In-band Network Telemetry: Network Load Balancing Use-cases

Presenter: Mukesh Hira, VMware

Slide content credits: Co-authors of HULA and CLOVE papers

Naga Katta (Salesforce.com), Aditi Ghag (Vmware), Aran Bergmann (Technion), Isaac Keslassy (Technion),

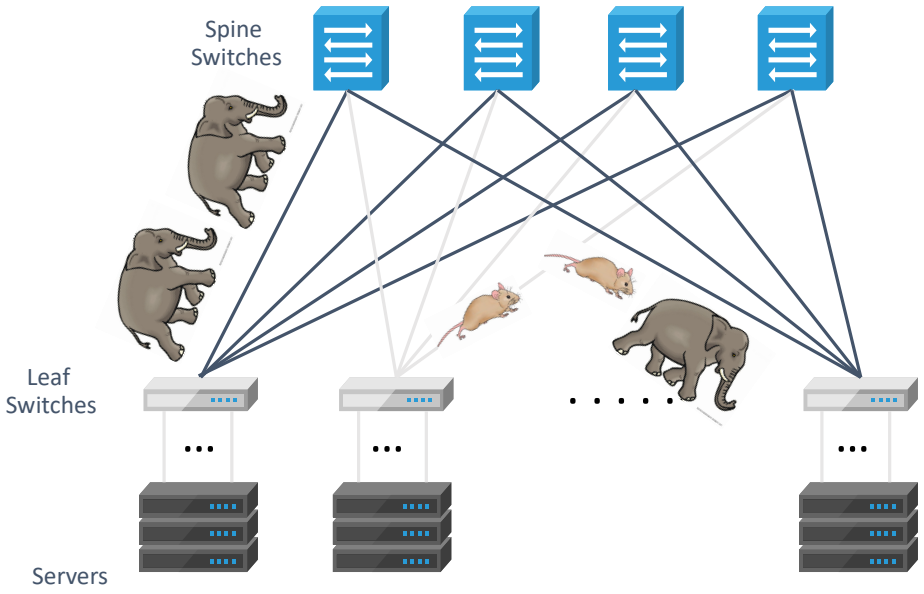
Changhoon Kim (Barefoot Networks), Anirudh Sivaraman (NYU), Jennifer Rexford (Princeton University)

Predominant Data center Network Load Balancing

Equal-**C**ost **M**ulti-**P**ath (**ECMP**) routing:

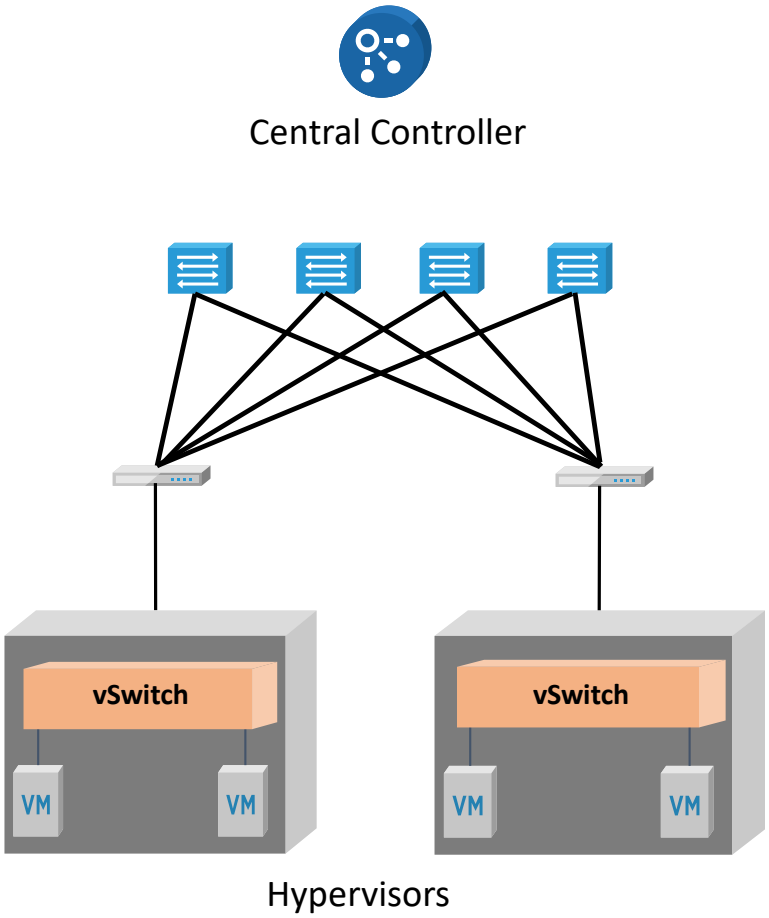
- Path(packet) = hash(packet's 5-tuple)

src,dst IP + src,dst port + protocol



- Coarse-grained
- Elephant Hash collisions
- Congestion-oblivious

Alternative Proposals

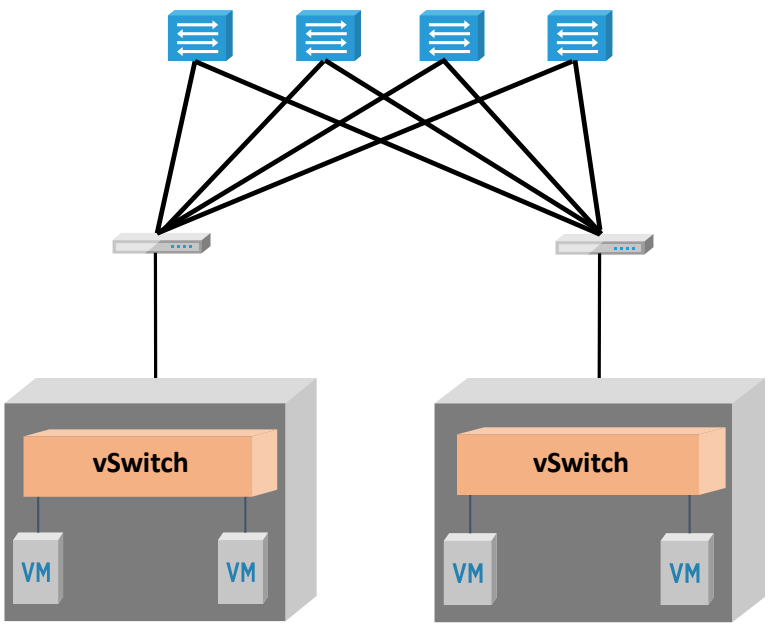


Centralized Load Balancing

Hedera, Fastpass, MicroTE, SWAN

- Control-driven feedback
- Slow reaction time
- Scalability Issues

Alternative Proposals

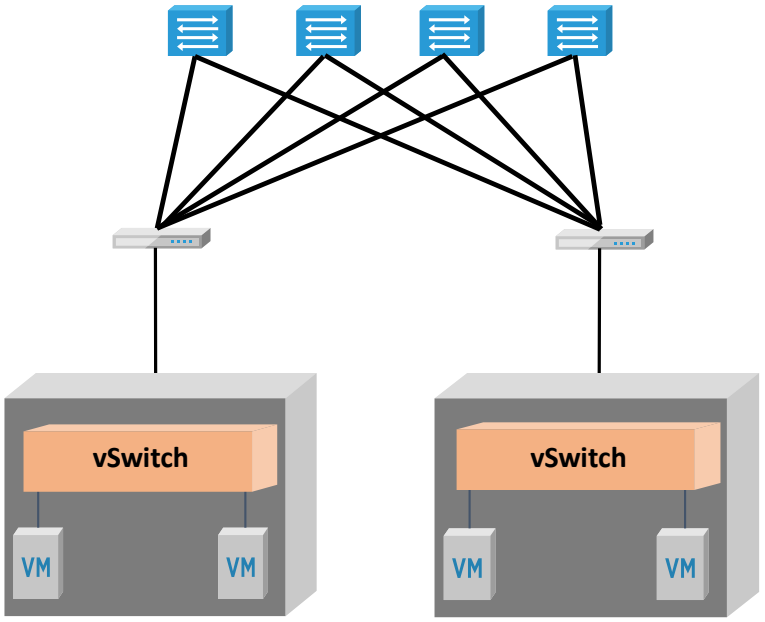


In-network load balancing

CONGA, LetFlow, DRILL

- Needs custom ASIC data center fabric
- High capital cost
- Controller Involvement may still be required

Alternative Proposals



End-host load balancing

Presto

- Congestion Oblivious
- Controller intervention in case of topology asymmetry

MPTCP

- Incast collapse
- Guest VM network stack changes

Congestion Aware Network Load Balancing based on INT

- Two approaches

  - In-Network Network Load Balancing at each Hop
    HULA: Scalable Load Balancing using Programmable Data Planes
    (**H**op-by-Hop Link **U**tilization Aware **L**oad Balancing **A**rchitecture)

  - Congestion Aware Network Load Balancing from the Edge
    CLOVE: **C**ongestion-aware **Lo**ad Balancing at the **V**irtual **E**dge

HULA: Scalable Load Balancing using Programmable Data Planes (**H**op-by-Hop Link **U**tilization Aware **L**oad Balancing **A**rchitecture)

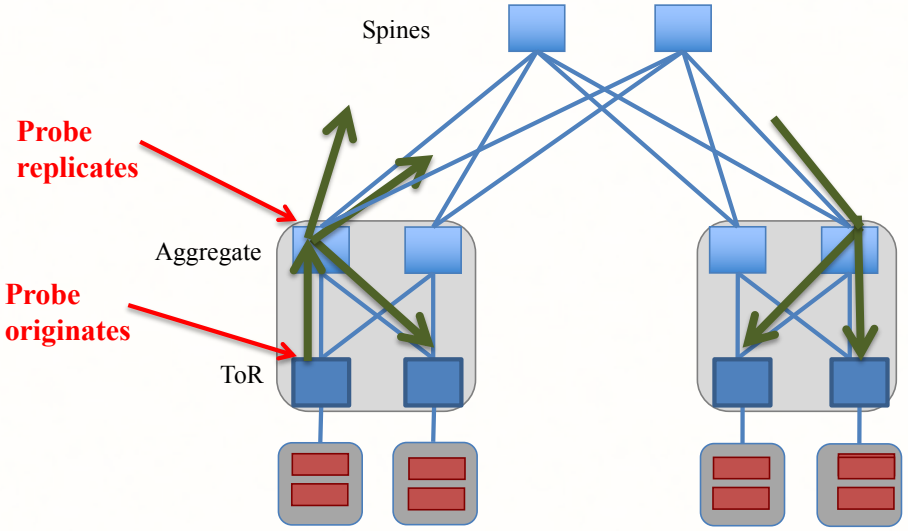
Naga Katta (Princeton University), Mukesh Hira (VMware), Changhoon Kim (Barefoot Networks),
Anirudh Sivaraman (MIT), Jennifer Rexford (Princeton University)

ACM SOSR 2016

HULA: Key Benefits

- Works in any network topology
  - Scalability via summarization of state
  - Only propagate network state for best next-hop upstream
  - Scales to any number of network tiers, any number of destinations
    - Subject to forwarding table size limitations
- Leverages INT to gather network state
- No vendor-proprietary state exchange protocol

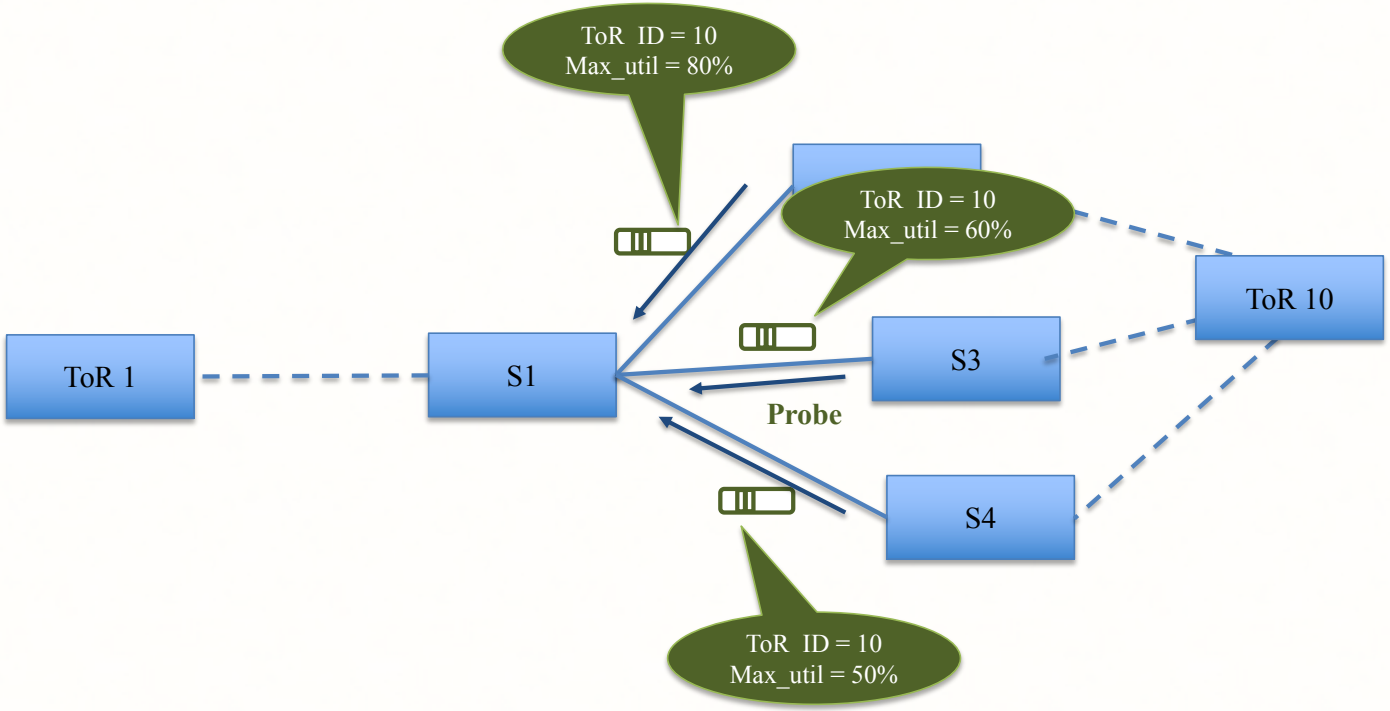
1. Probes accumulate path utilization in forward direction



P4 primitives
New header format
Programmable Parsing
Switch metadata

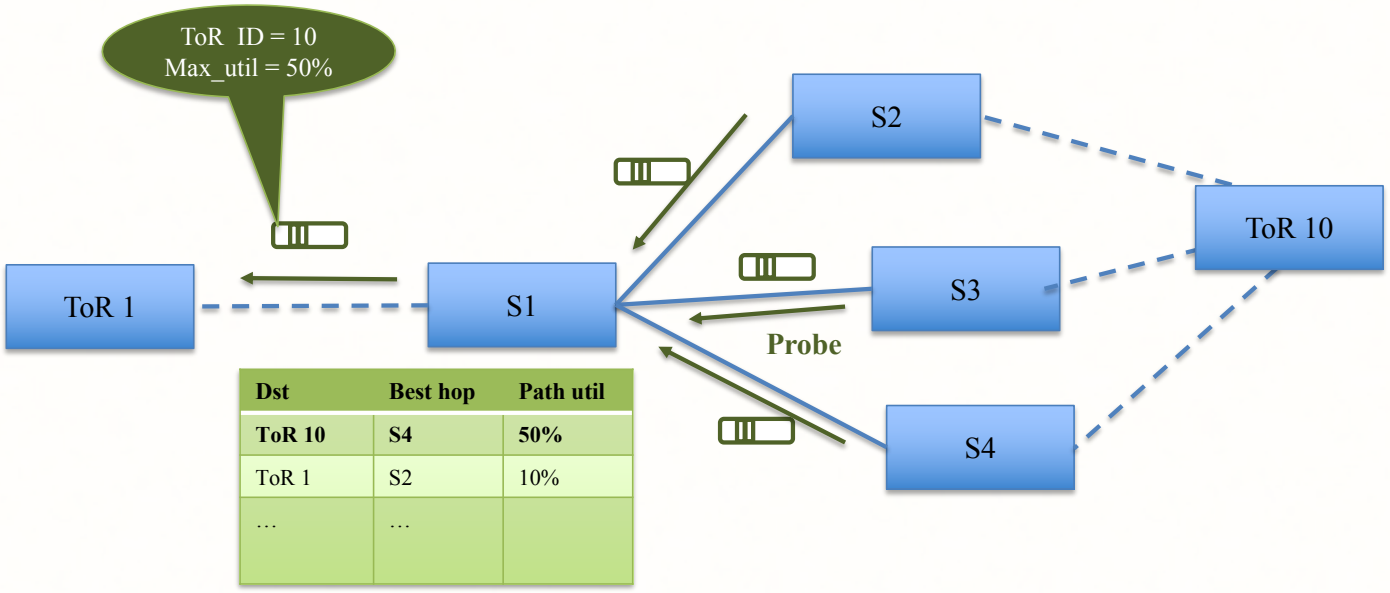
- Probes originate at each ToR
- Replicated at intermediate hops
- Path Utilization collected using INT

1. Forward direction path information is piggybacked onto reverse direction probes



Probe carries INT instructions to collect path state in its direction and previously accumulated state for opposite direction from probes received at probe-originating ToR

2. Switch identifies best next-hop to each ToR

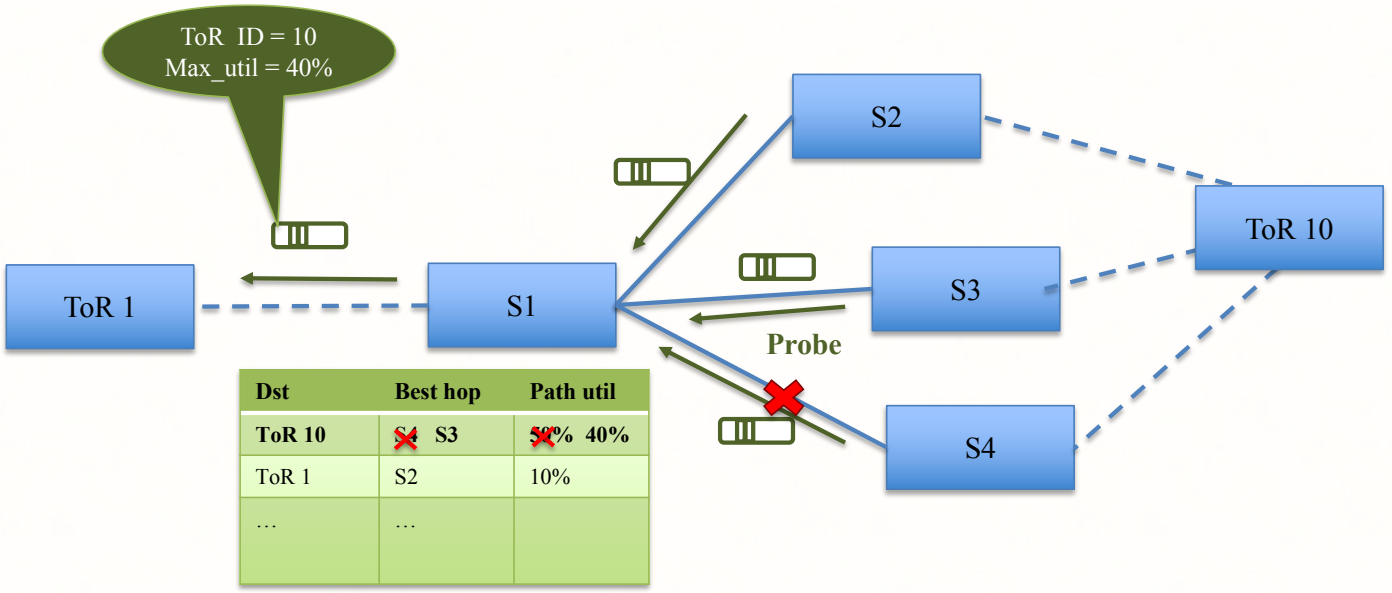


Dst	Best hop	Path util
ToR 10	**S4**	**50%**
ToR 1	S2	10%
…	…	

Best hop table

End-host to ToR mapping known via out of band mechanism

2. Switch identifies best next-hop to each ToR

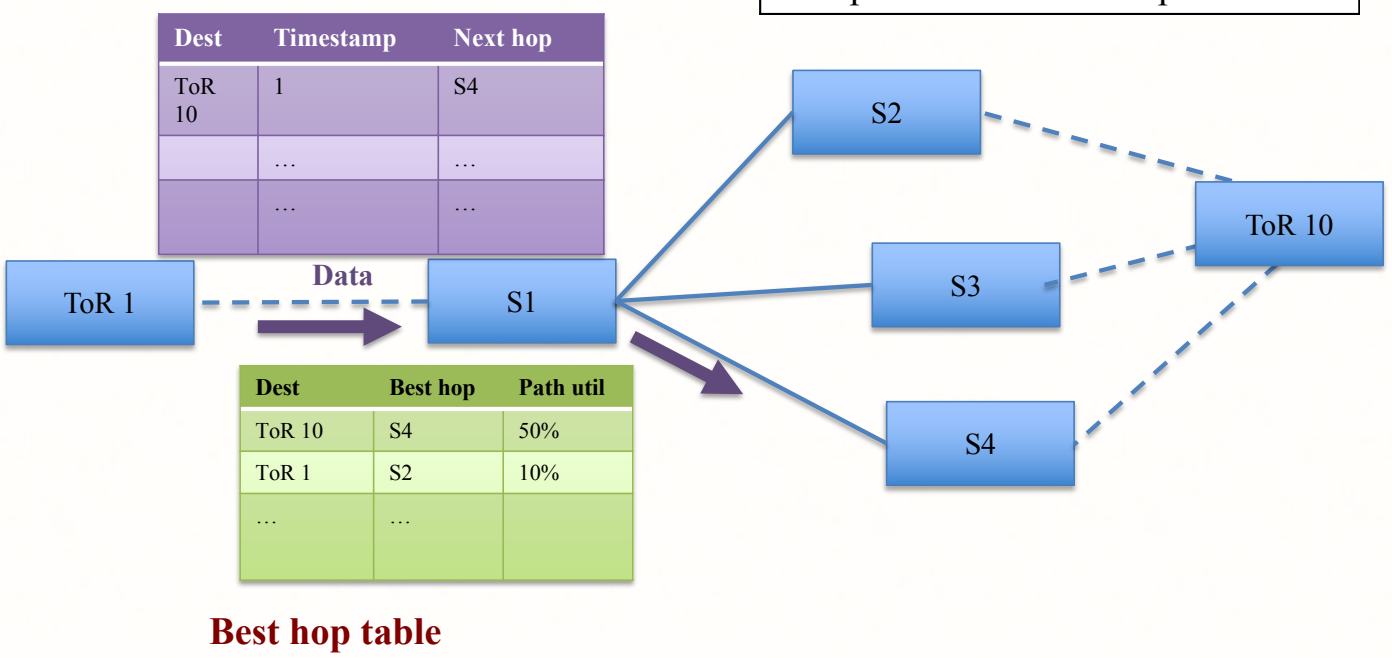


Dst	Best hop	Path util
ToR 10	~~S4~~ S3	~~50%~~ 40%
ToR 1	S2	10%
…	…	

Best hop table

3. Switches load balance flowlets based on real-time path utilization

P4 primitives
RW access to stateful memory
Comparison/arithmetic operators



Evaluation Setup

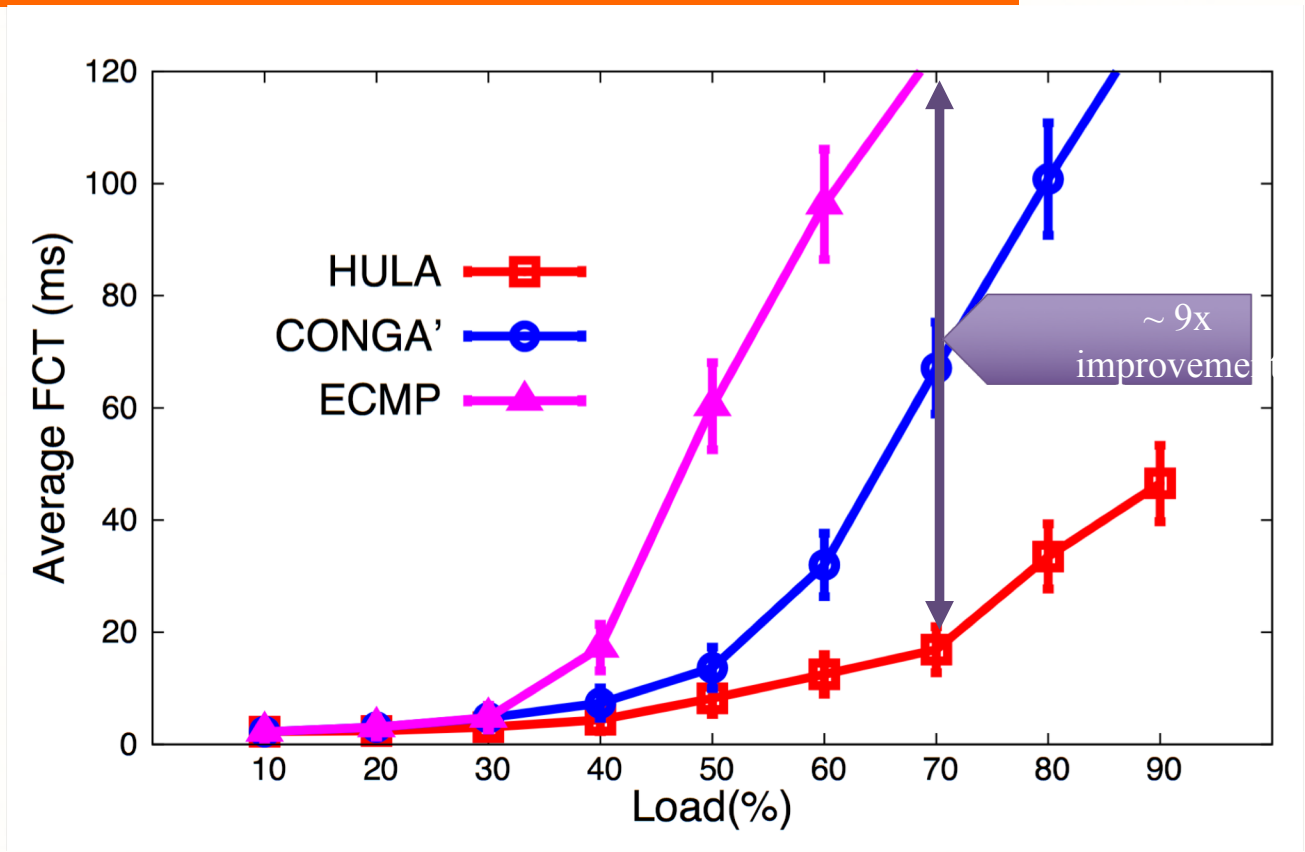
- NS2 packet-level simulator
- RPC-based workload generator
  - Empirical flow size distributions
  - Websearch and Datamining
- End-to-end metric
  - Average Flow Completion Time (FCT)

Compared with

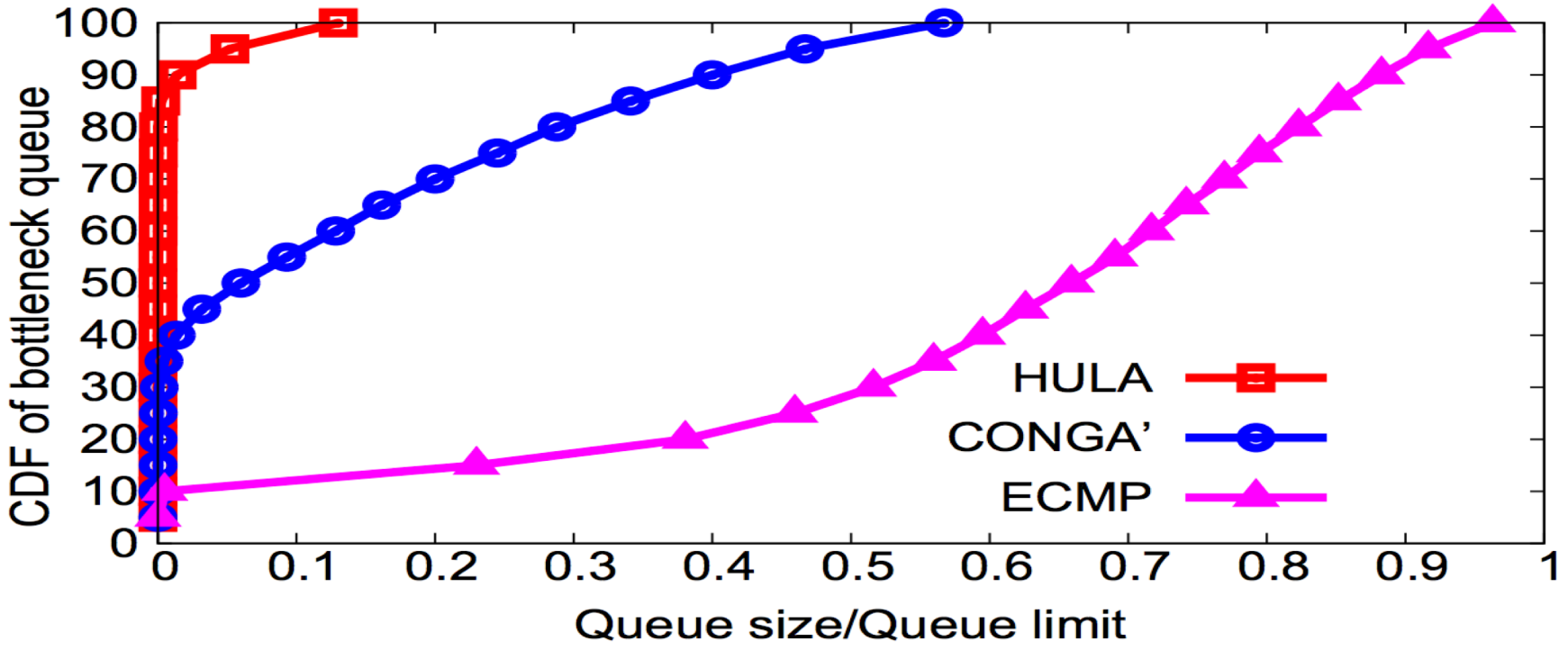
- ECMP
  - Flow level hashing at each switch
- CONGA'
  - CONGA within each leaf-spine pod
  - ECMP on flowlets for traffic across pods[1]

1. Based on communication with the authors

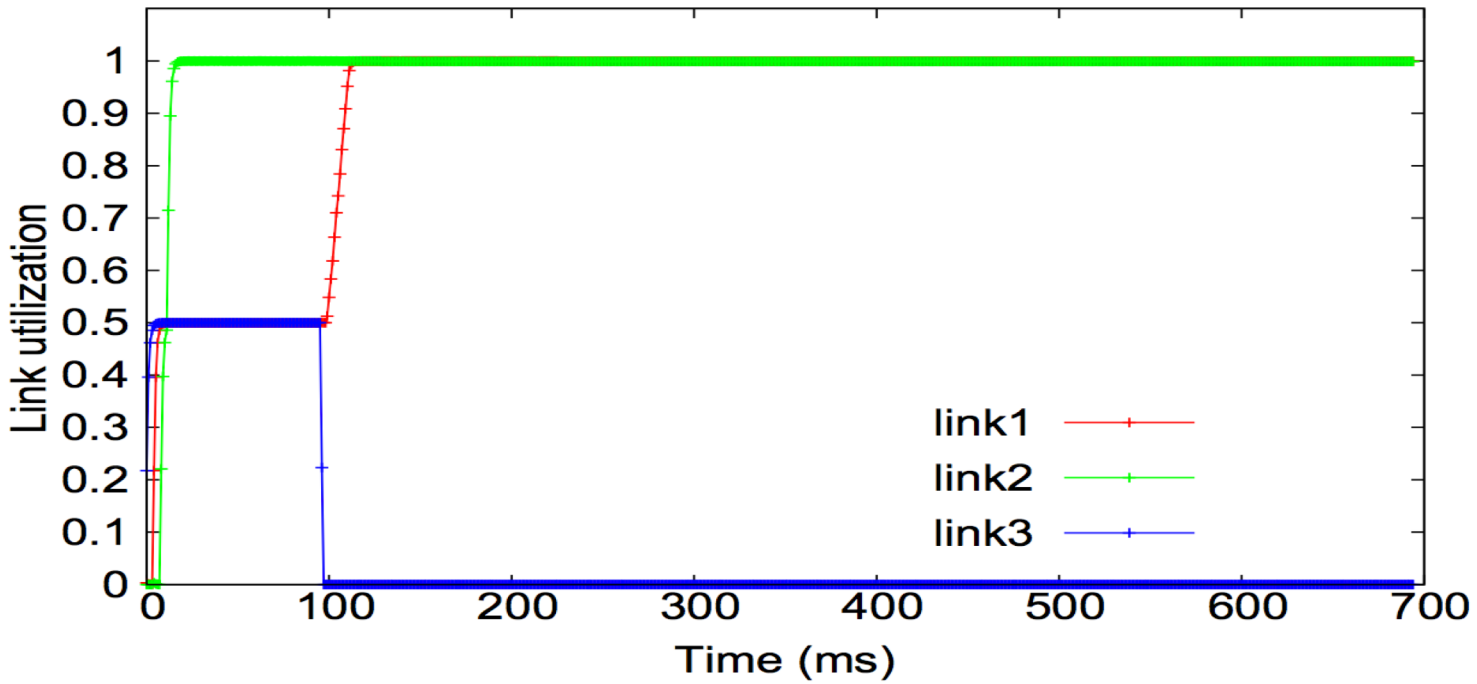
HULA handles high load much better



HULA keeps queue occupancy low



HULA is stable on link failure



Clove: Congestion-Aware Load Balancing at the Virtual Edge

Naga Katta[1,2], Aditi Ghag[3], Mukesh Hira[3], Isaac Keslassy[3,4],

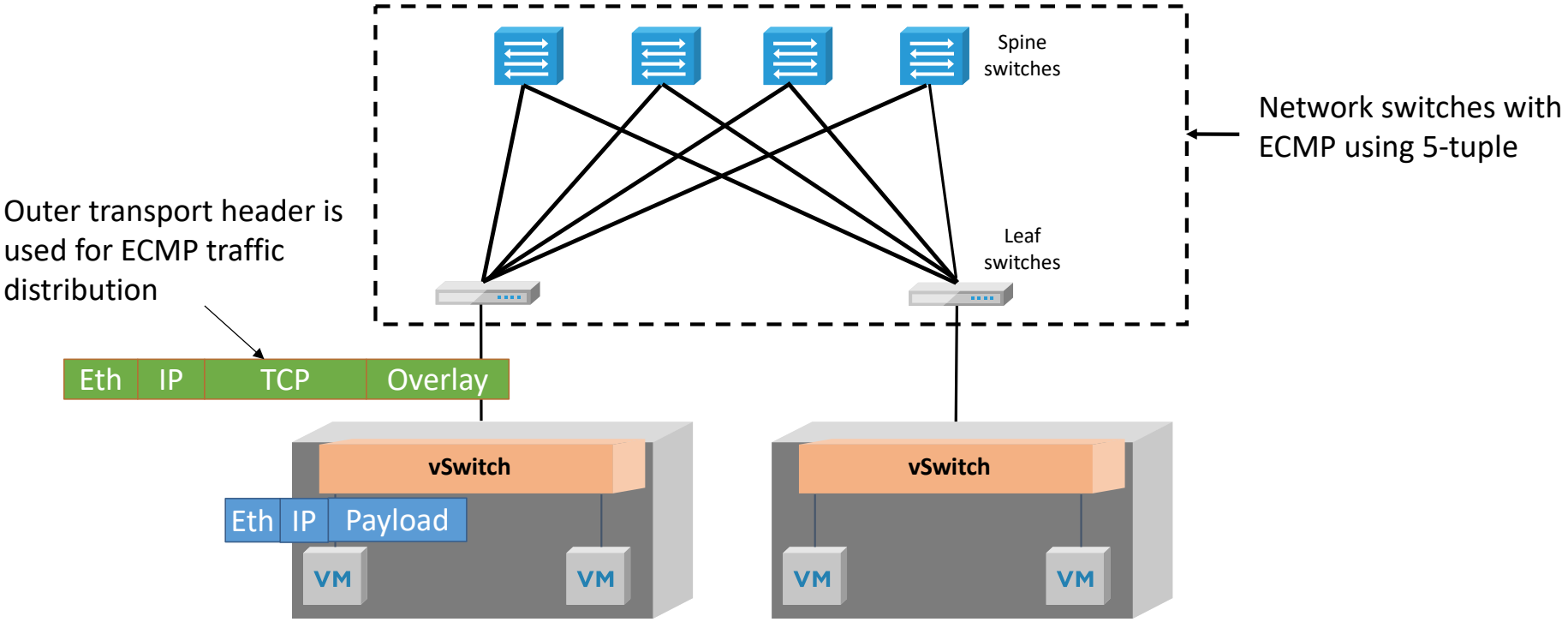
Aran Bergman[3,4], Changhoon Kim[5], Jennifer Rexford[2]

[1] *Salesforce.com* [2] *Princeton University* [3] *VMware* [4] *Technion* [5] *Barefoot Networks*

HotNets 2016, CoNEXT 2017

CLOVE assumptions

- Clove operates over a DC Overlay – e.g., Stateless Transport Tunneling (STT)



CLOVE in 1 slide

- **Path discovery using traceroute probes**
- **Load-balancing of flowlets [FLARE ‘05]**
- **vSwitch path selection based on RTT-scale feedback**
  - **E**xplicit **C**ongestion **N**otification **- ECN**
  - **I**n-band **N**etwork **T**elemetry **- INT**

Path Discovery

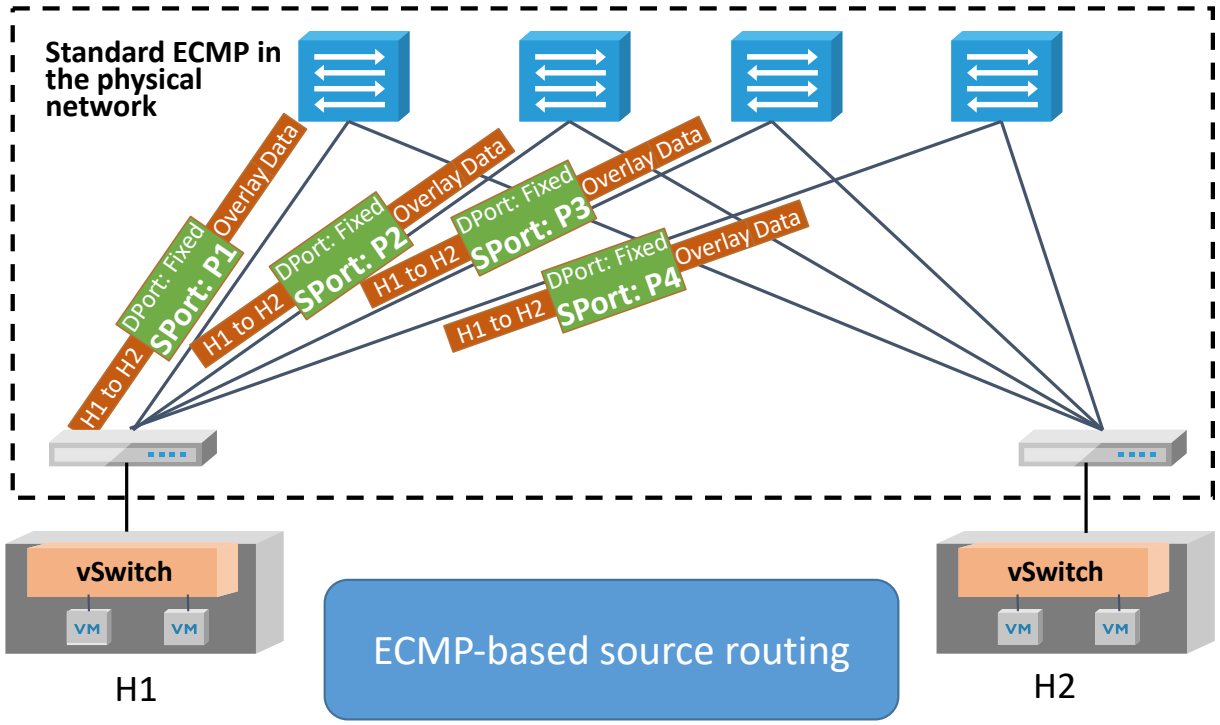
Load balancing flowlets

vSwitch Load balancing

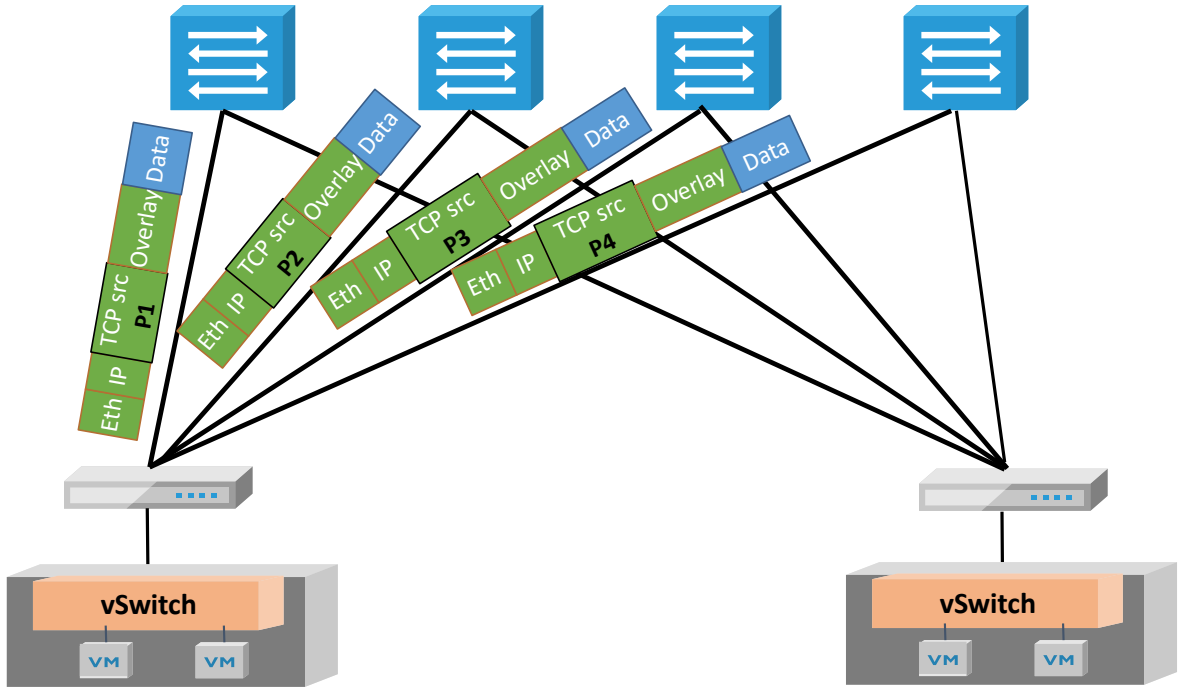
- Outer transport source port (with ECMP) maps to network path

Hypervisor learns source port to path mapping

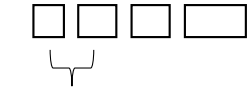
Dst	SPort
H2	P1
H2	P2
H2	P3
H2	P4



Scheme 1: Edge-Flowlet

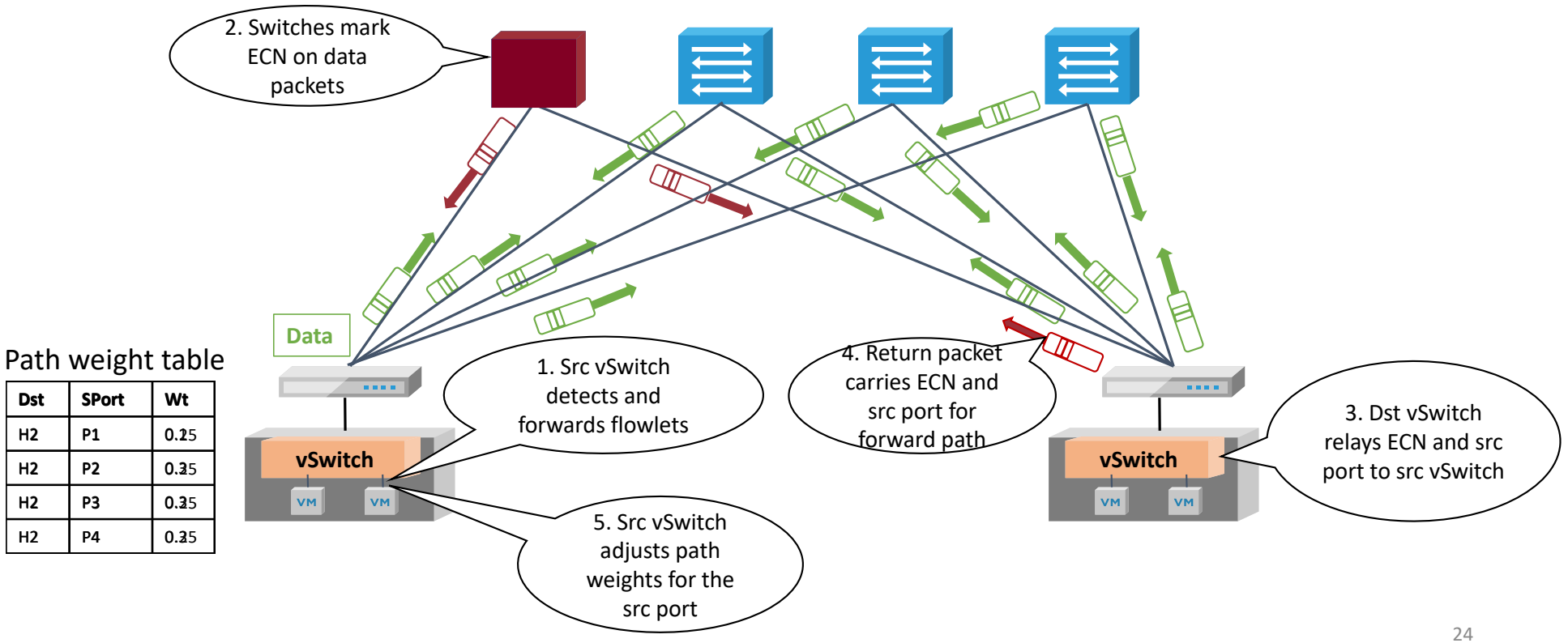


Dst	SPort
H2	P1
H2	P2
H2	P3
H2	P4



vSwitch Load balancing
Scheme2: CLOVE-ECN

- Congestion-aware balancing based on ECN feedback



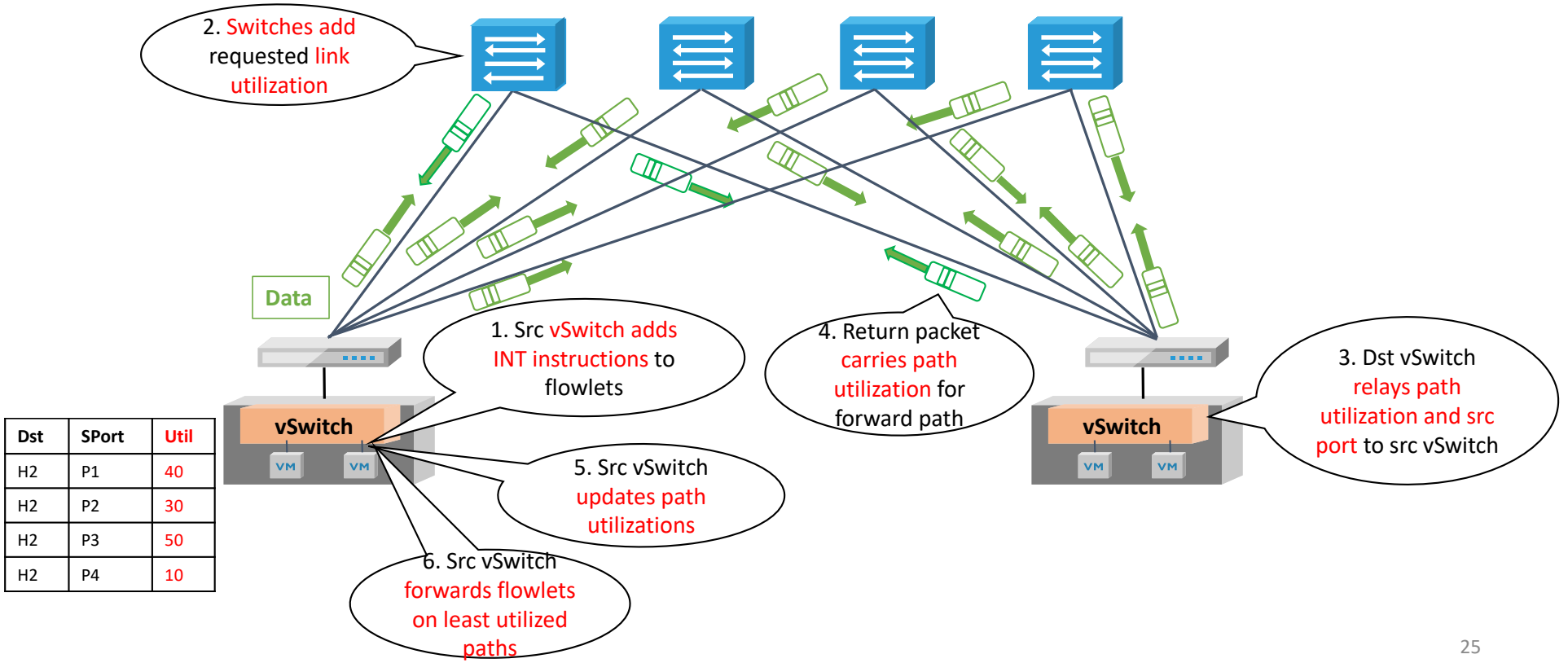
Path weight table

Dst	SPort	Wt
H2	P1	0.25
H2	P2	0.25
H2	P3	0.25
H2	P4	0.25

Path Discovery Load balancing flowlets

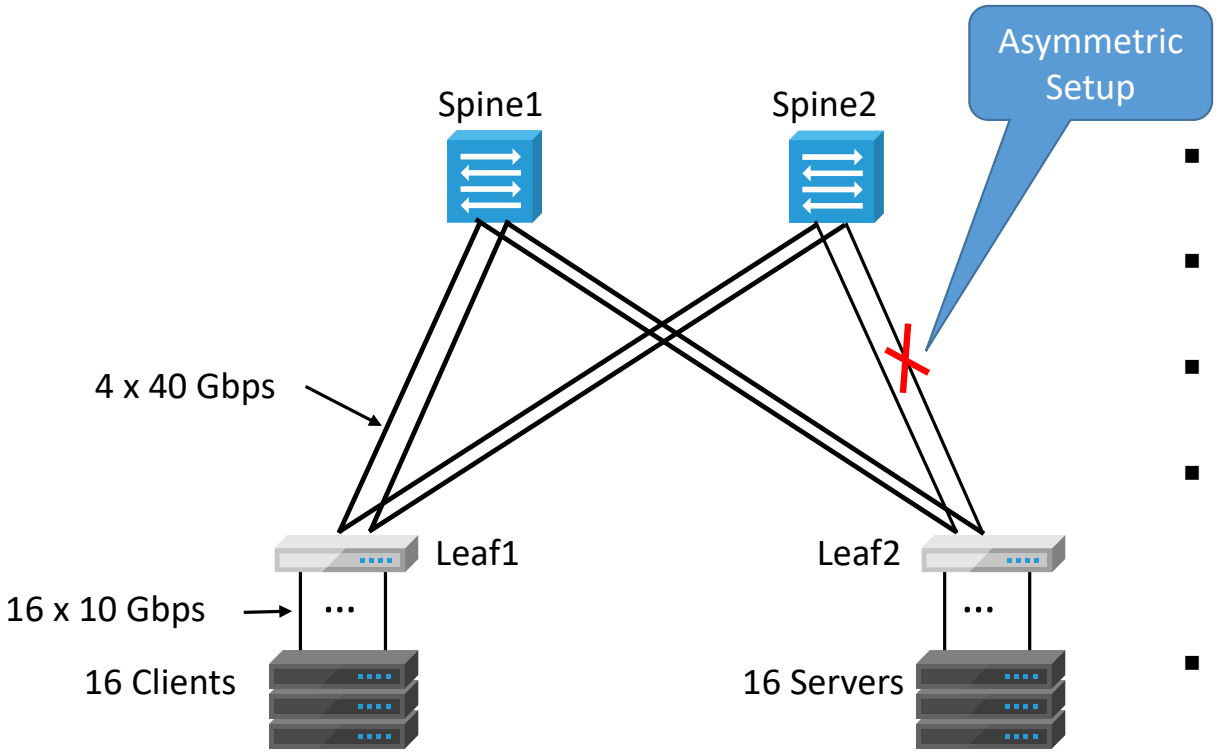
vSwitch Load balancing
Scheme 3: CLOVE-INT

- Utilization-aware balancing based on INT feedback



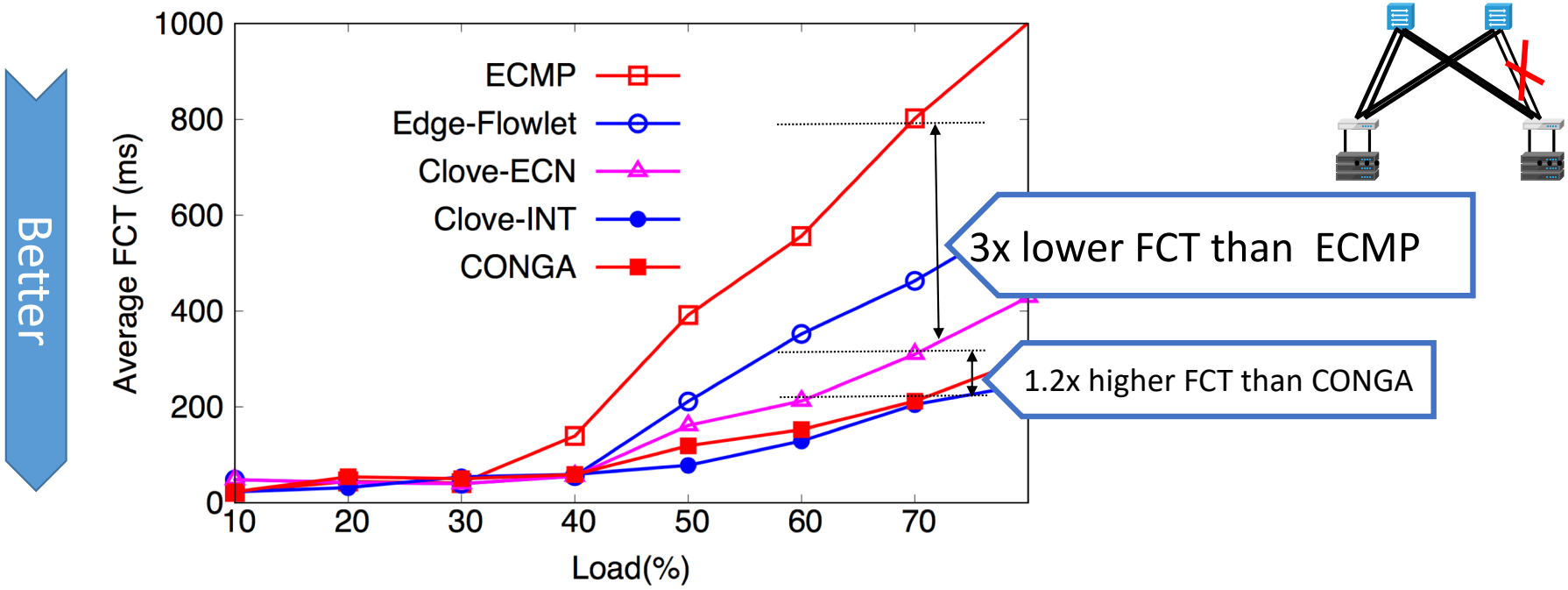
Dst	SPort	Util
H2	P1	40
H2	P2	30
H2	P3	50
H2	P4	10

Performance evaluation setup



- 2-tier leaf-spine symmetric topology
- Web Search Workload
- Client on Leaf1 <-> server on Leaf2
- Measure Average Flow Completion Time (FCT)
- Compare Edge-Flowlet and Clove-ECN to ECMP, MPTCP and Presto

NS2 Simulation with CONGA – Asymmetric



CLOVE-ECN captures 80% of the performance gain between ECMP and CONGA

CLOVE highlights

- CLOVE-ECN captures 80% of the performance gain of CONGA
- No changes to network hardware, VMs, applications
- Adapts to asymmetry within the data plane
- Scalable by virtue of distributed state