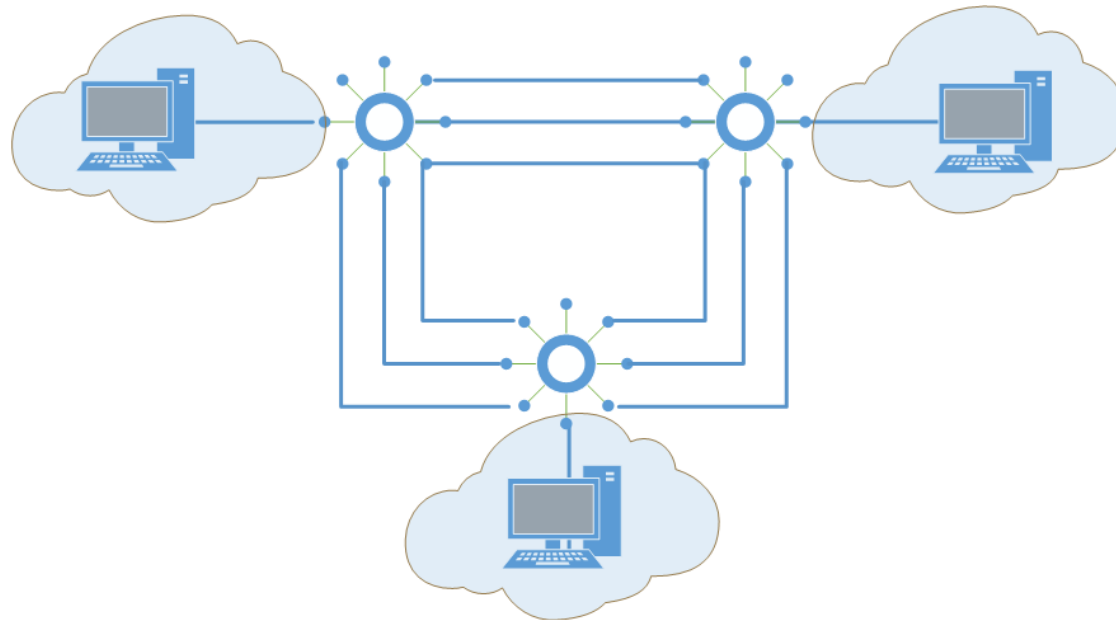
Main Questions

- Measuring the IXP multigraph
- Algorithms for embedding end-to-end paths
 - How to efficiently and centrally compute paths?
 - What are the opportunities of centralized control?



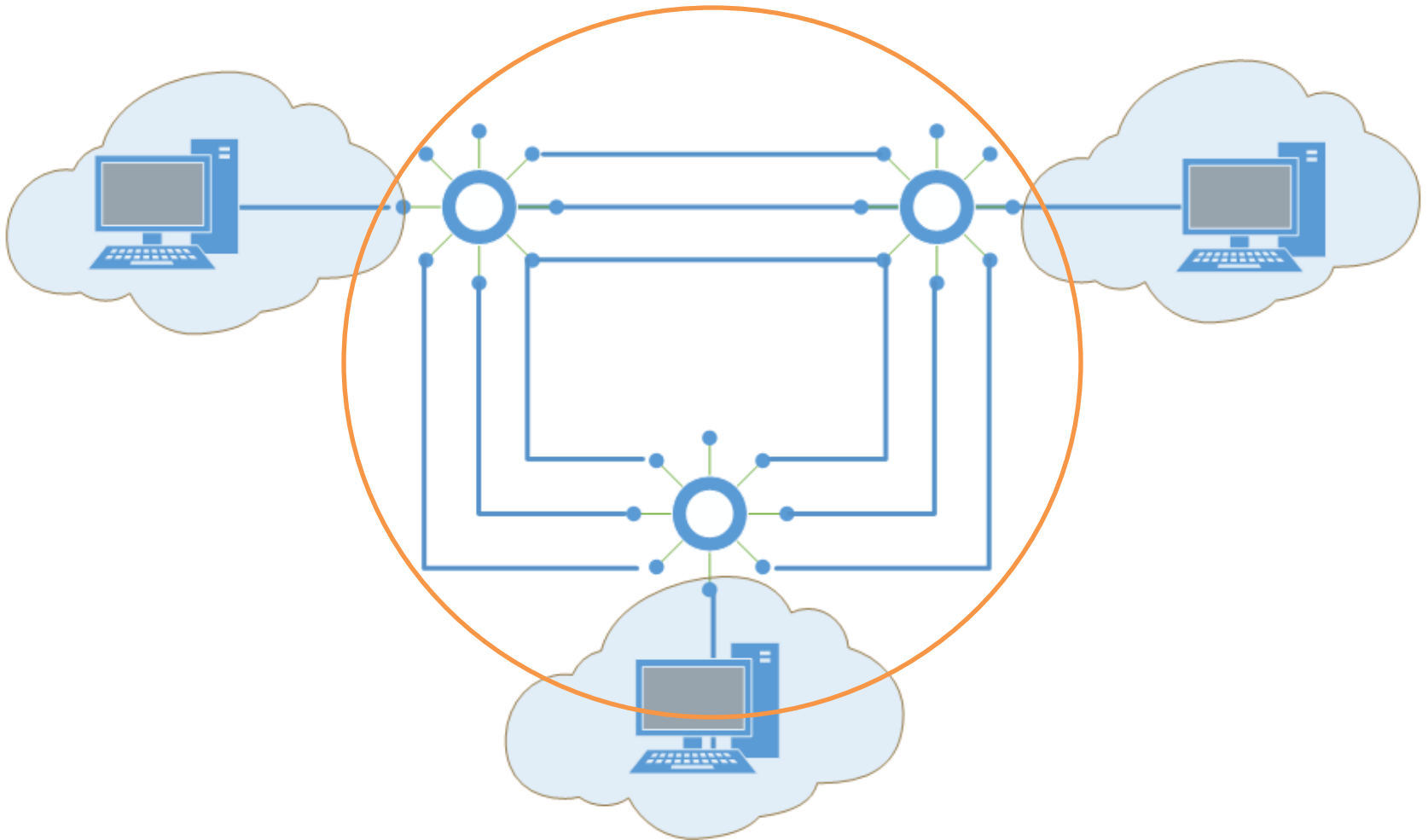
MEASURING THE IXP MULTIGRAPH

Methodology



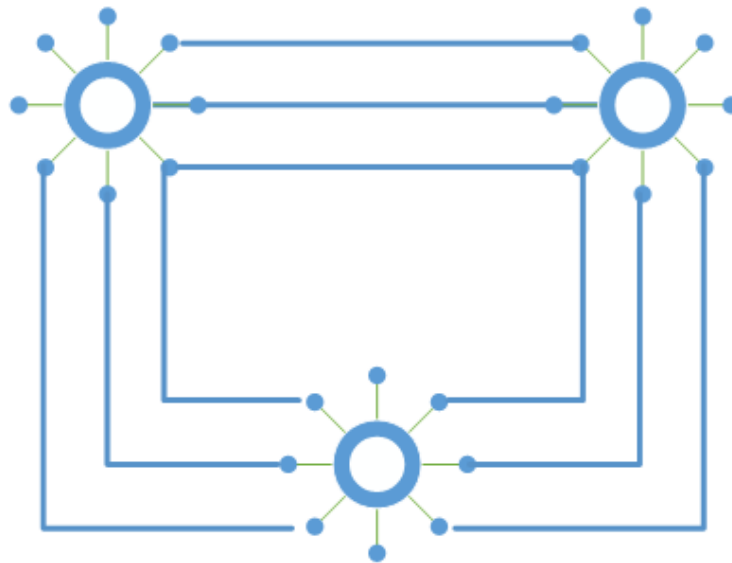
- Determine IXPs and the ASes connecting them
 - Euro-IX (and Peering-DB)
- Determine customer-cone of IXPs
 - CAIDA data

Results at a Glance



Results at a Glance: IXP Multigraph

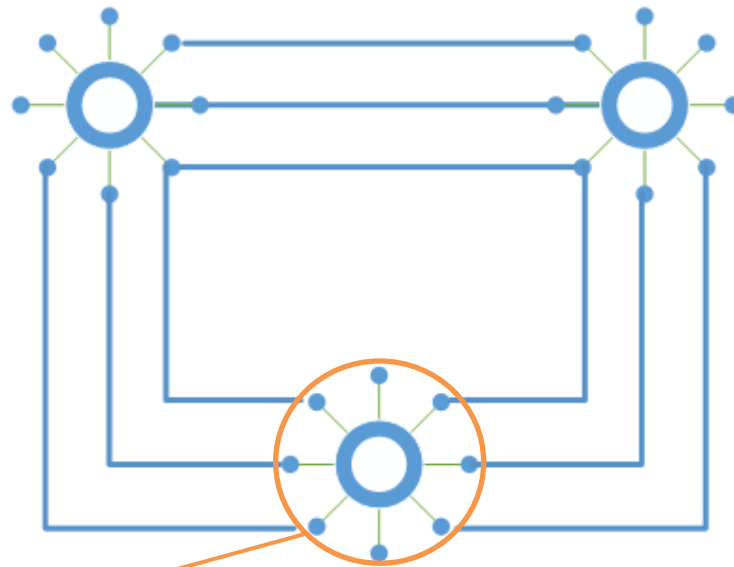
229 IXPs



49k edges

Results at a Glance: IXP Multigraph

229 IXPs



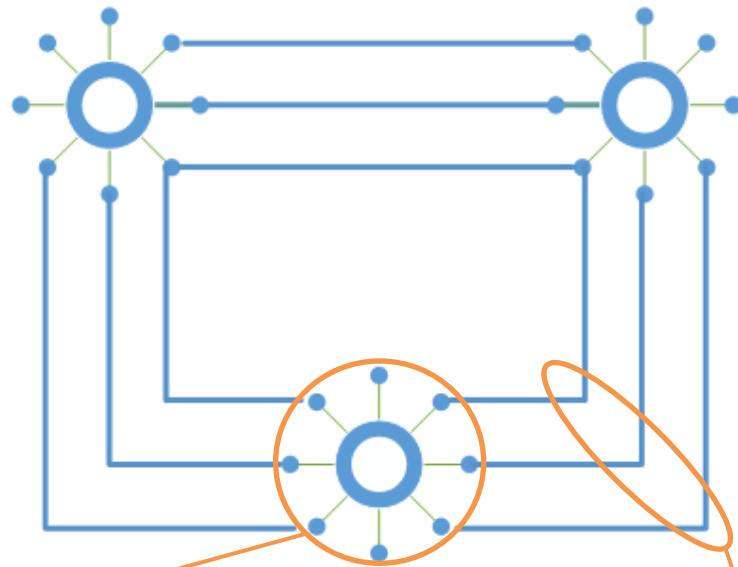
49k edges

Avg. degree: 220

Results at a Glance: IXP Multigraph

229 IXPs

49k edges

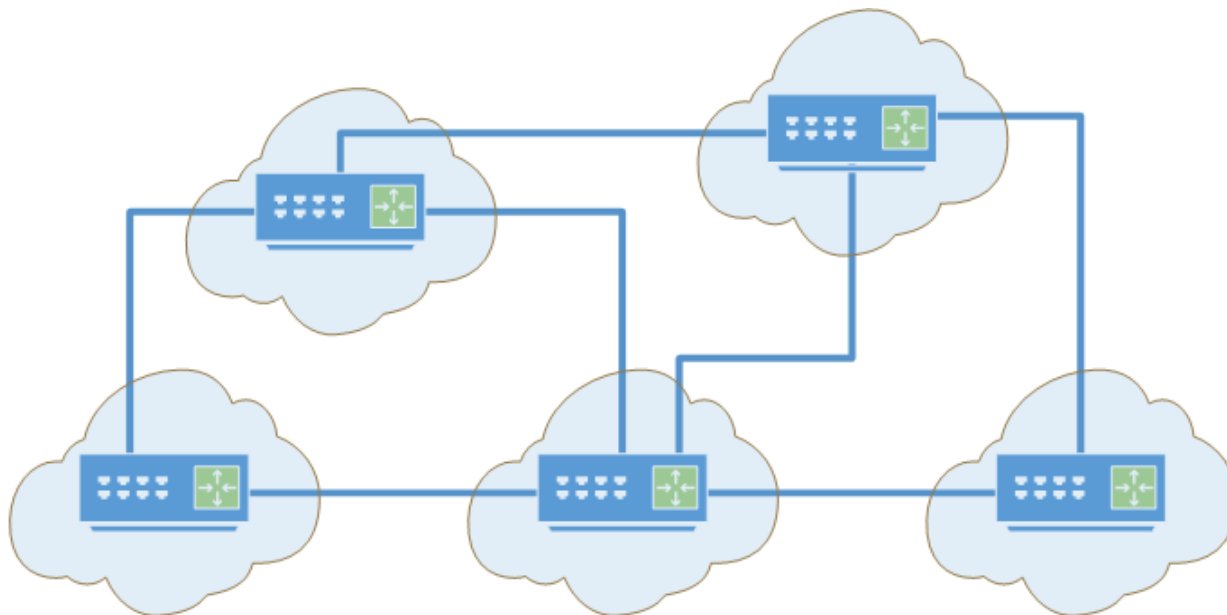


Avg. degree: 220

Avg. edge multiplicity: 4.3

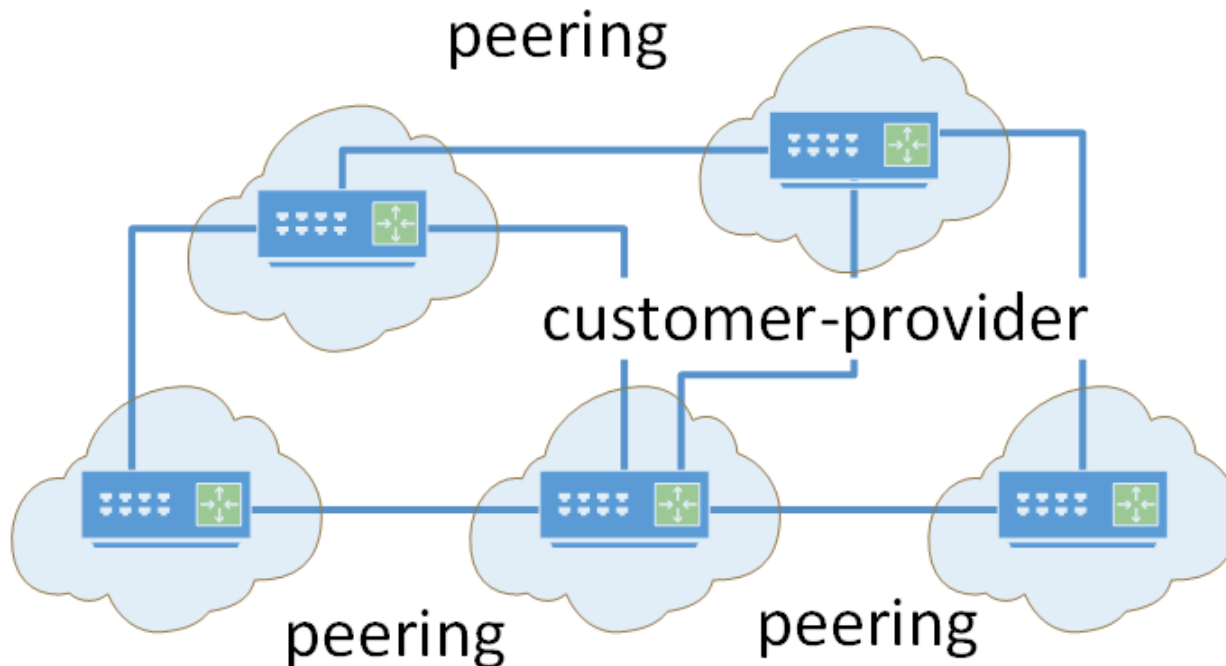
Path Diversity

- What is the gain in path diversity over BGP?



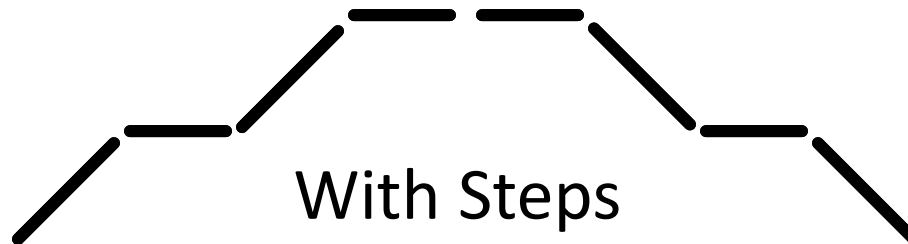
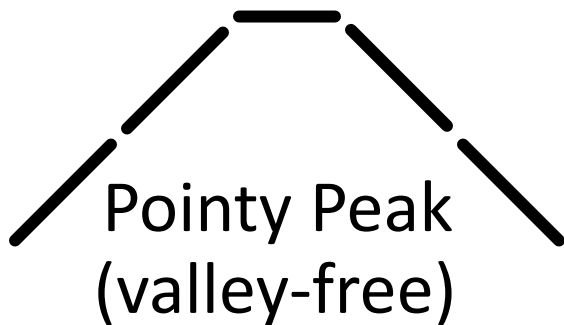
Path Diversity

- What is the gain in path diversity over BGP?
- BGP: valley-free (at most one peering link)

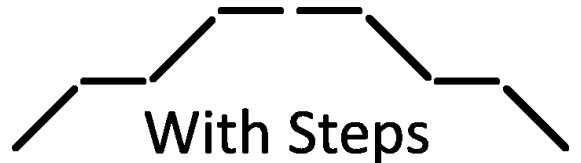
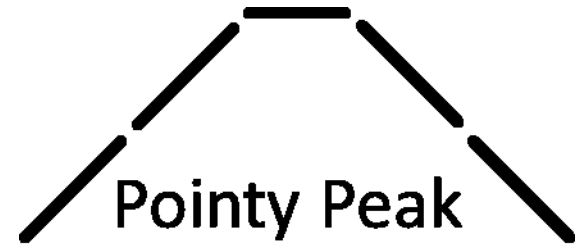
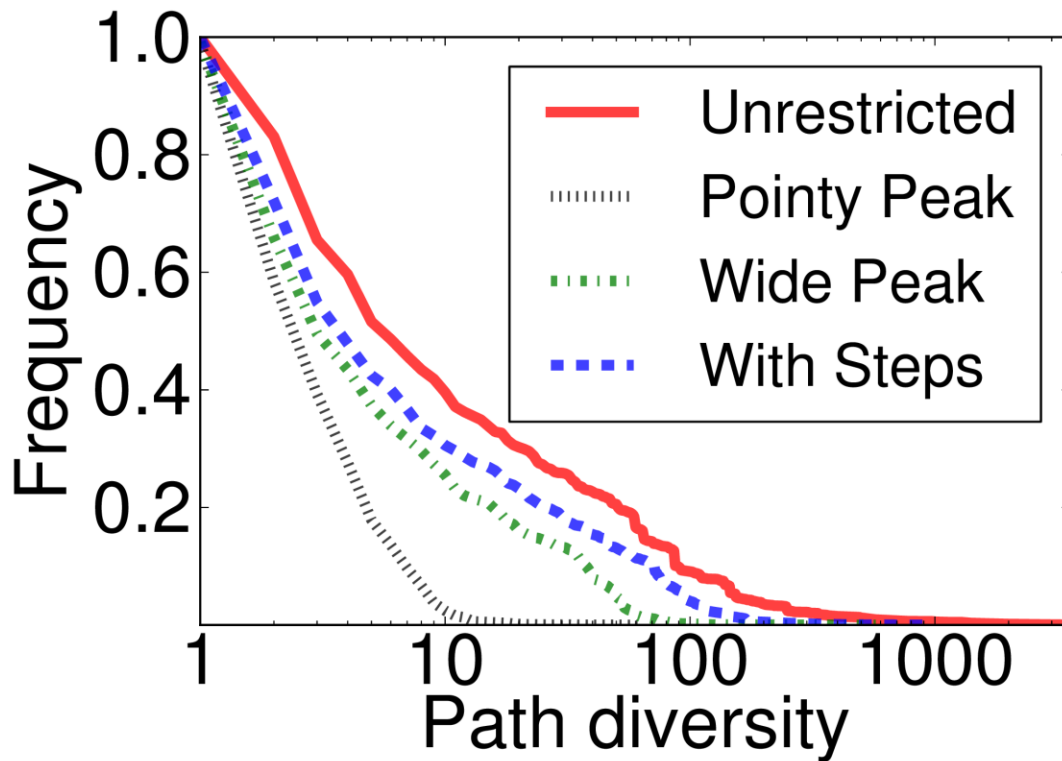


Path Diversity

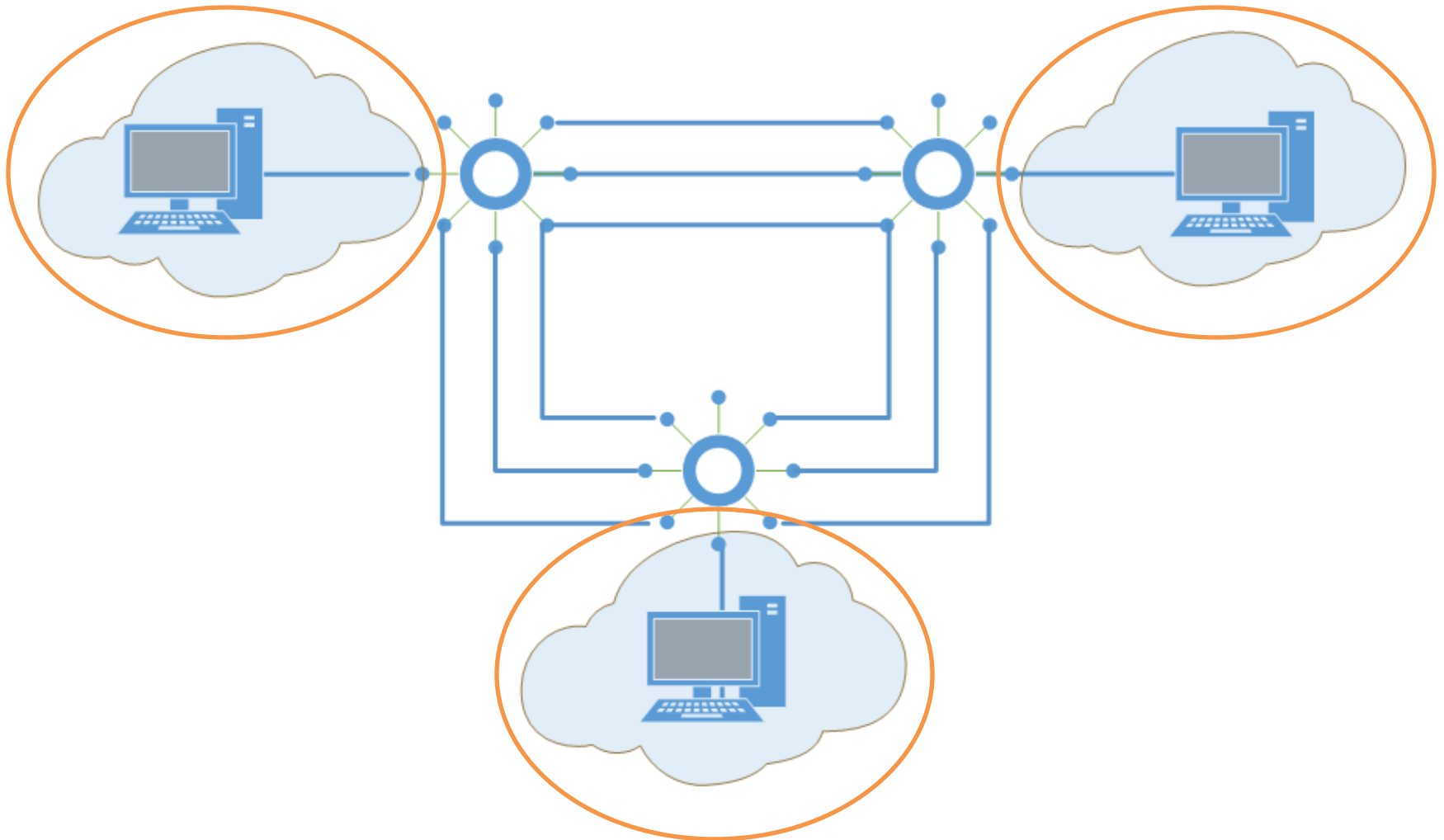
- What is the gain in path diversity over BGP?
- BGP: valley-free (at most one peering link)



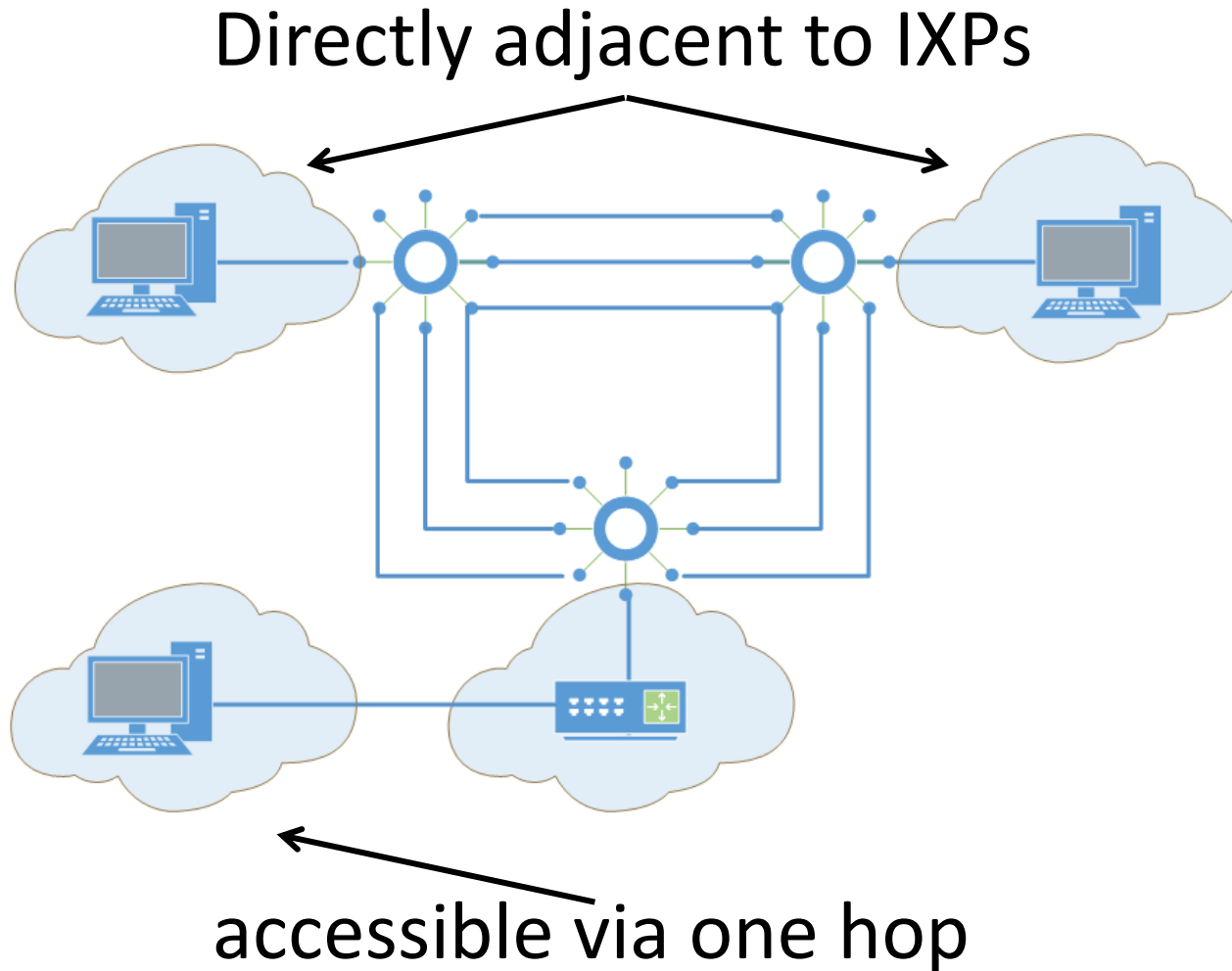
Path Diversity: Results



Results at a Glance: Customer Reach

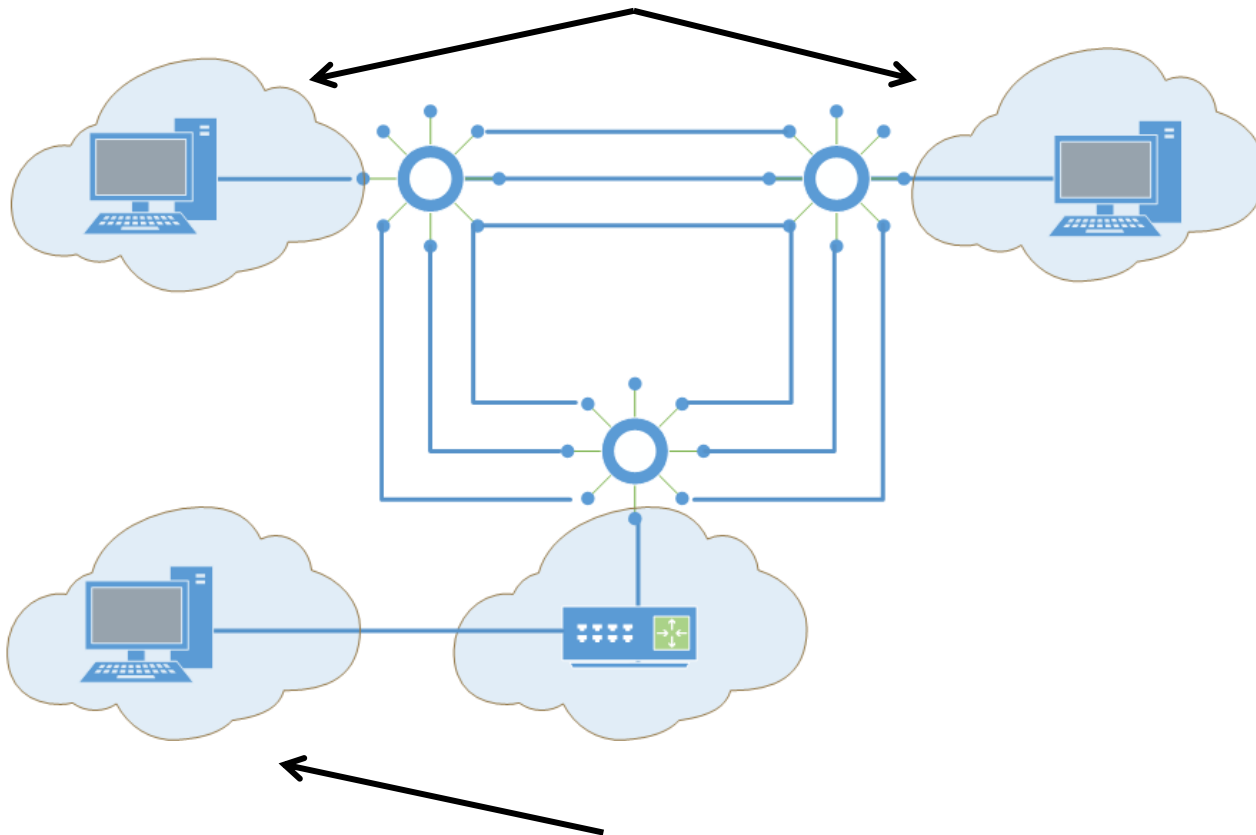


Results at a Glance: Customer Reach



Results at a Glance: Customer Reach

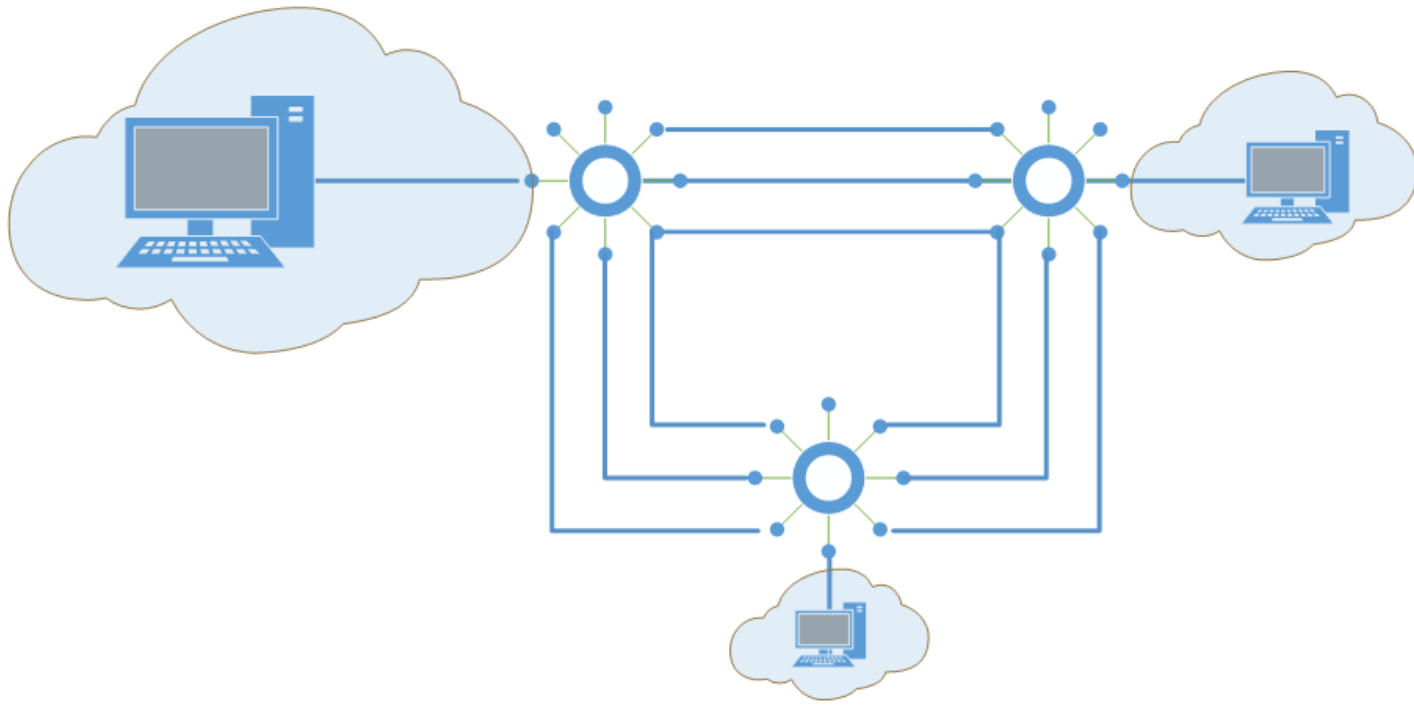
Approx. 61% of IPv4 addresses



Additional 30 % of IPv4 addresses

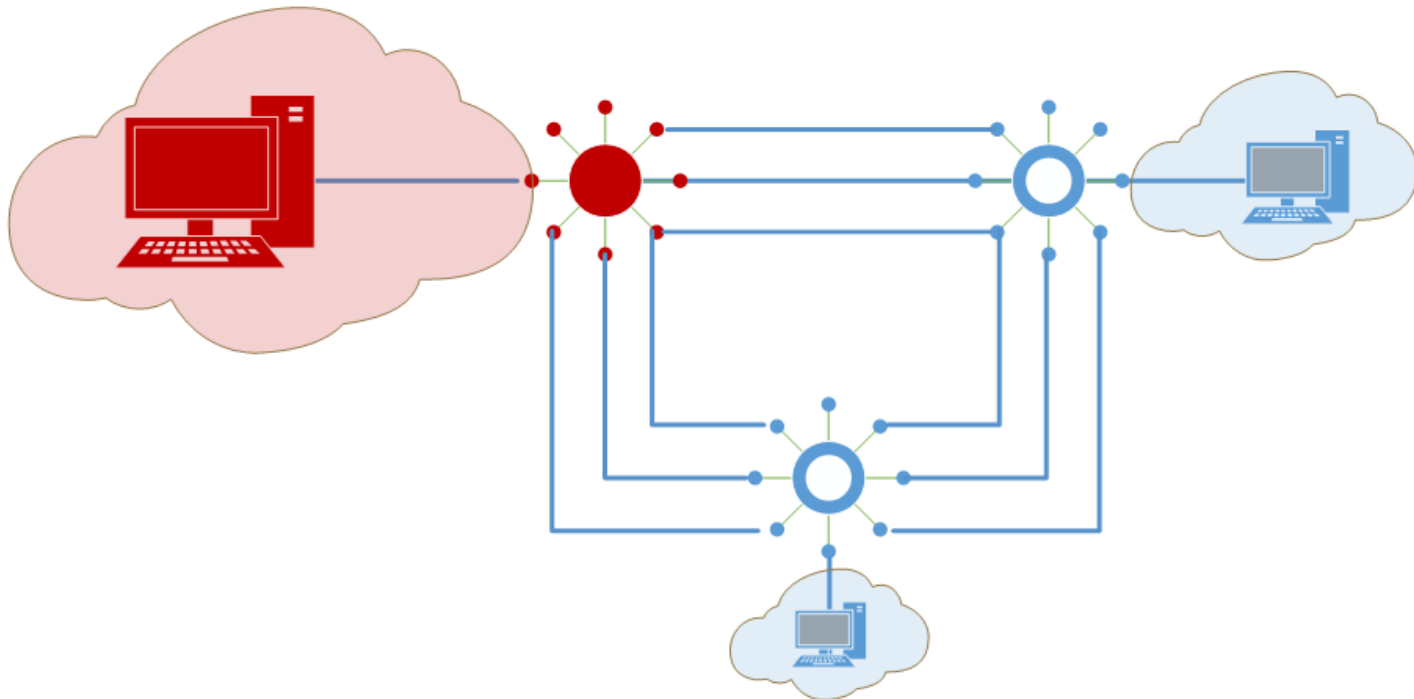
Incremental Deployment of CXPs

- Do we really need all of the 229 IXPs to offer end-to-end paths?
- Greedily select IXPs maximizing customer cone.



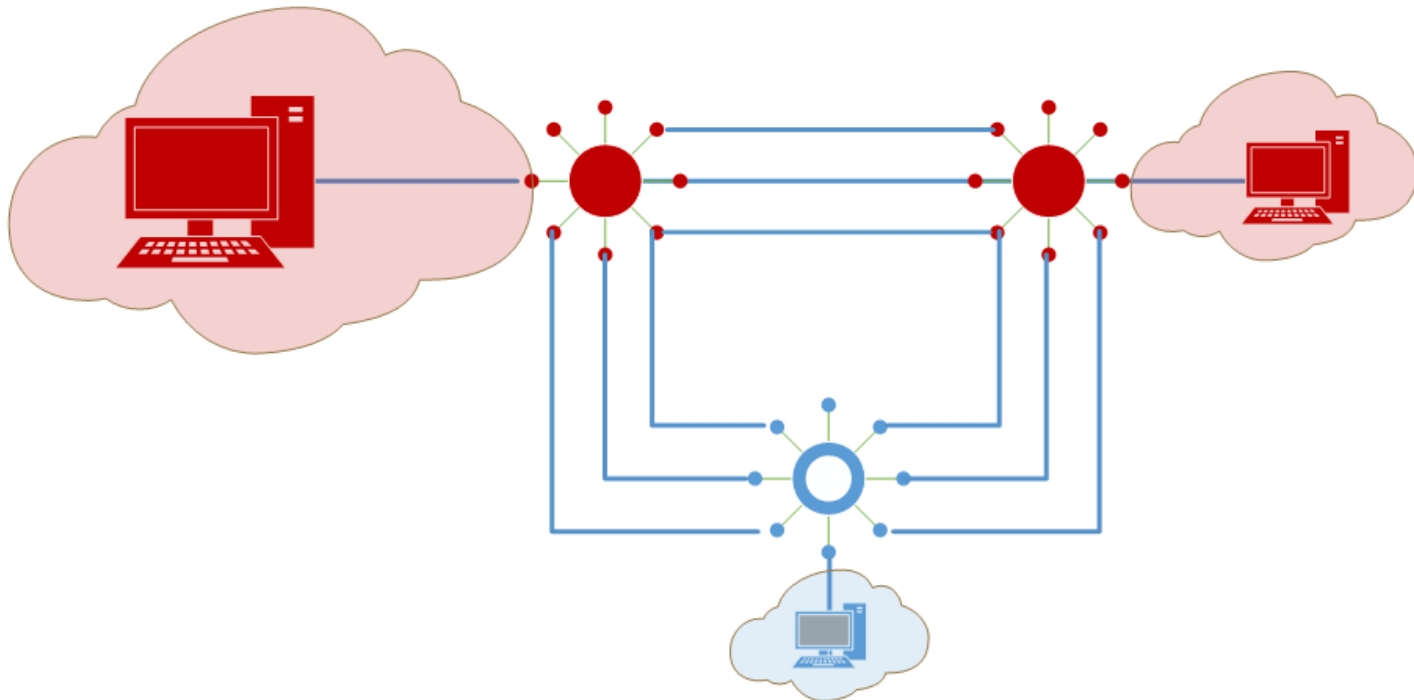
Incremental Deployment of CXPs

- Do we really need all of the 229 IXPs to offer end-to-end paths?
- Greedily select IXPs maximizing customer cone.



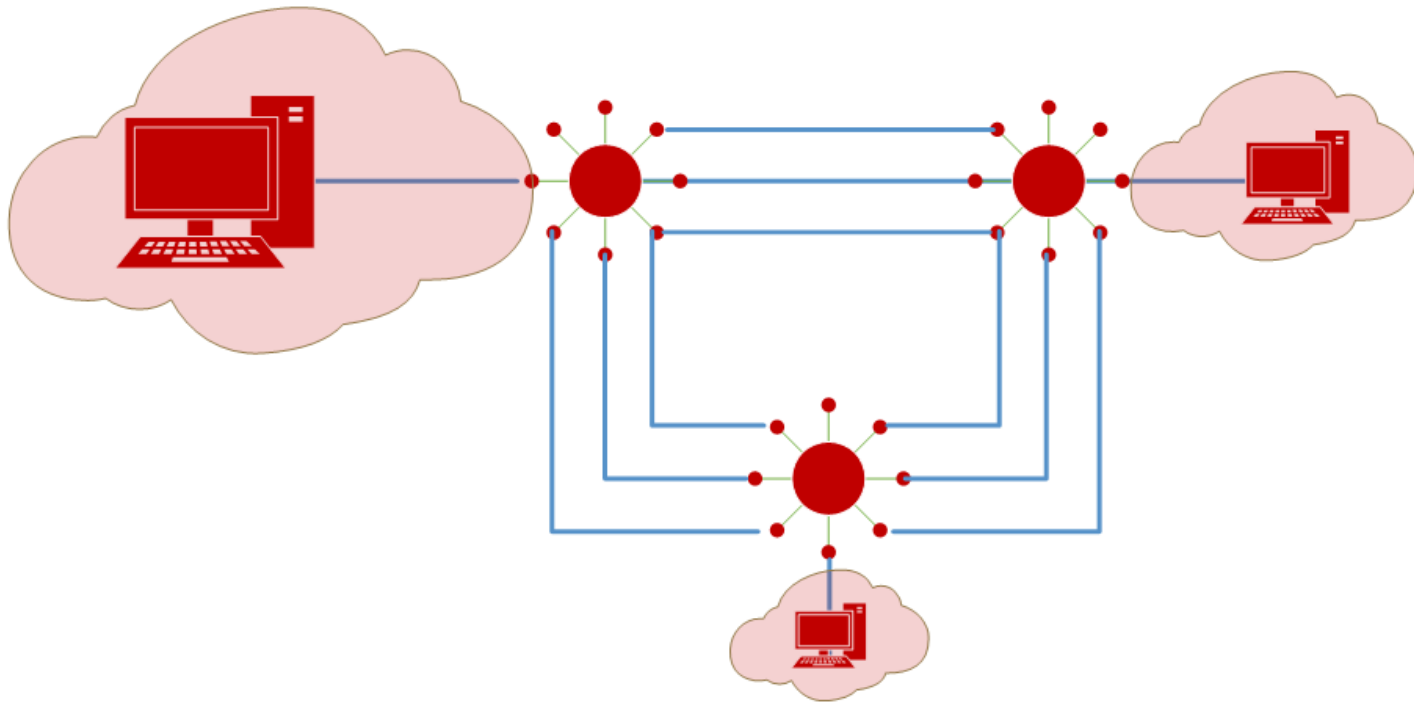
Incremental Deployment of CXPs

- Do we really need all of the 229 IXPs to offer end-to-end paths?
- Greedily select IXPs maximizing customer cone.



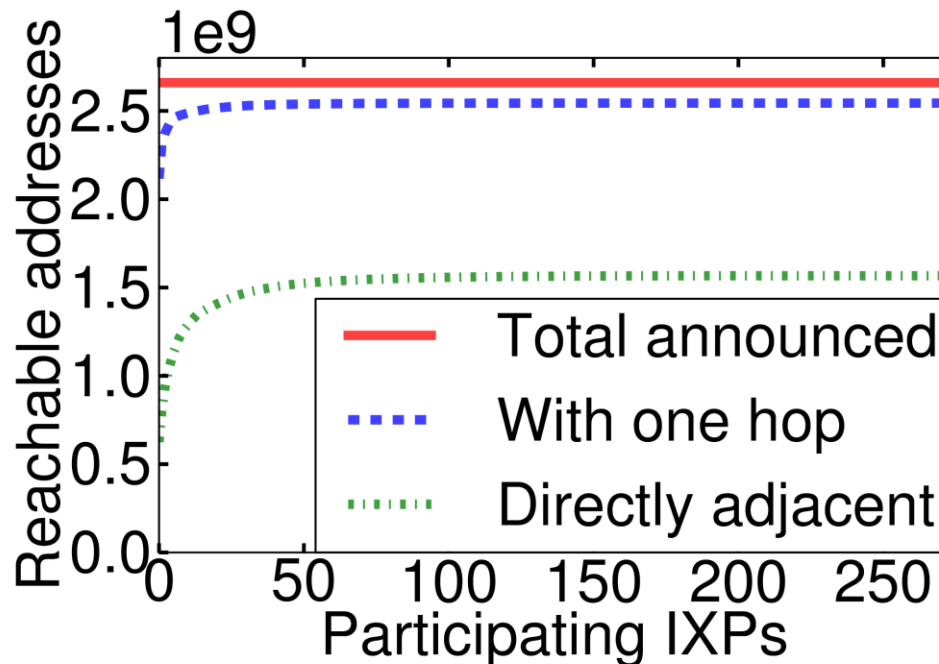
Incremental Deployment of CXPs

- Do we really need all of the 229 IXPs to offer end-to-end paths?
- Greedily select IXPs maximizing customer cone.



Incremental Deployment of CXPs

- Do we really need all of the 229 IXPs to offer end-to-end paths?
- Greedily select IXPs maximizing customer cone.



Incremental Deployment of CXPs

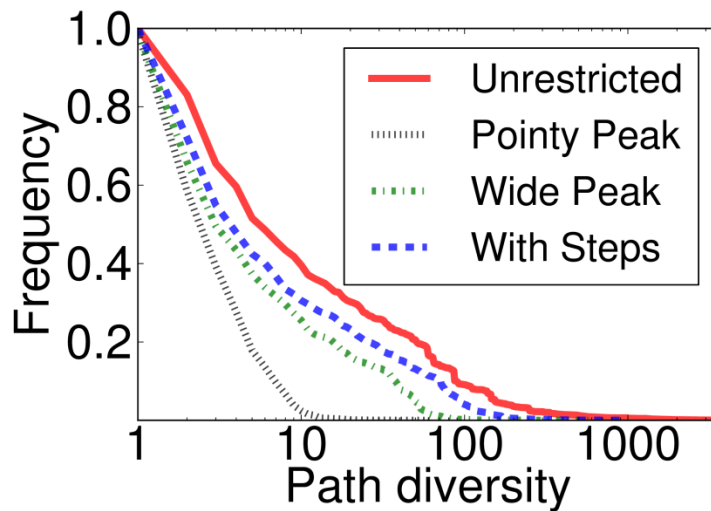
- Do we really need all of the 229 IXPs to offer end-to-end paths?
- Greedily select IXPs maximizing customer cone.

number of IXPs	reachable directly	with 1-hop
5	approx. 40%	approx. 91%
20	approx. 55%	approx. 92%

Conclusion Measurement Part

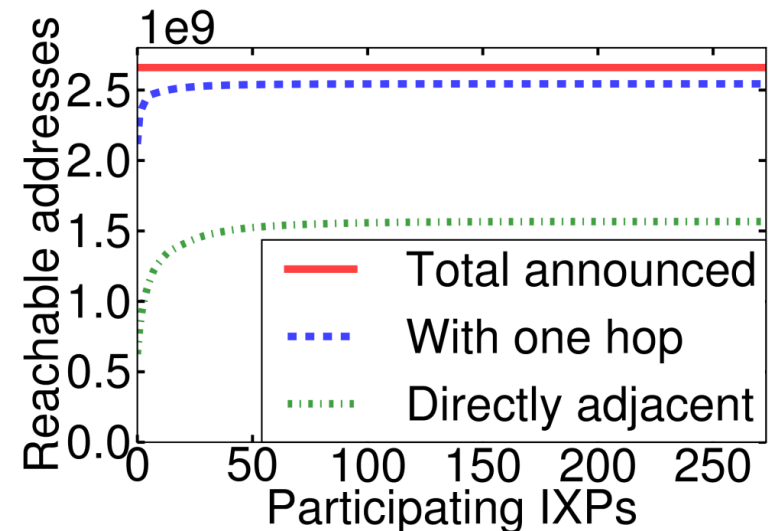
Gain in Path Diversity

- Significant improvement over BGP path diversity



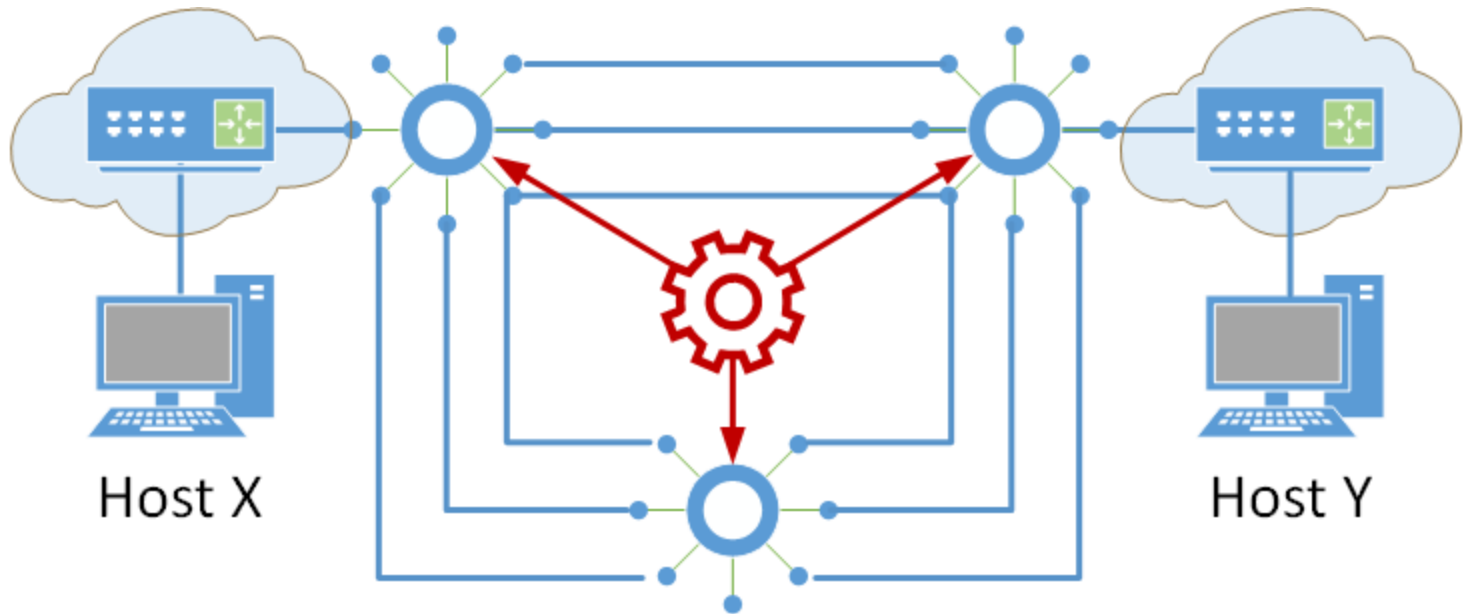
Deployment / Customer Reach

- Controlling a handful of IXPs is sufficient

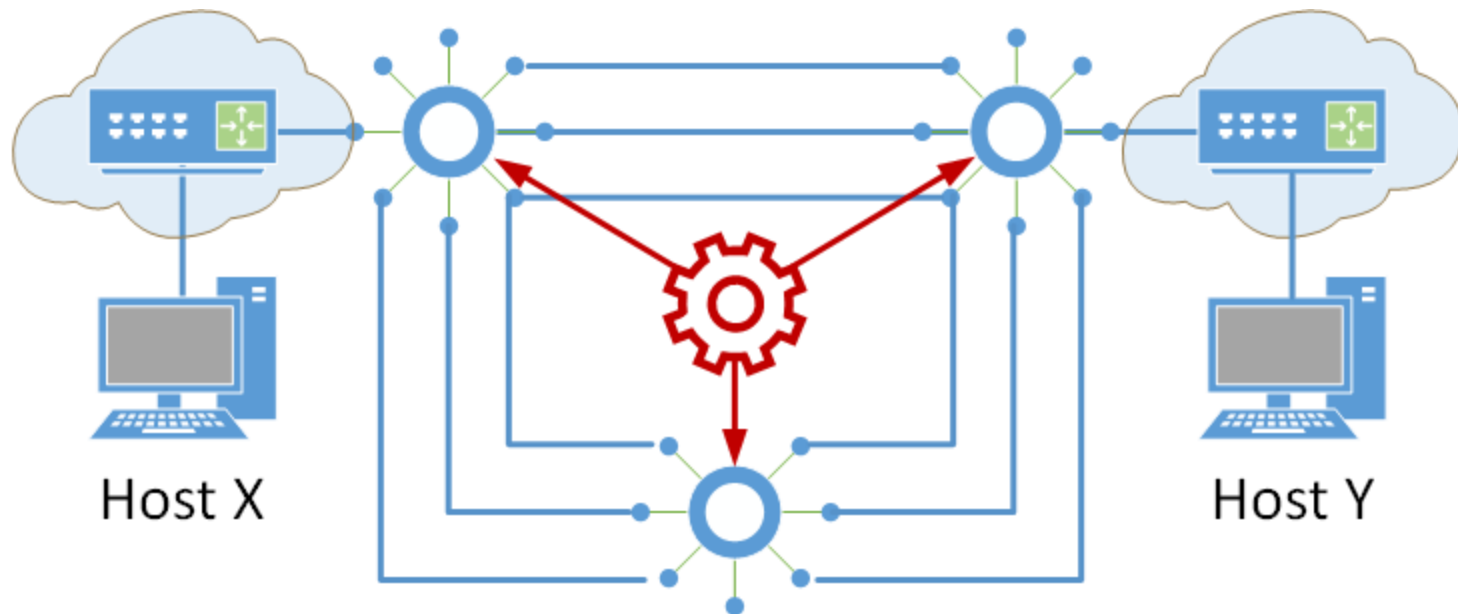


HOW TO EFFICIENTLY COMPUTE END-TO-END PATHS AT CXPS

Model

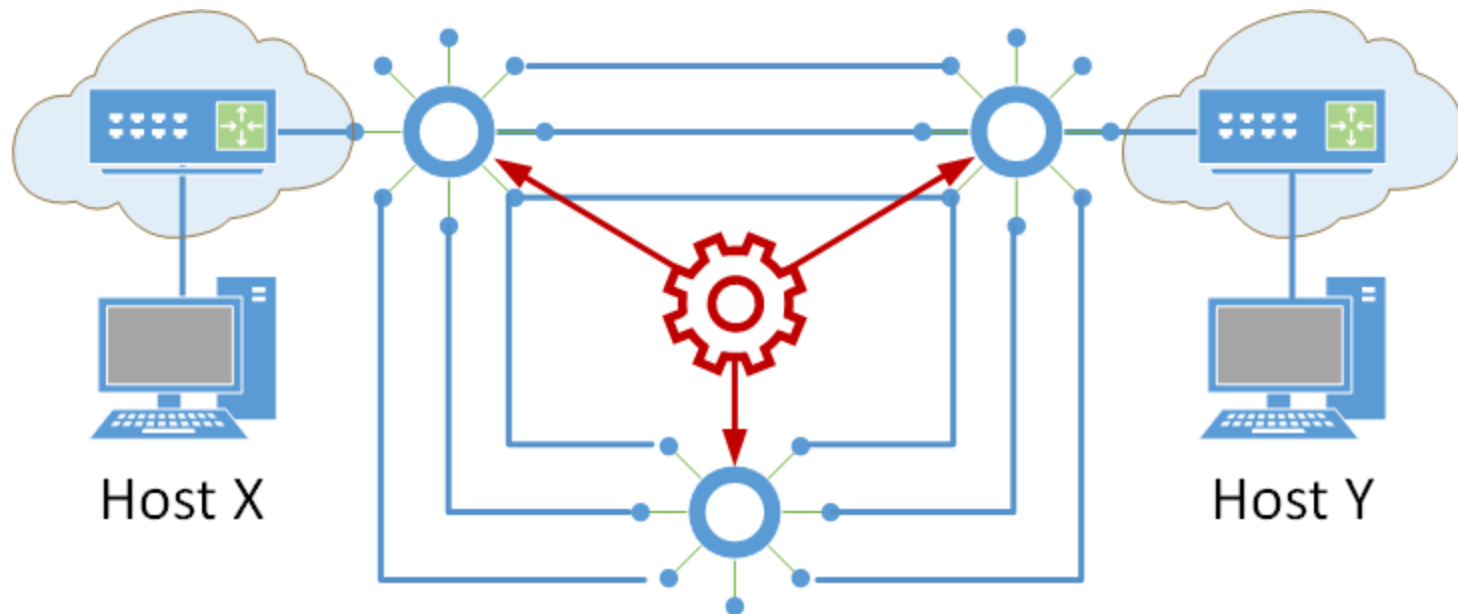


Model



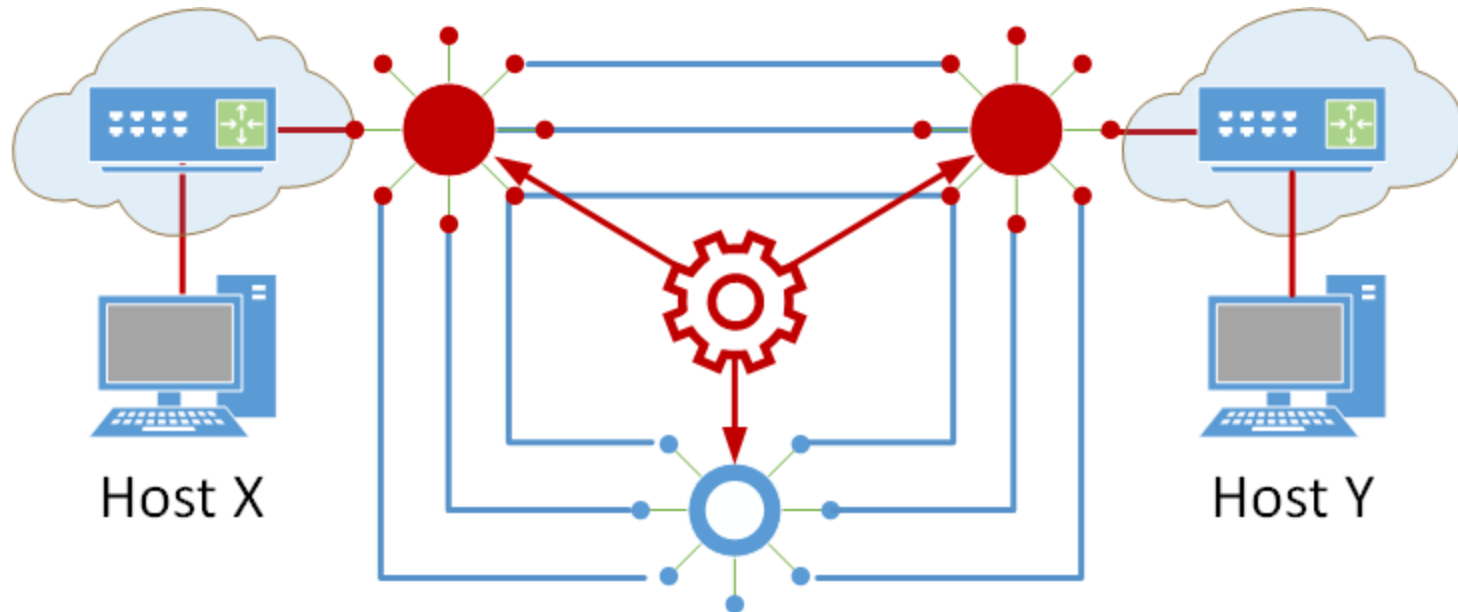
Hosts:	X,Y
Bandwidth:	50 Mbps
Max. Latency:	100 ms


Model



Hosts:	X,Y
Bandwidth:	50 Mbps
Max. Latency:	100 ms

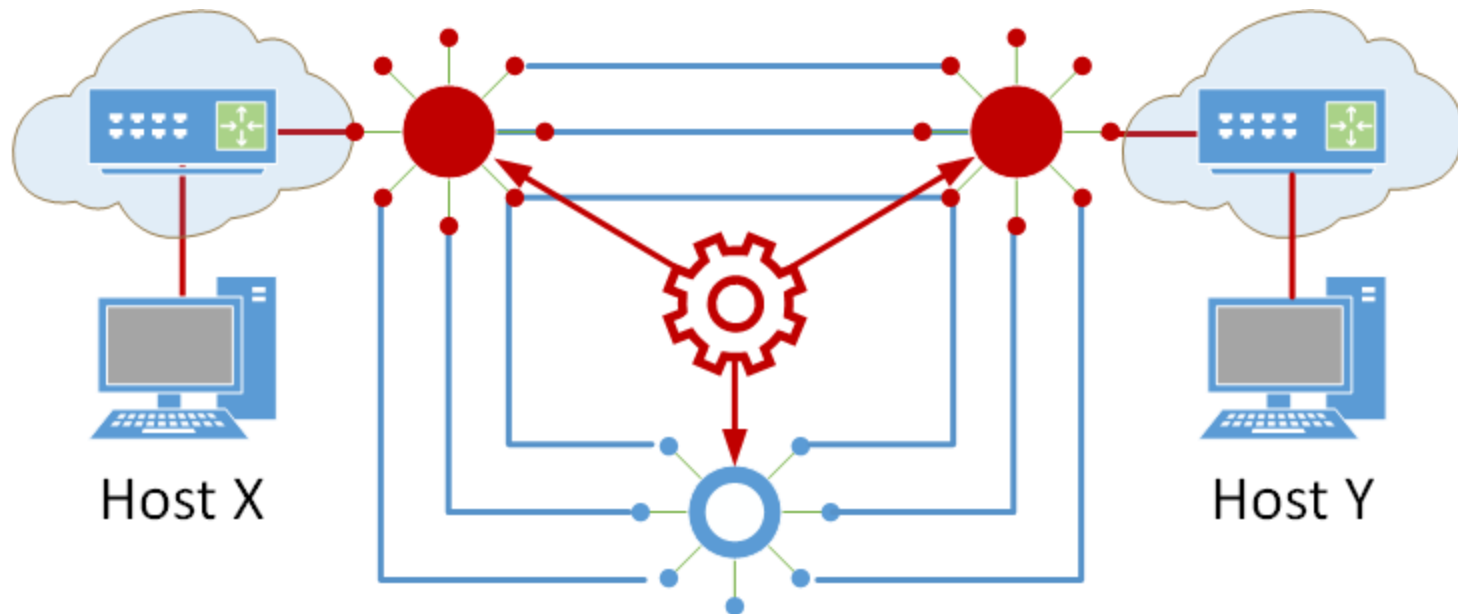
Model




	Hosts:	X,Y
	Bandwidth:	50 Mbps
	Max. Latency:	70 ms

Account for fixed latency to IXPs of X and Y

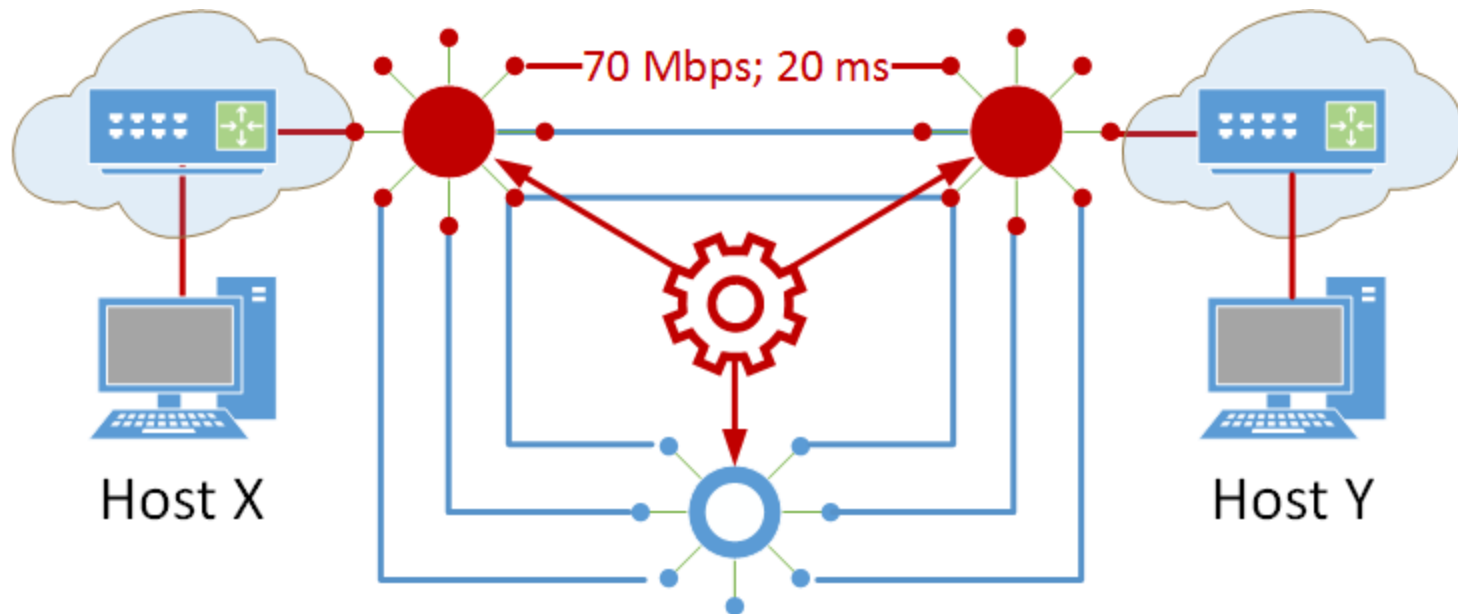
Model




	Hosts:	X,Y
	Bandwidth:	50 Mbps
	Max. Latency:	70 ms

Task: find appropriate path for connecting the IXPs

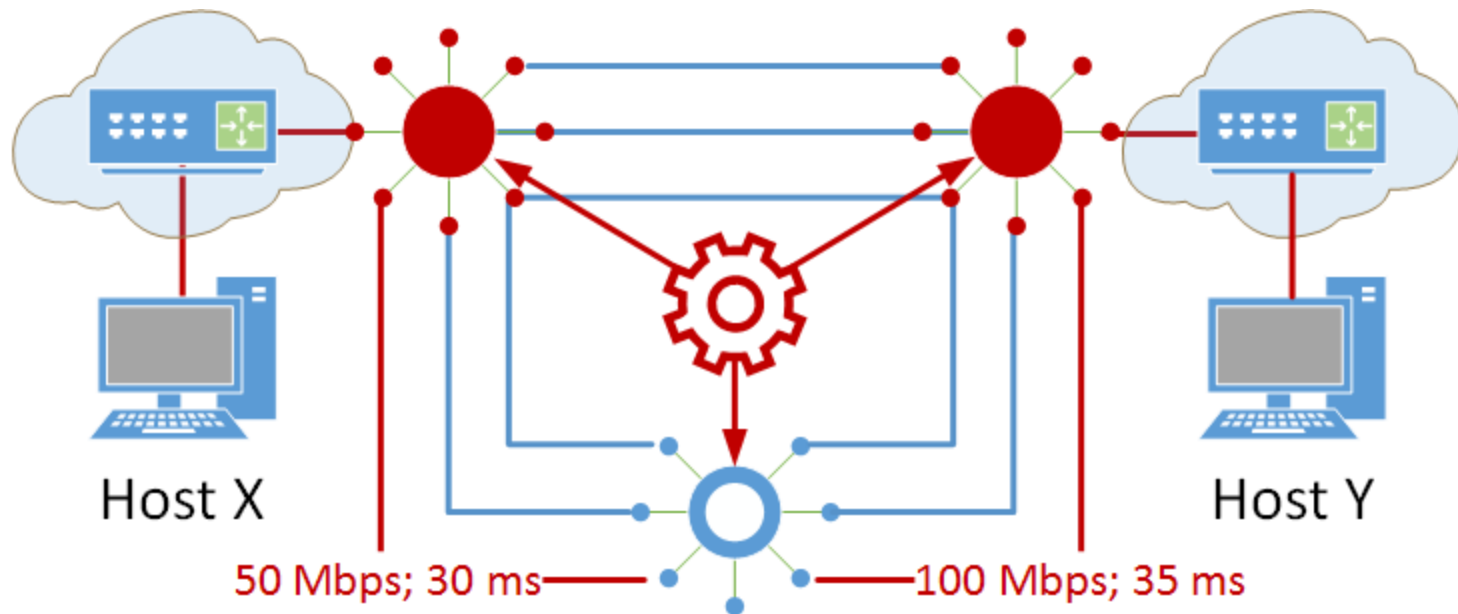
Model



	Hosts:	X,Y
	Bandwidth:	50 Mbps
	Max. Latency:	70 ms

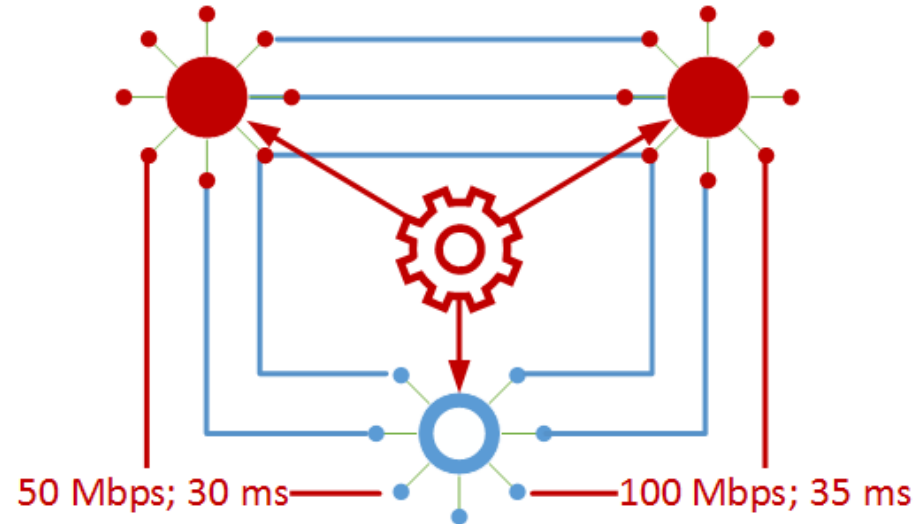
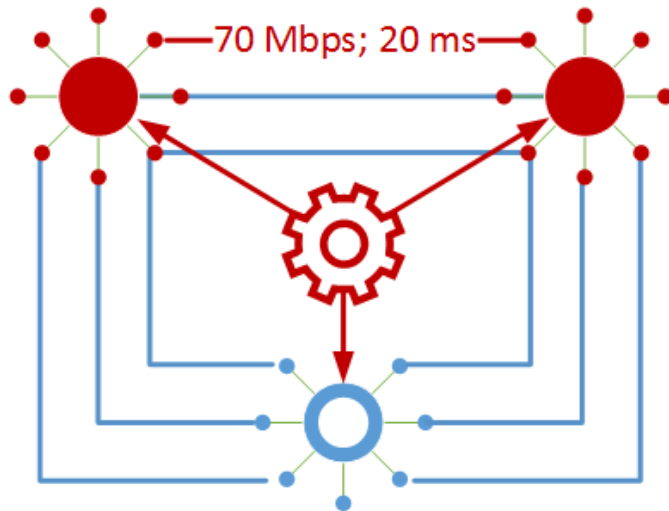
Task: find appropriate path for connecting the IXPs

Model



Hosts: X,Y
Bandwidth: 50 Mbps
Max. Latency: 70 ms

Trading Off Objectives



CXP should consider

- Minimize resource utilization
- Avoid resource fragmentation
- Avoid utilization of scarce resources

Finding Good Paths is Challenging

Theory

- Finding optimal paths is NP-hard when considering latency etc.!
- Feasible paths can be found in polynomial time.

Practice

- Even when only considering 14 IXPs, the IXP multigraph contains around 4k edges.

Finding Good Paths is Challenging

Theory

- Finding optimal paths is NP-hard when considering latency etc.!
- Feasible paths can be found in polynomial time.

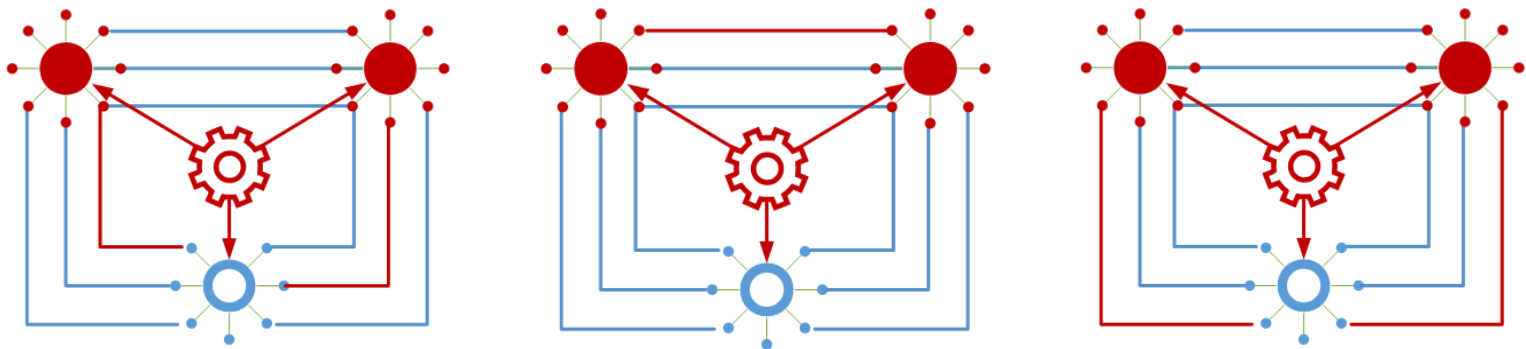
Practice

- Even when only considering 14 IXPs, the IXP multigraph contains around 4k edges.

Algorithmic Framework: Sample-Select

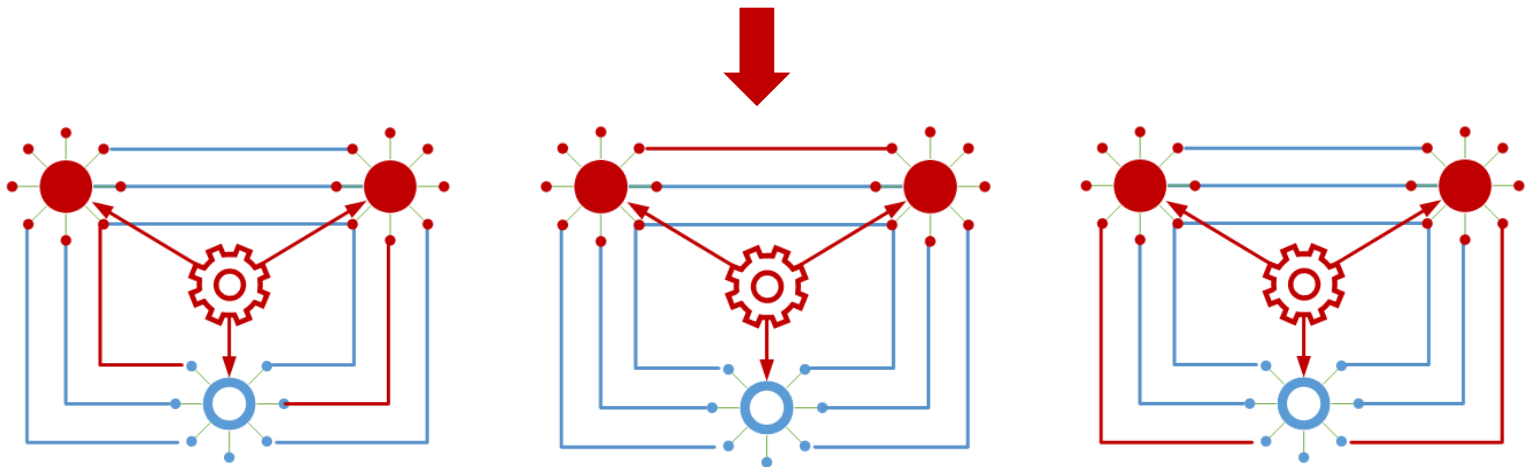
Sample-Select Framework

- Sample-Step:
 - Generate a set of different feasible paths.
- Select-Step:
 - Select one of the sampled paths according to a global objective.



Sample-Select Framework

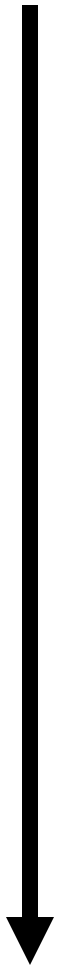
- Sample-Step:
 - Generate a set of different feasible paths.
- Select-Step:
 - Select one of the sampled paths according to a global objective.



Path Sampling Strategies

- Perturbed Dijkstra (PD)
 - project inter-IXP links on the lowest latency one and apply Dijkstra
 - Iterate while not considering previously used links
- Guided Dijkstra (GD)
 - Dijkstra choosing a single inter-IXP link at random during neighborhood exploration
- Guided Walk (GW)
 - Choose next IXP node and the respective edge uniformly at random

randomization



Path Selection Strategy

- Strictly prefer paths with smaller hop count
- Break ties by ...
 - trying to avoid using scarce low latency links
 - trying to avoid depleting bandwidth between adjacent IXPs

Reconfiguration Support

- Given previously sampled paths, we propose an Integer Program HeurPaths for computing re-optimizations.
- HeurPaths can select to reconfigure any existing path embeddings to enable the embedding of additional requests.

Integer Program 2: Heuristic Path Formulation (HeurPaths)

$$\max \sum_{R \in \mathcal{R}} x_R \quad (\text{OBJ})$$

$$x_R = \sum_{P_R \in \mathcal{P}_R} y_{P_R} \quad \forall R \in \mathcal{R} \quad (\text{HP-1})$$

$$\text{bw}_e \geq \sum_{R \in \mathcal{R}, e \in P_R} \text{bw}_R \cdot y_R \quad \forall e \in E_G \quad (\text{HP-2})$$

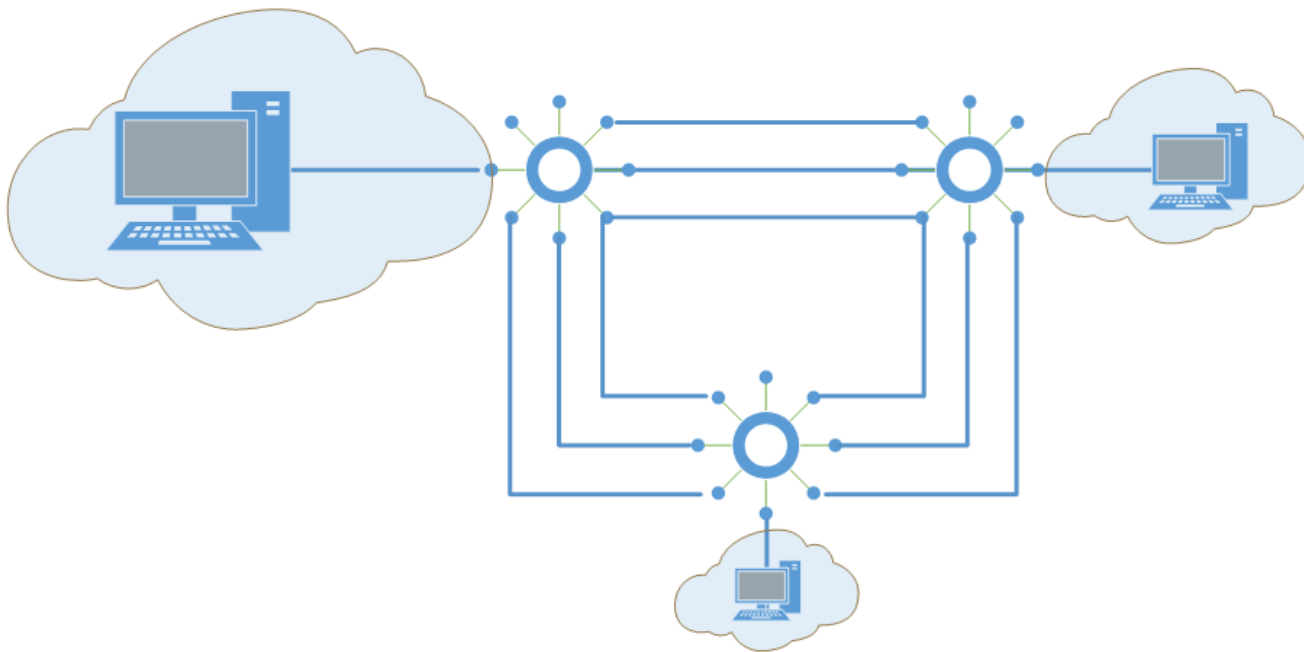
$$x_R \in \{0, 1\} \quad \forall R \in \mathcal{R} \quad (\text{HP-3})$$

$$y_{P_R} \in \{0, 1\} \quad \forall R \in \mathcal{R}, P_R \in \mathcal{P}_R \quad (\text{HP-4})$$

ALGORITHMIC EVALUATION

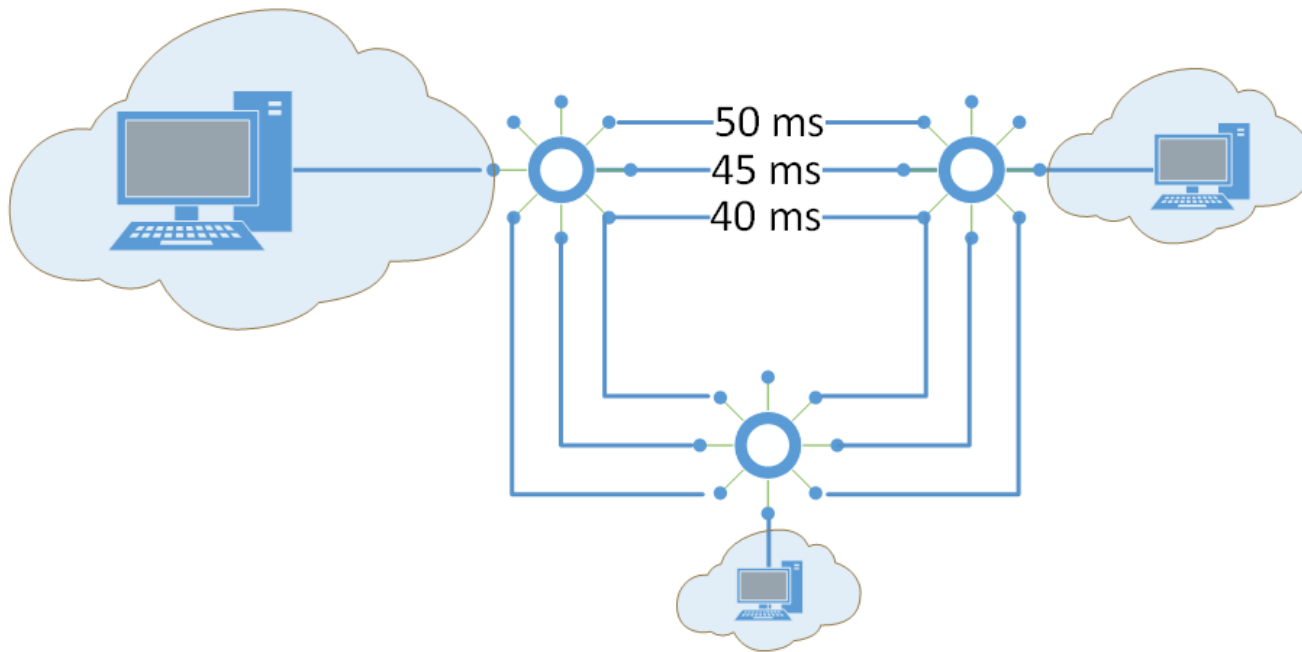
Simulation Setup

- Create IXP graph using incremental approach



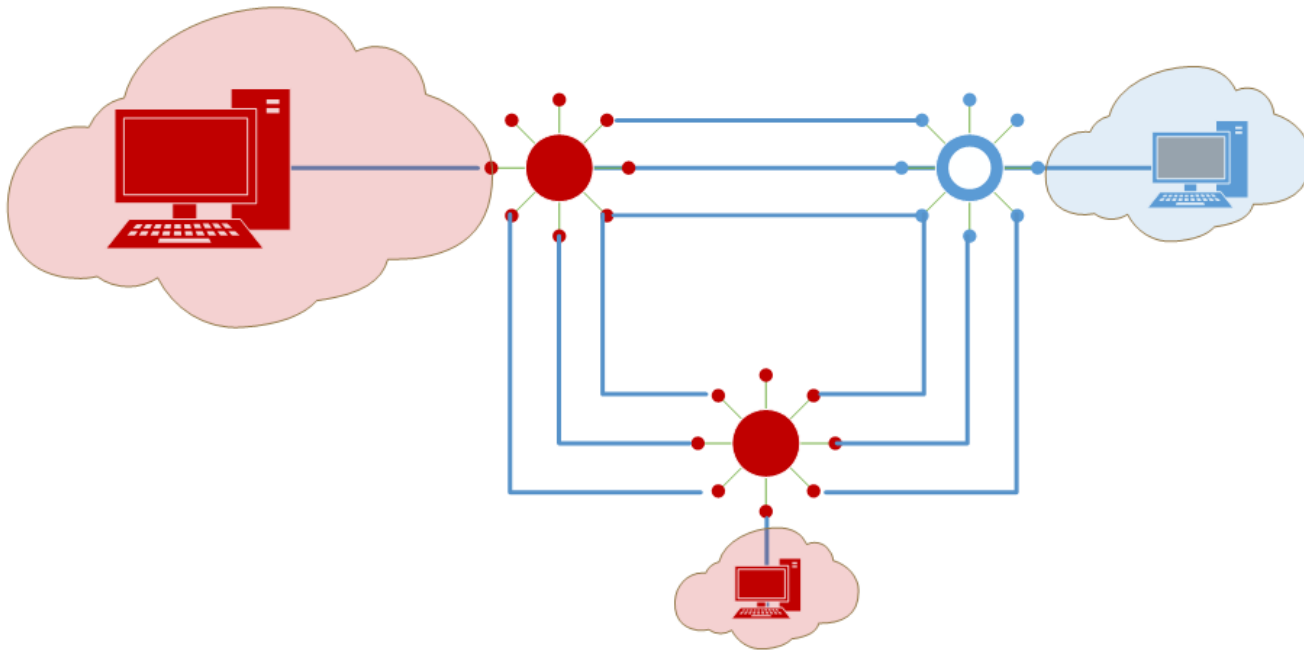
Simulation Setup

- Create IXP graph using incremental approach
 - unitary bandwidth across all links
 - latencies are estimated using geo. location + noise



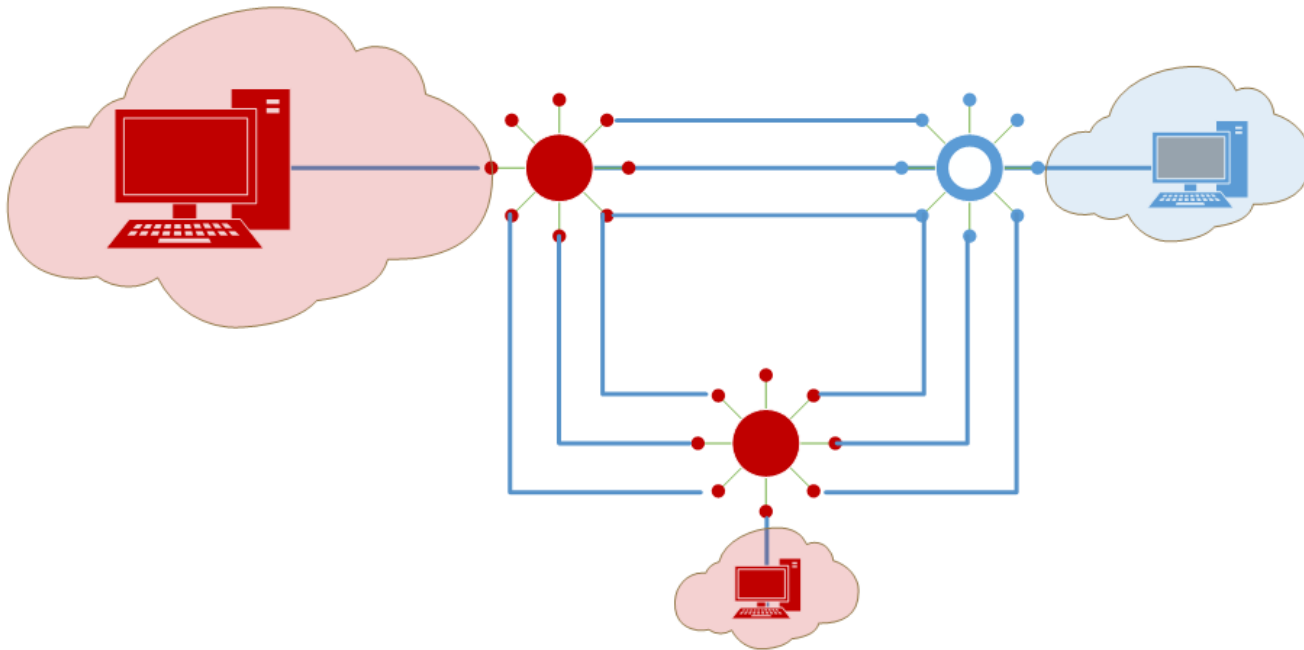
Simulation Setup

- Create IXP graph using incremental approach
- Draw IPv4 addresses to connect randomly
 - estimate latencies using estimated geo. distances + noise



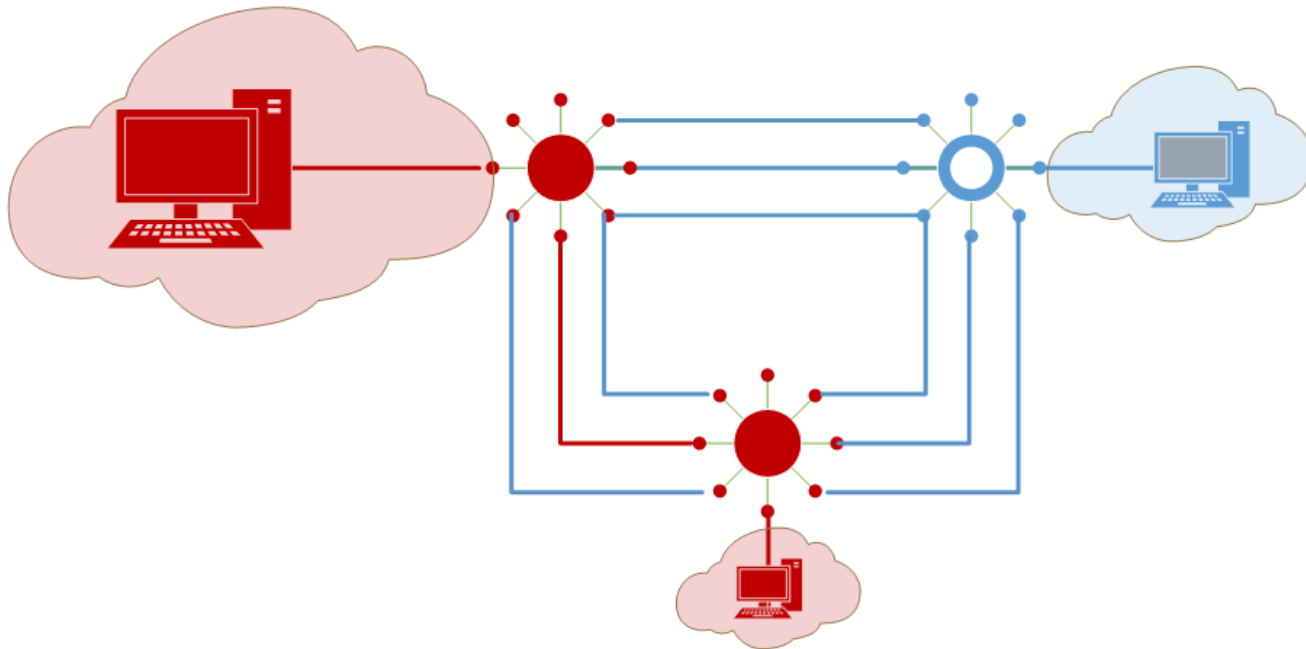
Simulation Setup

- Create IXP graph using incremental approach
- Draw IPv4 addresses to connect randomly
- Try to embed path using inter-IXP links



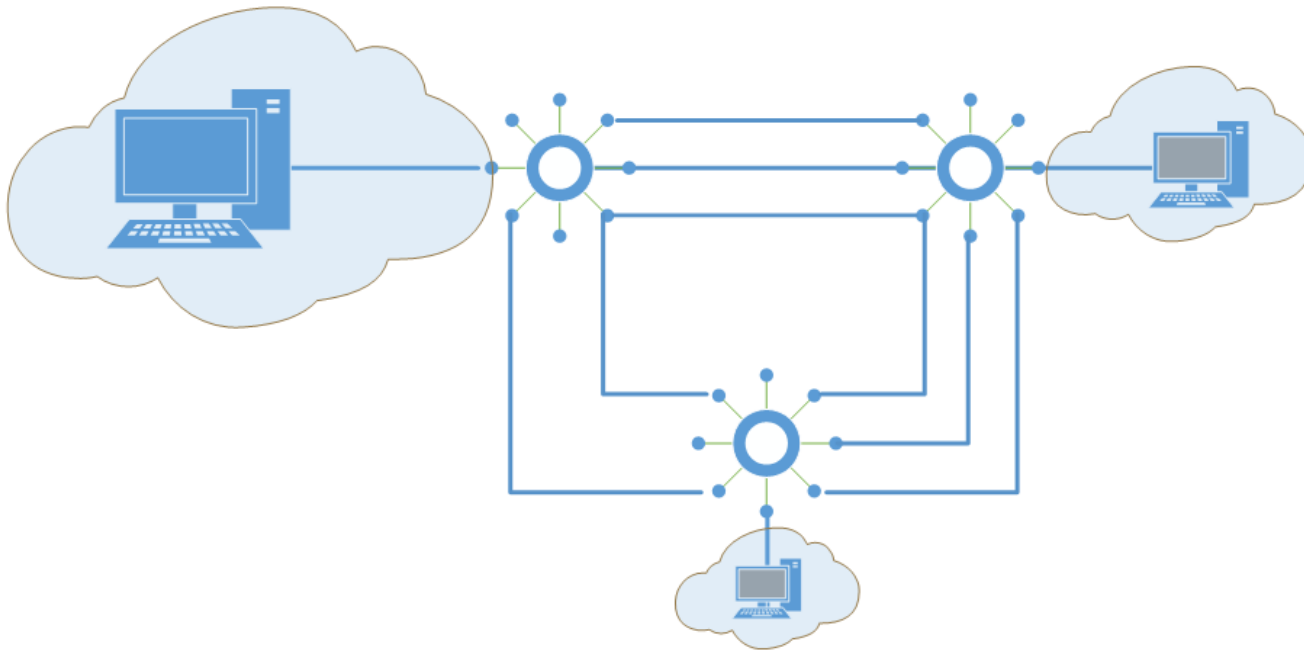
Simulation Setup

- Create IXP graph using incremental approach
- Draw IPv4 addresses to connect randomly
- Try to embed path using inter-IXP links



Simulation Setup

- Create IXP graph using incremental approach
- Draw IPv4 addresses to connect randomly
- Try to embed path using inter-IXP links
 - requests have unit bandwidth demand, fully utilizing links



Simulation Setup

- Create IXP graph using incremental approach
- Draw IPv4 addresses to connect randomly
- Try to embed path using inter-IXP links

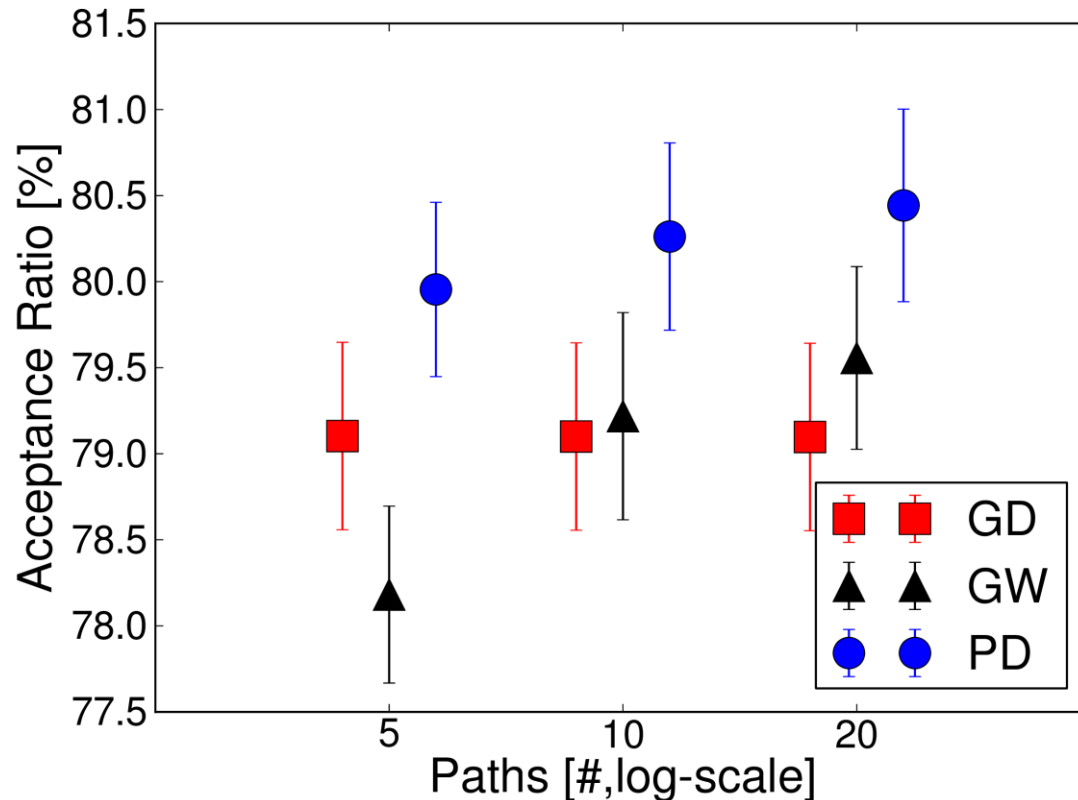
IXP nodes	7, 14, 28, 57, 115, 229
-----------	-------------------------

requests	10k
----------	-----

req. latency	(100,150), (150,200), (200, 250), (250,300)
--------------	---

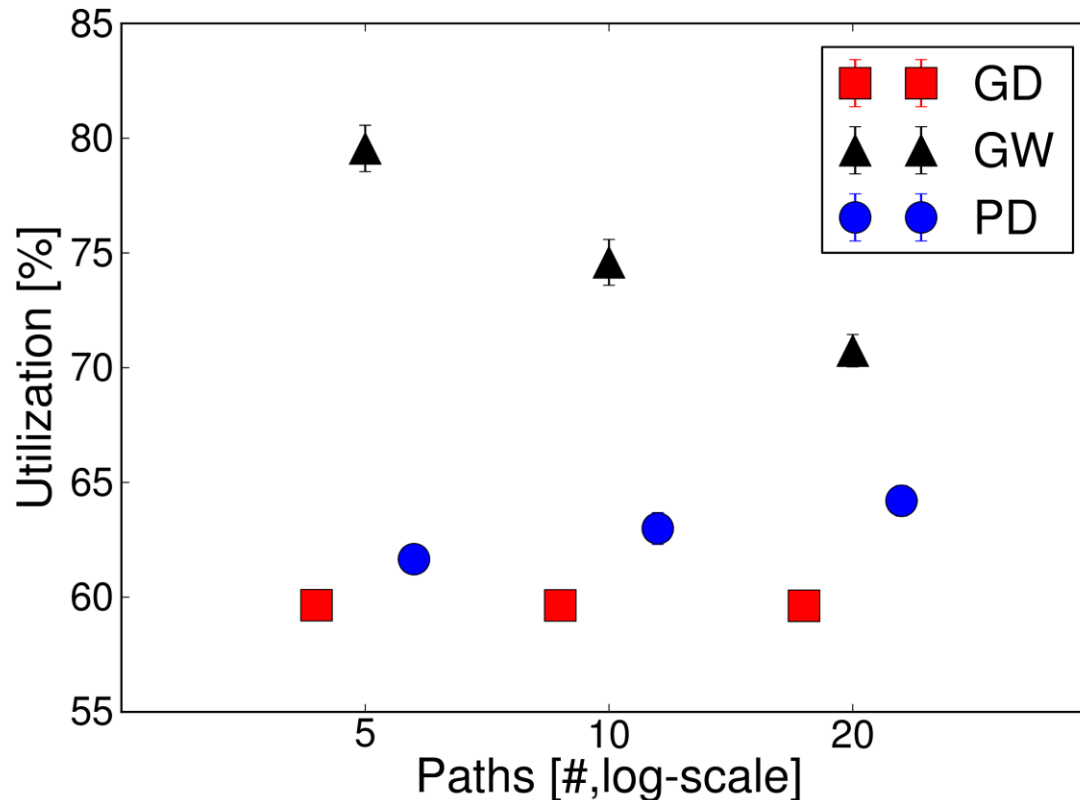
sampled paths	5, 10, 20
---------------	-----------

Online: Acceptance Ratio



- IXP Graph: 28 nodes, 6.5k edges
- Request latencies: (200, 250) ms

Online: Utilization

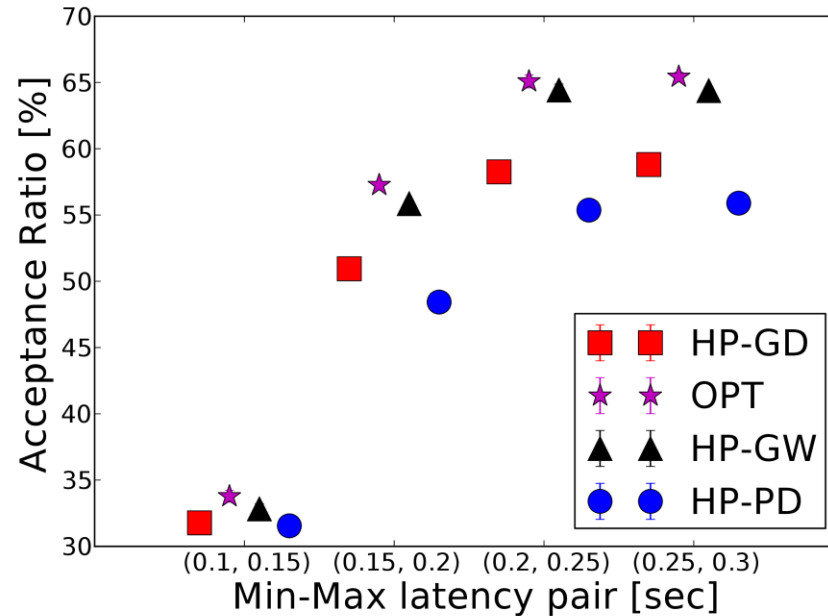


- IXP Graph: 28 nodes, 6.5k edges
- Request latencies: (200, 250) ms

Offline Scenario

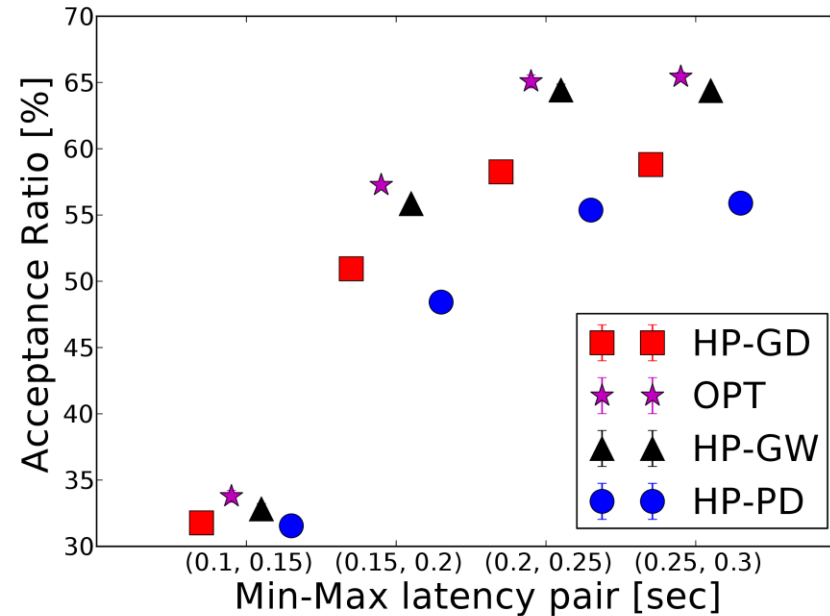
- We found minor performance gains using reconfigurations (approx. 1% improvement).
- Bundle path sampling with HeurPaths to embed 10k requests as a batch.
- Evaluate how well path sampling algorithms use the diverse IXP multigraph structure.
- Comparison with optimal Integer Programming solution.

Offline: Acceptance Ratio



- IXP Graph: 14 nodes, 3.9k edges
- 20 sampled paths / request

Offline: Acceptance Ratio



- IXP Graph: 14 nodes, 3.9k edges
- 20 sampled paths / request

Average runtime: 20 seconds for 10k requests

Summary

Measurement

- Stitching paths via IXPs offers new opportunities
- Handful of IXPs suffice to reach many customers

Algorithms

- Centralized orchestration is possible at scale
- Optimal re-optimizations feasible

Thanks!

**Code (Apache 2.0) available at
https://bitbucket.org/vkotronis/cxp_experimentation**