DRILL

Micro Load Balancing for Low-latency Data Center Networks

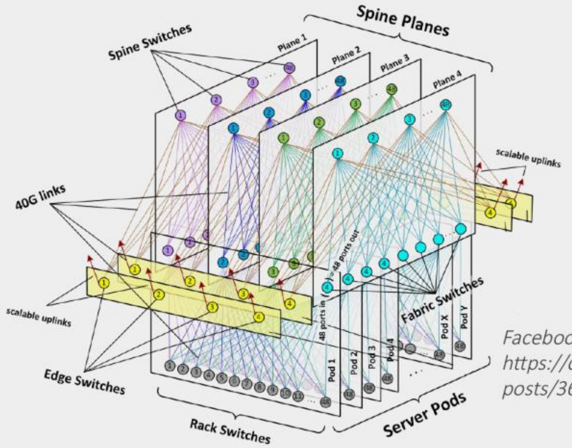
WISCONSIN
UNIVERSITY OF WISCONSIN–MADISON

ILLINOIS

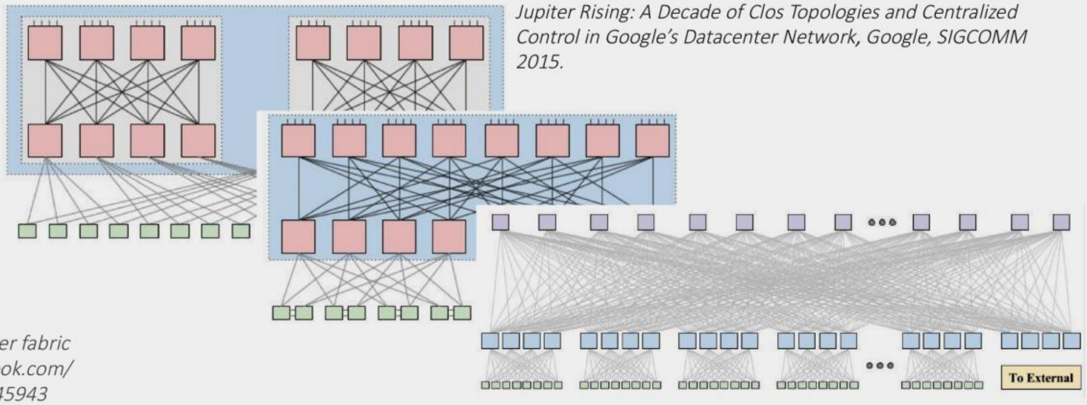
UNIVERSITY OF TORONTO

intel

Soudeh Ghorbani
Zibin Yang
Brighten Godfrey
Yashar Ganjali
Amin Firoozshahian

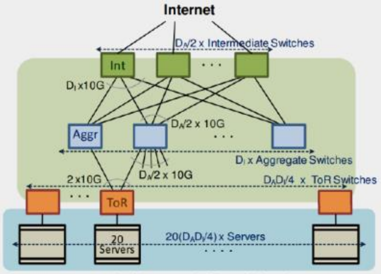


Facebook data center fabric https://code.facebook.com/posts/360346274145943

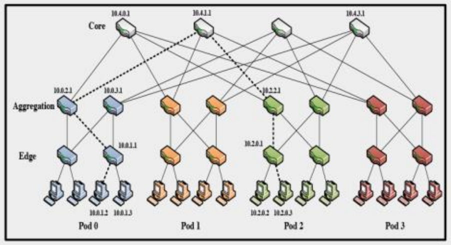


Jupiter Rising: A Decade of Clos Topologies and Centralized Control in Google's Datacenter Network, Google, SIGCOMM 2015.

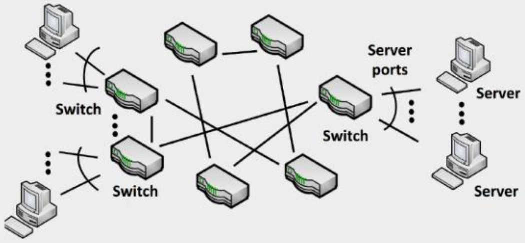
Data center topologies provide high capacity



VL2: a scalable and flexible data center network, C. Kim et al., SIGCOMM 2009



A scalable, commodity data center network architecture, M. Al-Fares et al., SIGCOMM 2008



Jellyfish: Networking Data Centers Randomly., A. Singla et al., NSDI 2012

BUT WE ARE STILL NOT USING THE CAPACITY EFFICIENTLY!

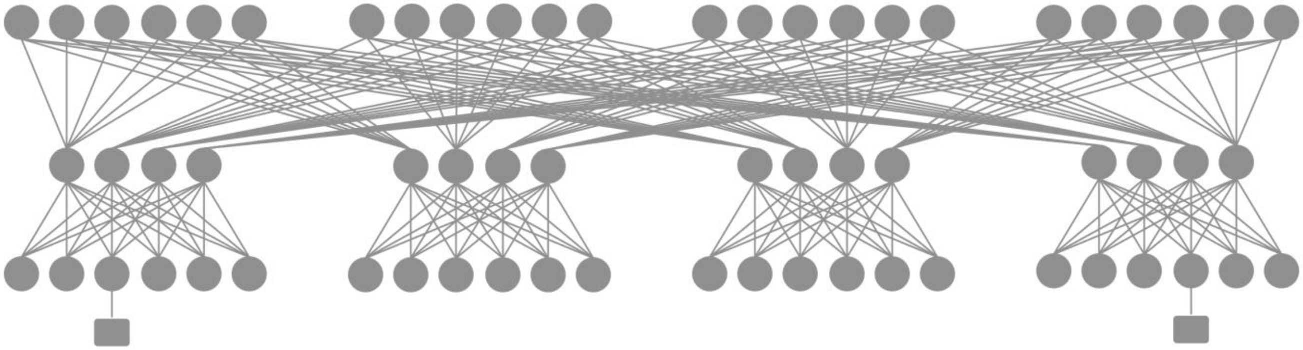
Networks experience high congestion drops as utilization approaches 25%[1].

Further improving fabric congestion response remains an ongoing effort[1].

[1] Jupiter Rising : A Decade of Clos Topologies and Centralized Control in Google's Datacenter Network, Google, SIGCOMM 2015.

THE GAP?

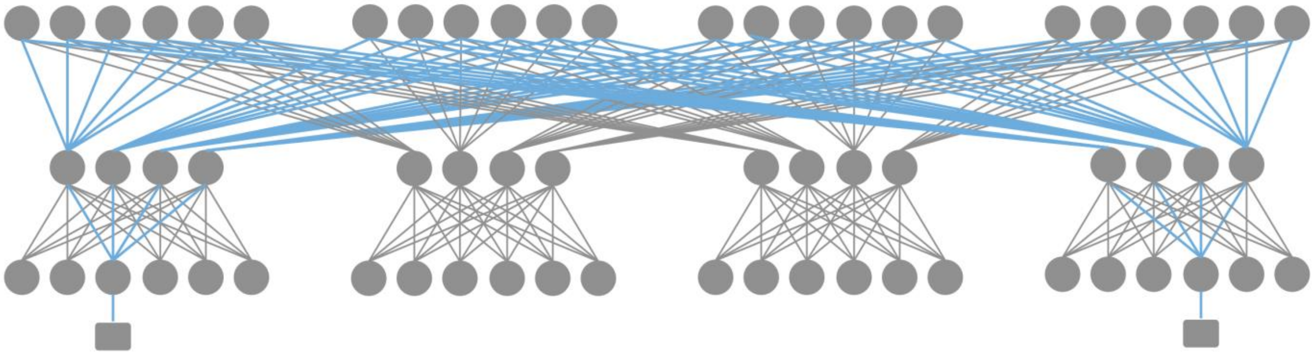
High bandwidth provided via massive Multipathing.



Facebook data center fabric : https://code.facebook.com/posts/360346274145943

THE GAP?

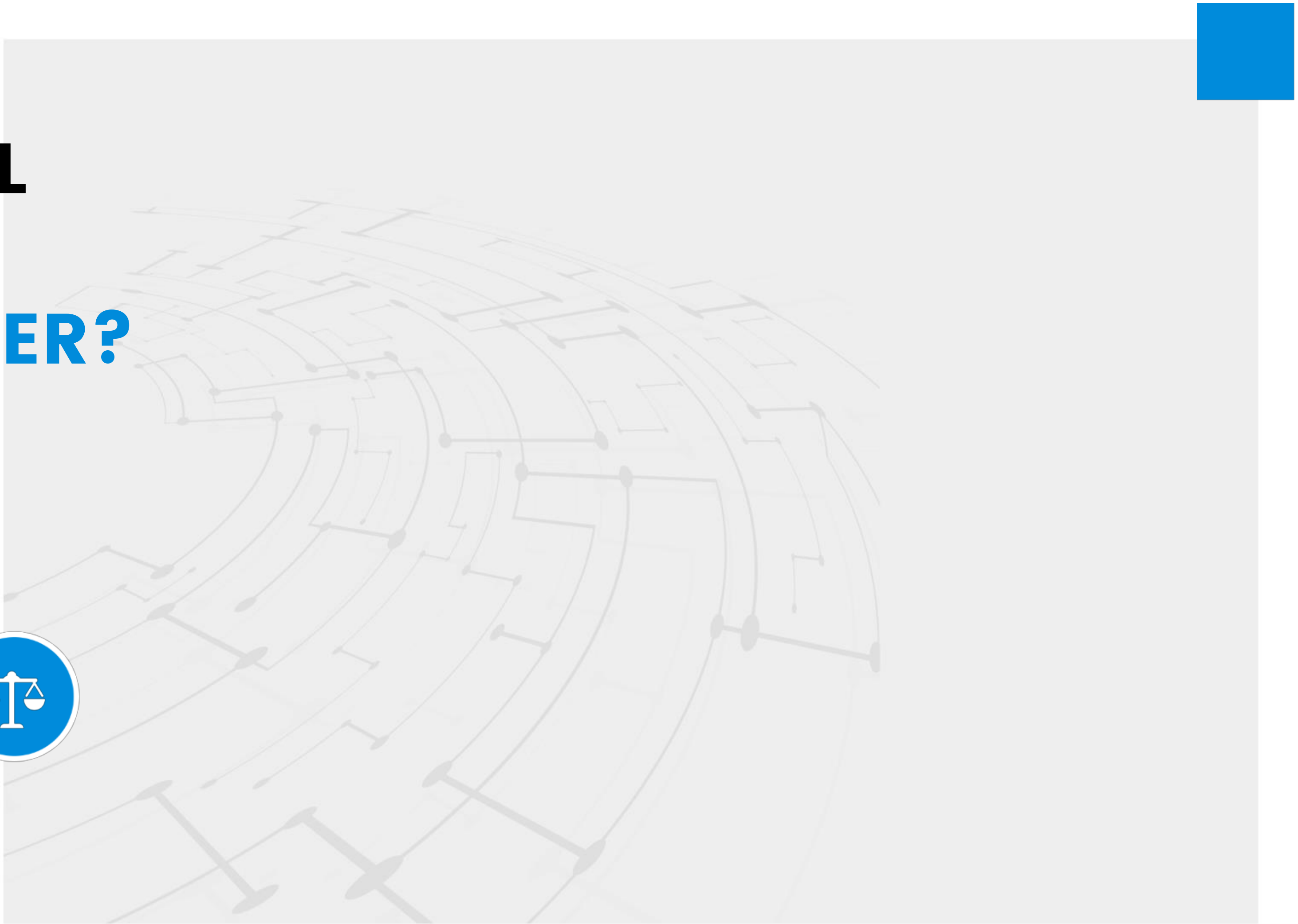
High bandwidth provided via massive Multipathing.



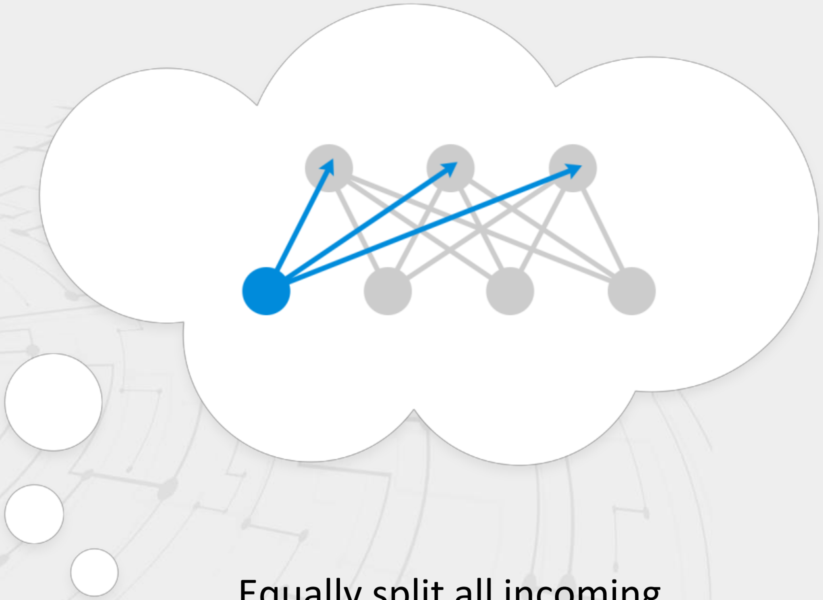
Congestion happens even when there is spare capacity to mitigate it elsewhere[2].

[2] Network Traffic Characteristics of Data Centers in the Wild, T. Benson et al., IMC 2010.

OPTIMAL LOAD BALANCER?

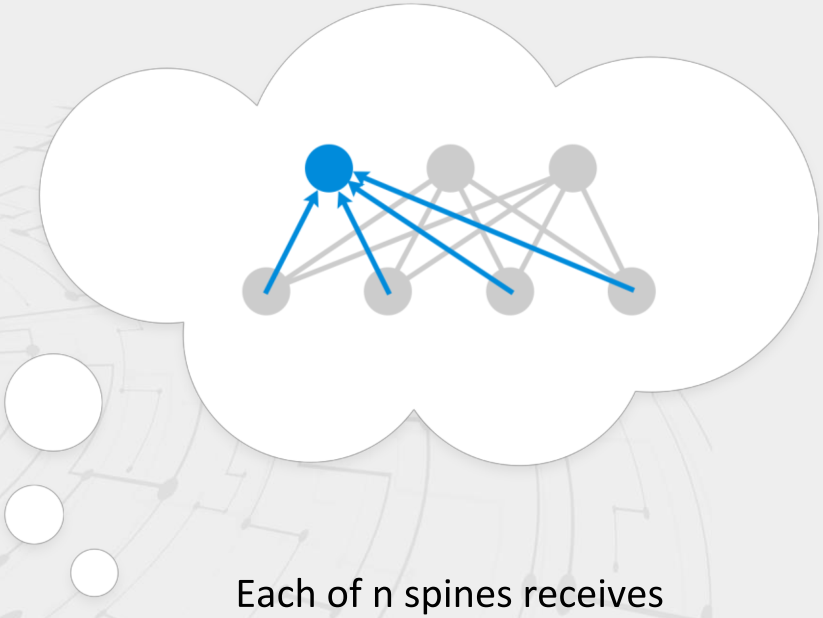


A STARTING POINT : “EQUAL SPLIT FLUID” (ESF)



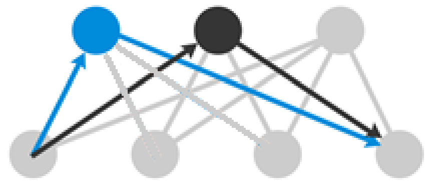
Equally split all incoming flow to other leaves along all shortest outgoing paths.

A STARTING POINT : “EQUAL SPLIT FLUID” (ESF)



Each of n spines receives 1/n of all inter-leaf traffic.

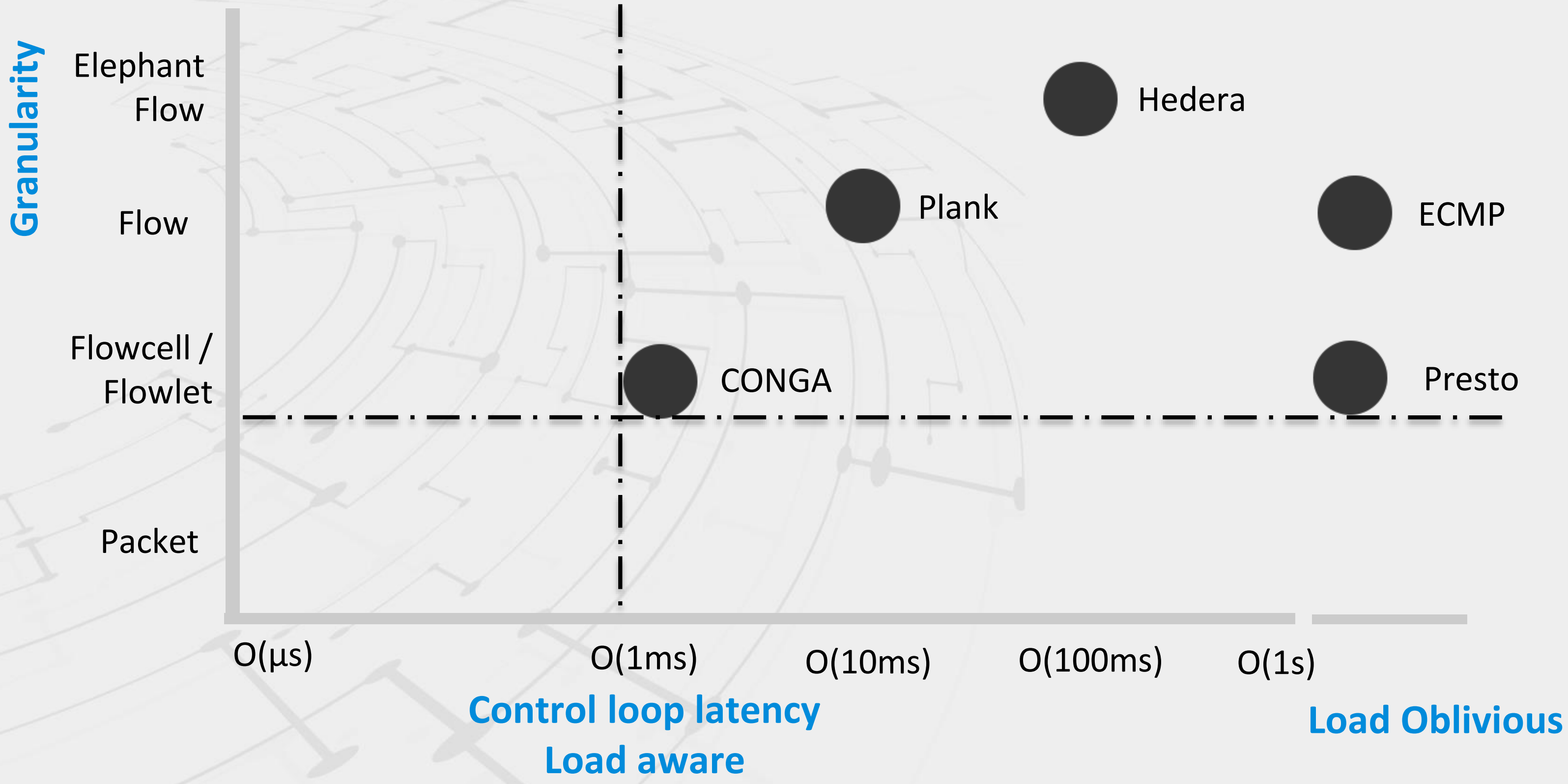
A STARTING POINT : "EQUAL SPLIT FLUID" (ESF)



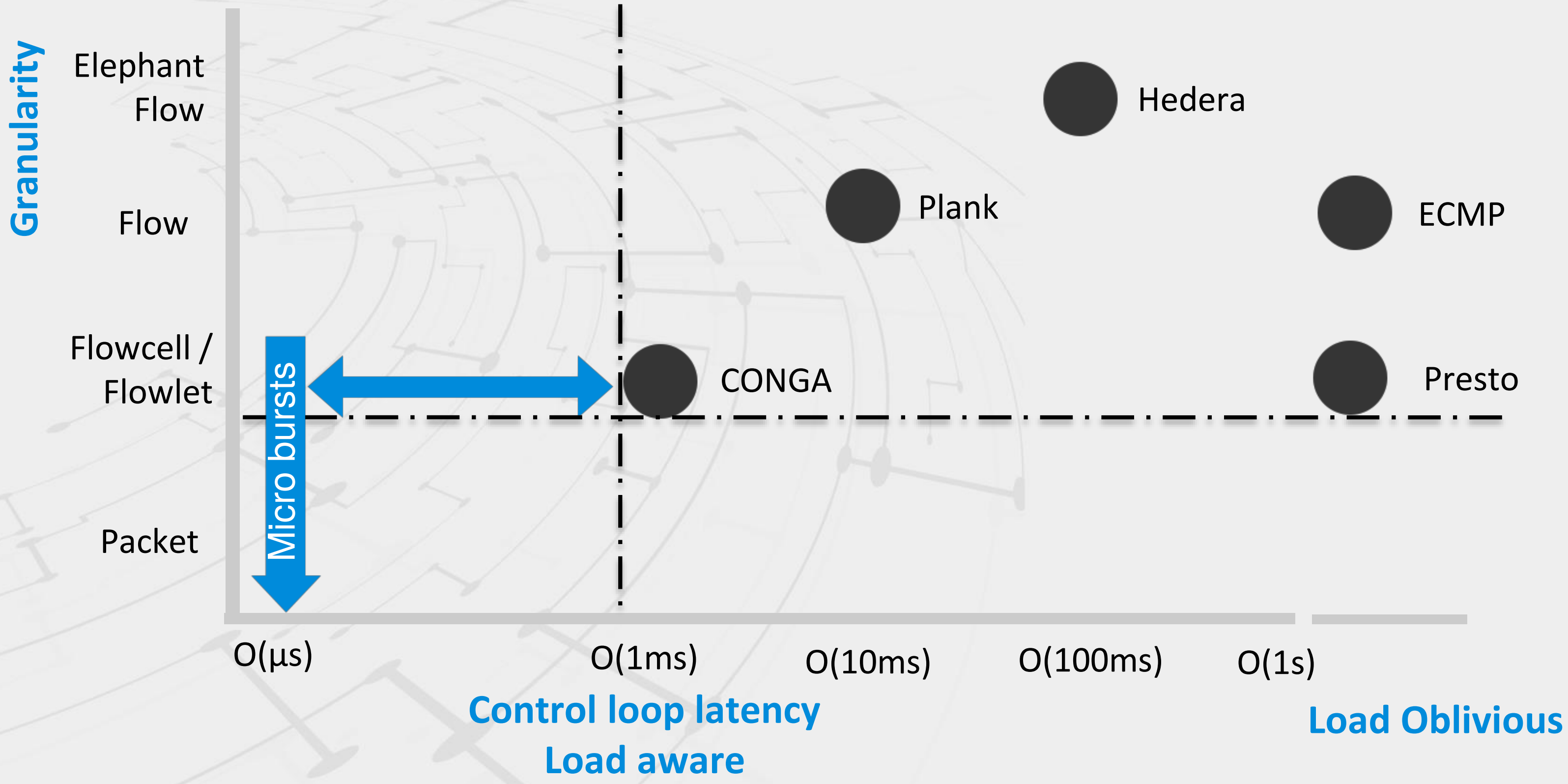
Therefore, any two paths between the same source and destination experience the same utilization (and mix of traffic) at all corresponding hops.

ESF is optimal for all traffic demands.

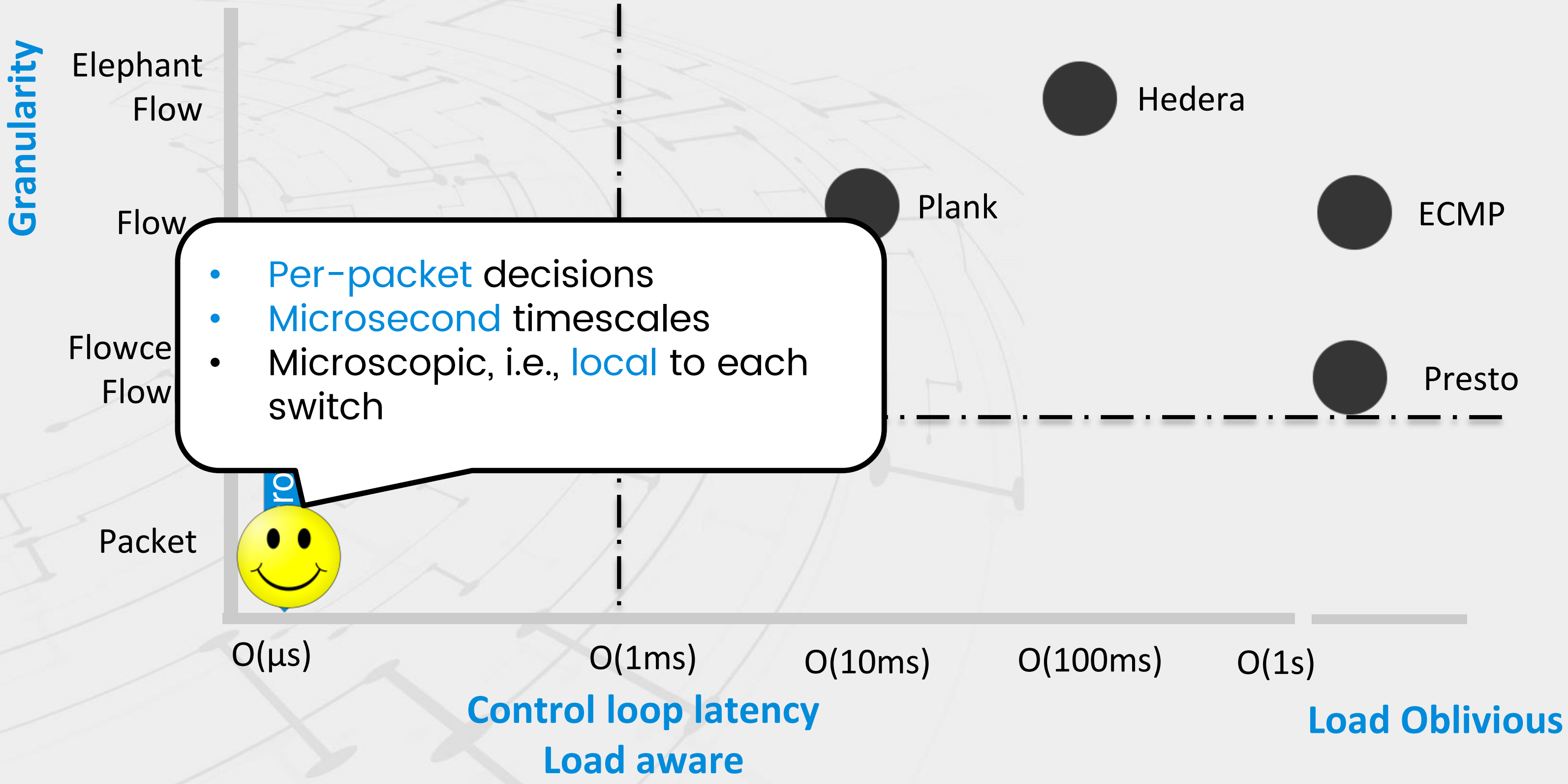
APPROXIMATING ESF



APPROXIMATING ESF

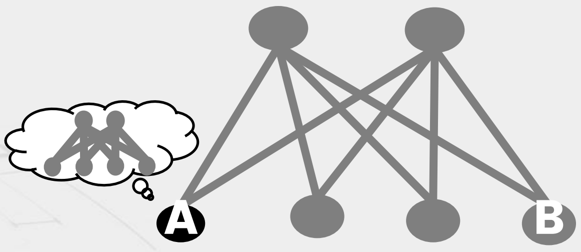


MICRO LOAD BALANCING



- Per-packet decisions
- Microsecond timescales
- Microscopic, i.e., local to each switch

SIMPLIFIED MICRO LOAD BALANCING



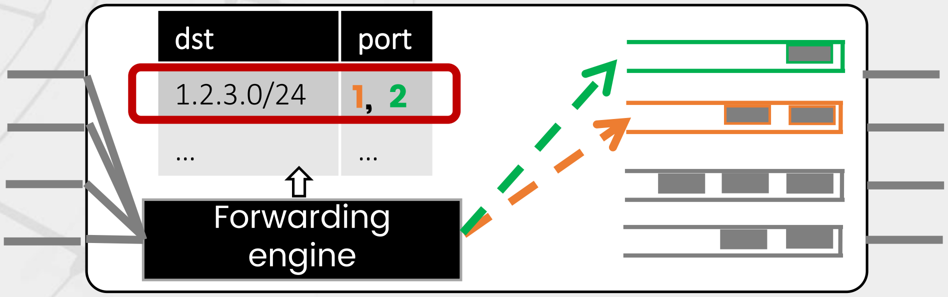
~ECMP

Switch control plane :

(1) Discover topology.
(2) Compute multiple shortest paths.
(3) Install into FIB.

Switch data plane:

(1) Look up the candidate ports.
(2) Check the queue occupancy of the candidate ports.
(3) Enqueue the packet Dst: 1.2.3.4 east loaded candidate port.



dst	port
1.2.3.0/24	1, 2
…	…

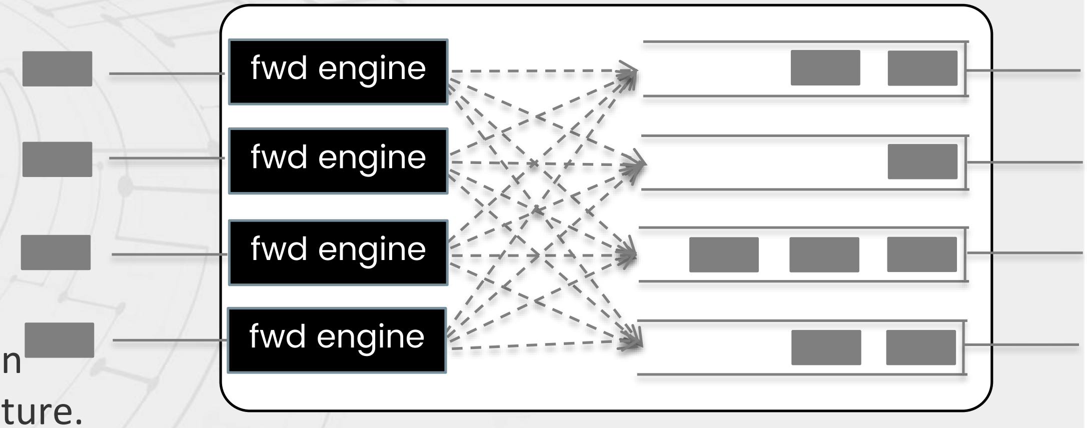
CHALLENGES

- Efficient micro load balancing implementation inside a switch
- Reordering caused by per-packet decisions
- Poor decisions in asymmetric topologies

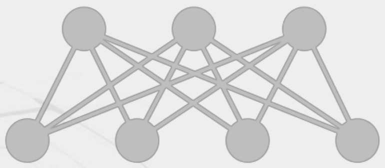
MICRO LOAD BALANCING
INSIDE A SWITCH

Checking all queues at line rate for every packet:

- is hard, especially for high radix switches.
- can cause synchronization effect in switches with distributed architecture.



MICRO LOAD BALANCING
WITH DRILL

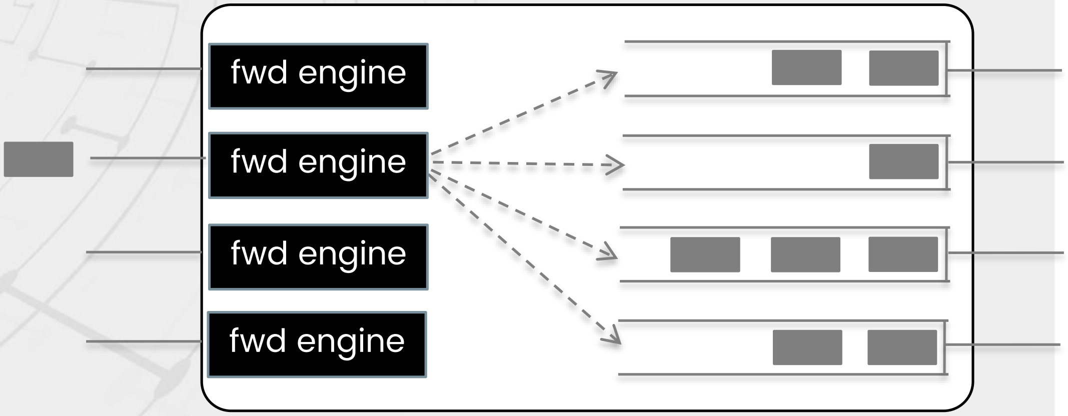


Switch control plane :

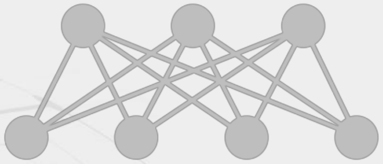
- Discover topology.
- Compute multiple shortest paths.
- Install into FIB.

Switch data plane

- Look up the candidate ports.
- Sample queue lengths of 2 random queues plus the least loaded queue from the previous packet.
- Send the packet to the least loaded one among those three.



MICRO LOAD BALANCING WITH DRILL



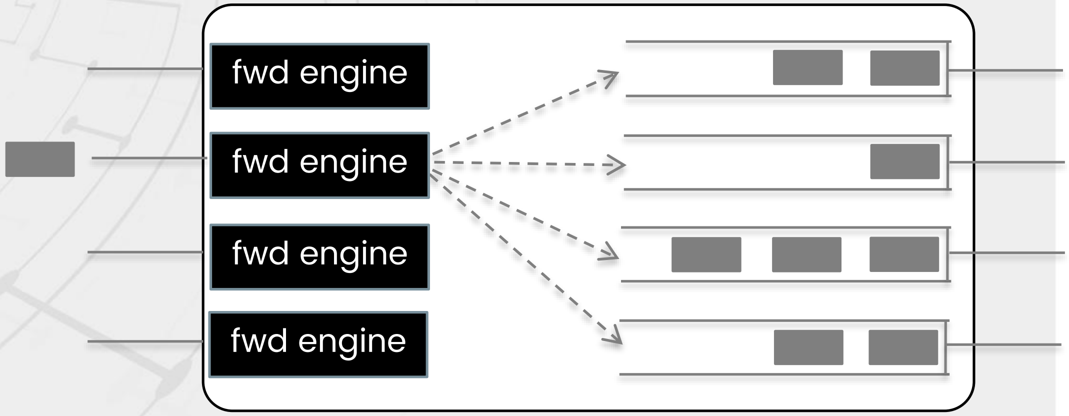
Switch control plane :

- Discover topolo
- Compute multip
- Install into FIB.

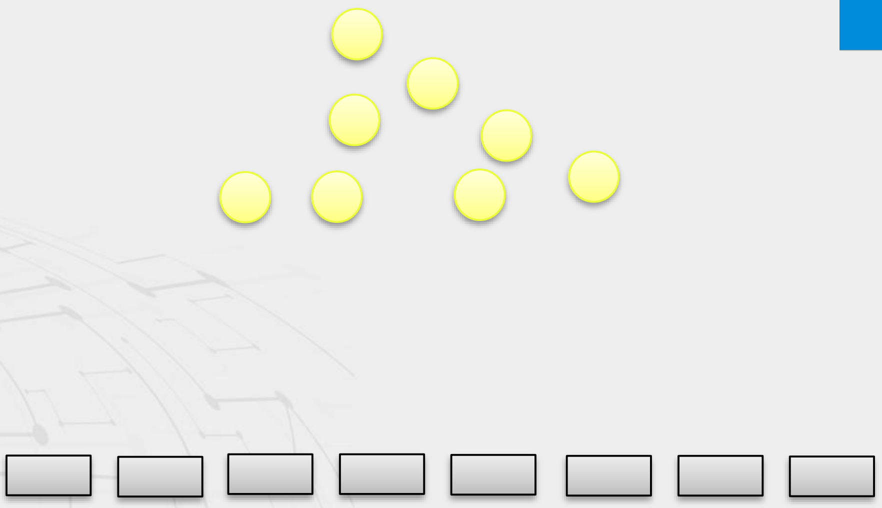
Verilog switch implementation shows that DRILL imposes less than 1% switch area overhead.

Switch data plane

- Look up the candidate ports.
- Sample queue lengths of 2 random queues plus the least loaded queue from the previous packet.
- Send the packet to the least loaded one among those three.



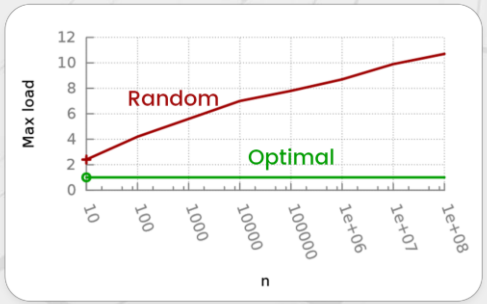
THE POWER OF TWO CHOICES



- n bins and n balls
- Each ball choosing a bin independently and uniformly at random

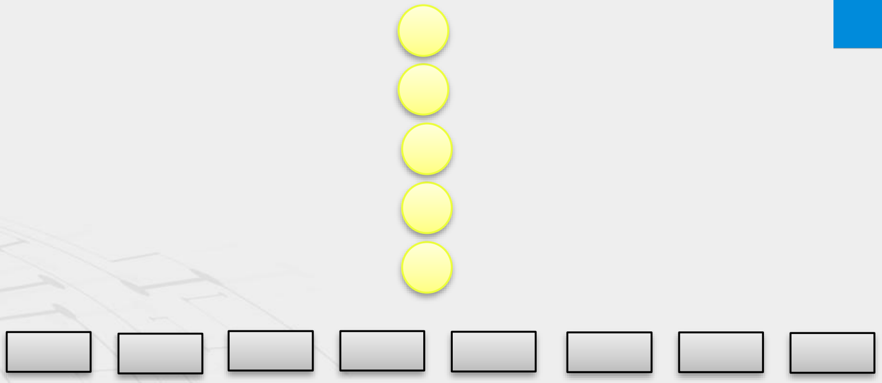
- **Max load:**

$$\theta(\frac{logn}{loglogn})$$



Y. Azar, A. Z. Broder, A. R. Karlin, and E. Upfal. Balanced allocations. SIAM Journal on Computing, 1994

THE POWER OF TWO CHOICES



- n bins and n balls
- Balls placed sequentially,
- in the least loaded of d≥2 random bins

- Max load:

$$\theta(\frac{loglogn}{logd})$$

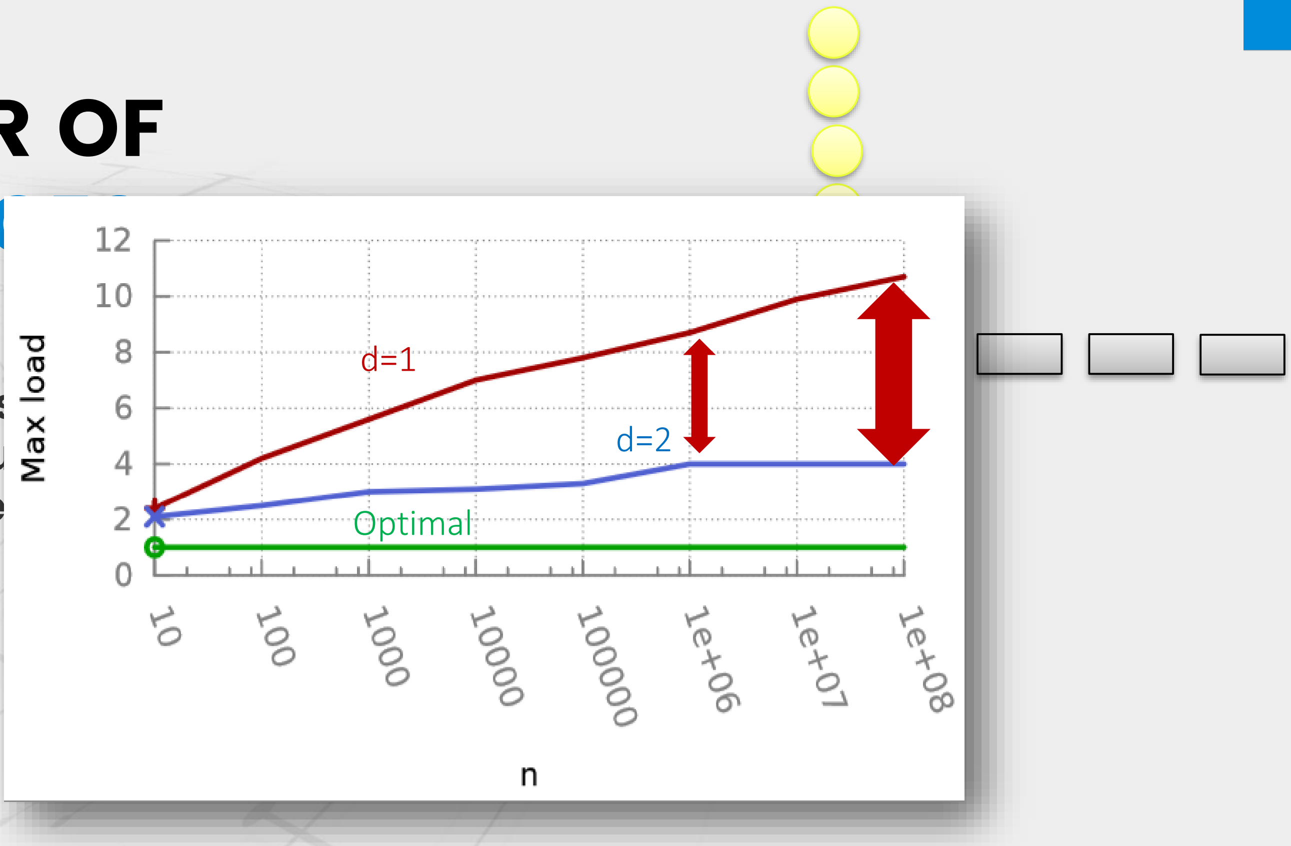
Y. Azar, A. Z. Broder, A. R. Karlin, and E. Upfal. Balanced allocations. SIAM Journal on Computing, 1994

THE POWER OF TWO CHOIC

- n bins and n balls
- Balls placed sequ
- in the least loade random bins

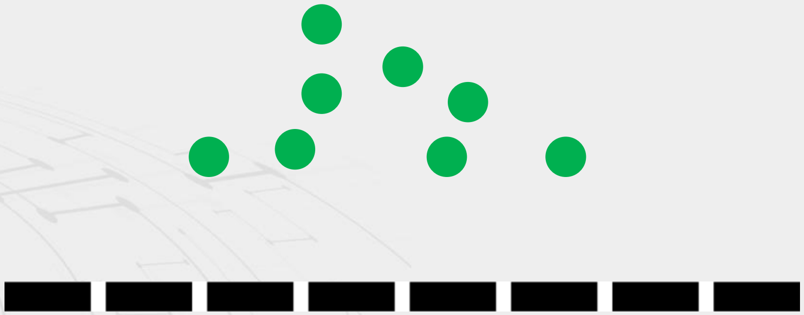
- **Max load:**

$$\theta(\frac{loglogn}{logd})$$



Y. Azar, A. Z. Broder, A. R. Karlin, and E. Upfal. Balanced allocations. SIAM Journal on Computing, 1994

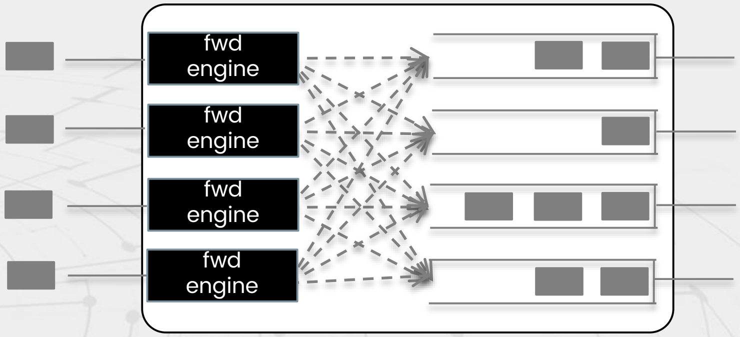
WHAT WE WANT



Switch has queues, instead of bins, from which packets leave.

Each forwarding engine chooses a queue independently, no coordination.

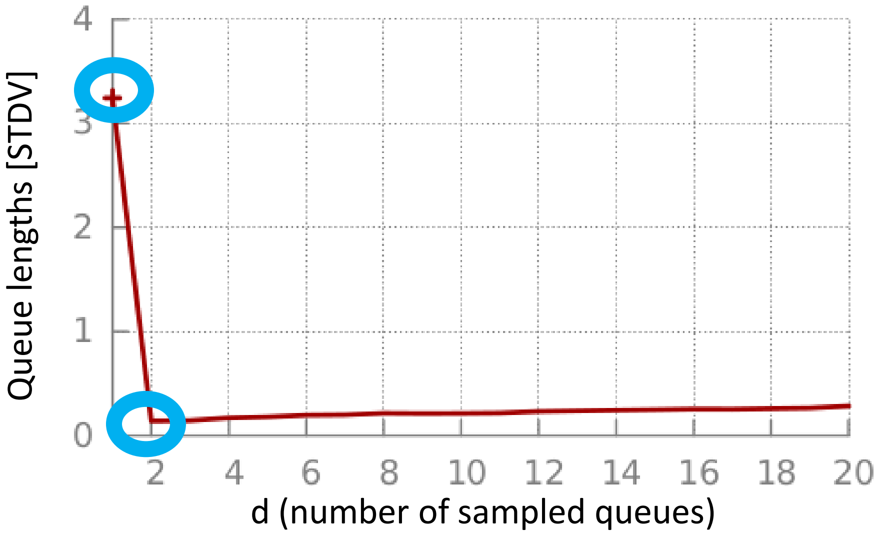
WHAT WE WANT



Switch has queues, instead of bins, from which packets leave.

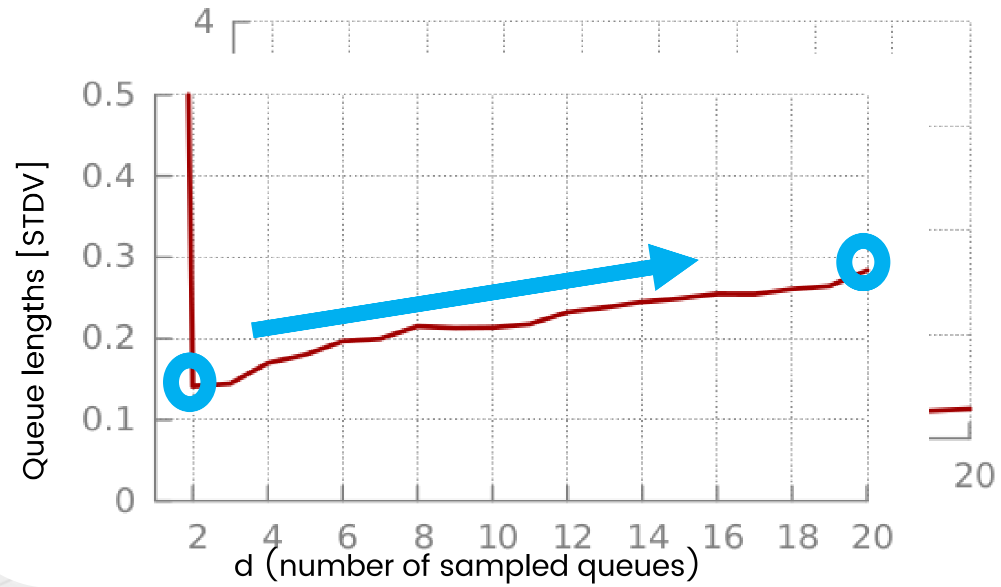
Each forwarding engine chooses a queue independently, no coordination.

THE PITFALLS OF CHOICE



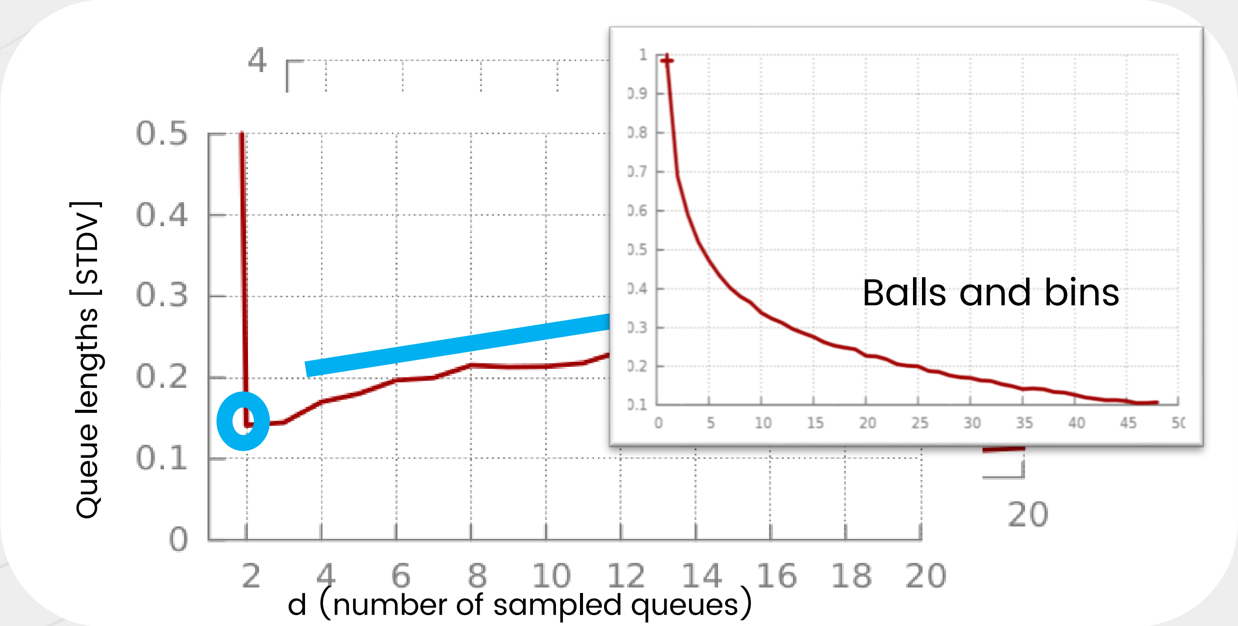
Topology: Clos topology with 8 spines, 10 leafs, each connected to 48 hosts. Links: 10Gbps.
Switches have 6 forwarding engines. Network is under 75% load.
Trace: Inside the social network's (datacenter) network, Facebook, SIGCOMM 2015.

THE PITFALLS OF CHOICE



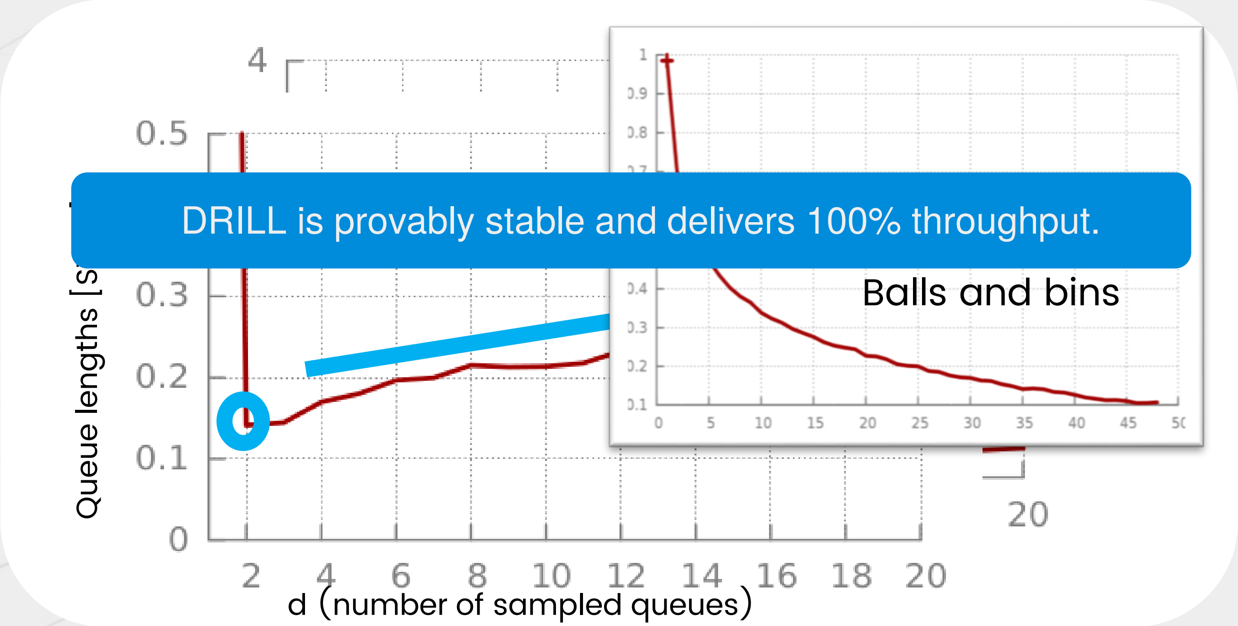
Topology: Clos topology with 8 spines, 10 leafs, each connected to 48 hosts. Links: 10Gbps. Switches have 6 forwarding engines. Network is under 75% load. Trace: Inside the social network's (datacenter) network, Facebook, SIGCOMM 2015.

THE PITFALLS OF CHOICE



Topology: Clos topology with 8 spines, 10 leafs, each connected to 48 hosts. Links: 10Gbps.
Switches have 6 forwarding engines. Network is under 75% load.
Trace: Inside the social network's (datacenter) network, Facebook, SIGCOMM 2015.

THE PITFALLS OF CHOICE



DRILL is provably stable and delivers 100% throughput.

Topology: Clos topology with 8 spines, 10 leafs, each connected to 48 hosts. Links: 10Gbps.
Switches have 6 forwarding engines. Network is under 75% load.
Trace: Inside the social network's (datacenter) network, Facebook, SIGCOMM 2015.

CHALLENGES

- Efficient micro load balancing implementation inside a switch
- Reordering caused by per-packet decisions
- Poor decisions in asymmetric topologies

THOU SHALT NOT
SPLIT FLOWS!

Splitting flows

Reordering

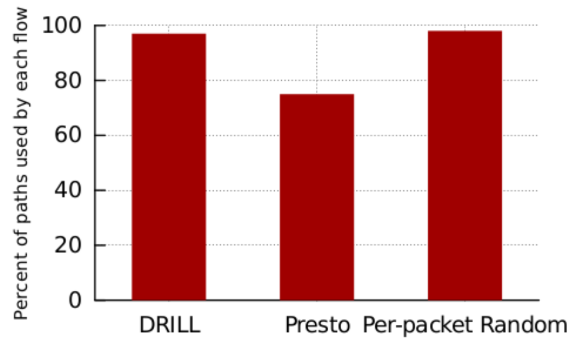
Throughput degradation

Splitting flows

Reordering

Throughput degradation

DRILL SPLITS FLOWS ALONG MANY PATHS

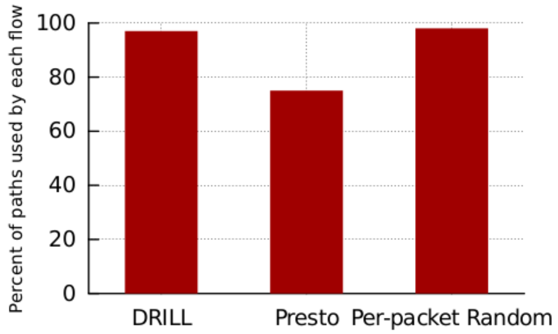


Splitting flows

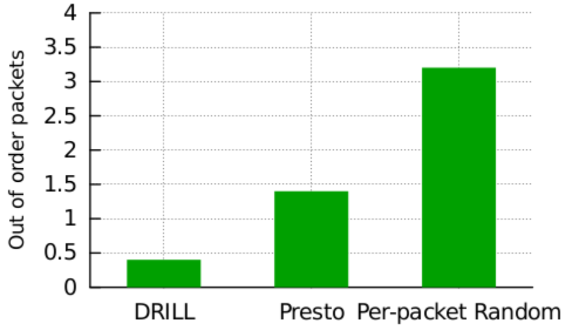
Reordering

Throughput degradation

DRILL SPLITS FLOWS ALONG MANY PATHS



YET CAUSES LITTLE REORDERING!

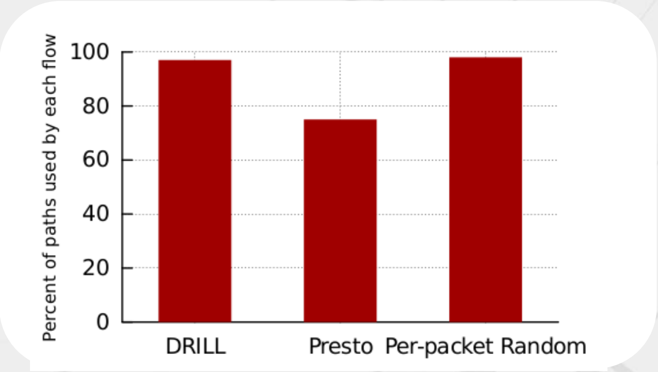


Splitting flows

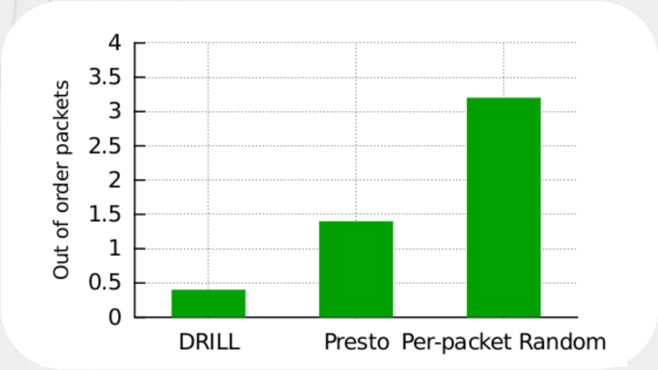
Reordering

Throughput degradation

DRILL SPLITS FLOWS ALONG MANY PATHS



YET CAUSES LITTLE REORDERING!



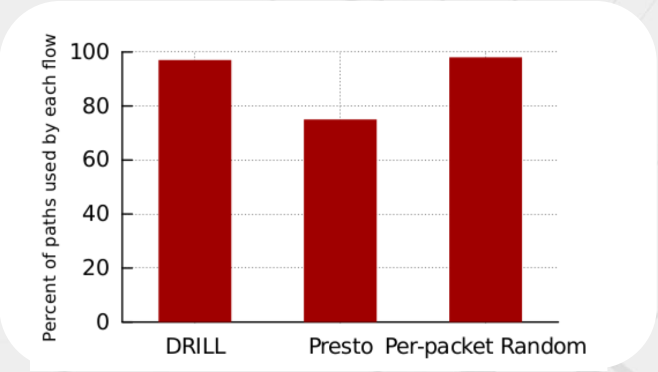
Splitting flows

Latency variance

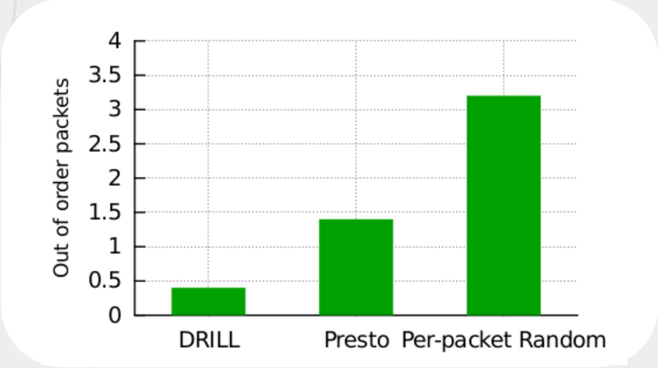
Reordering

Throughput degradation

DRILL SPLITS FLOWS ALONG MANY PATHS



YET CAUSES LITTLE REORDERING!



Splitting flows

Latency variance

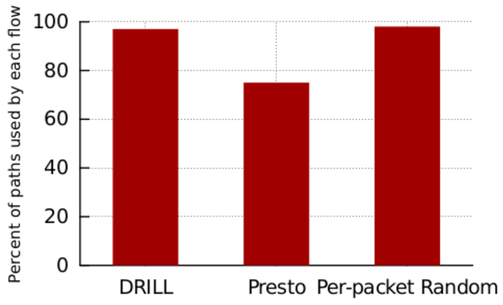
Reordering

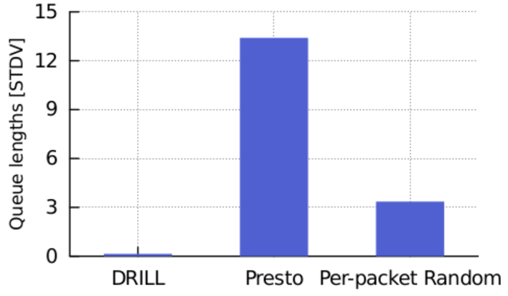
Throughput degradation

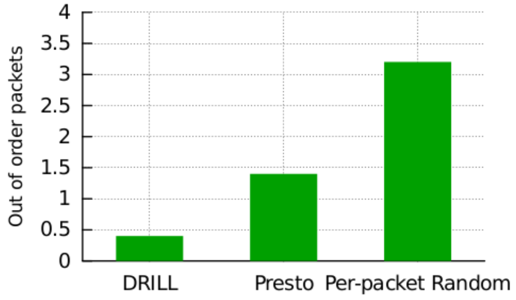
DRILL SPLITS FLOWS ALONG MANY PATHS

BUT HAS LOW QUEUEING DELAY VARIANCE

AND THEREFORE CAUSES LITTLE REORDERING!







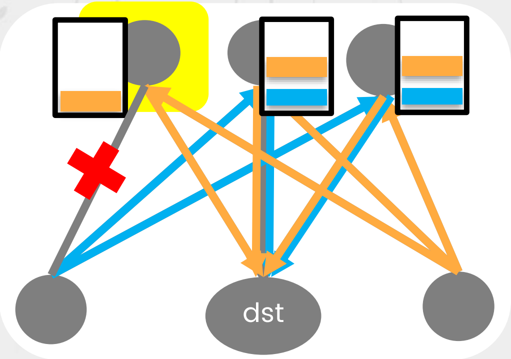
Insight: Queueing delay variance is so small that it doesn't matter what path the packet takes.

- Efficient micro load balancing implementation inside a switch
- Reordering caused by per-packet decisions
- Poor decisions in asymmetric topologies

ASYMMETRY
WHAT CAN GO WRONG?

KEY PROBLEMS

1. Different queueing results in **reordering**.
2. TCP flows split along multiple paths will respond to congestion on the worst path, causing **bandwidth inefficiency.**



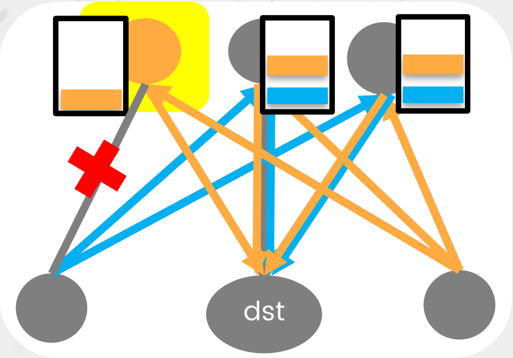
ASYMMETRY
WHAT CAN GO WRONG?

ROOT CAUSE:

Flow splitting across asymmetric paths.

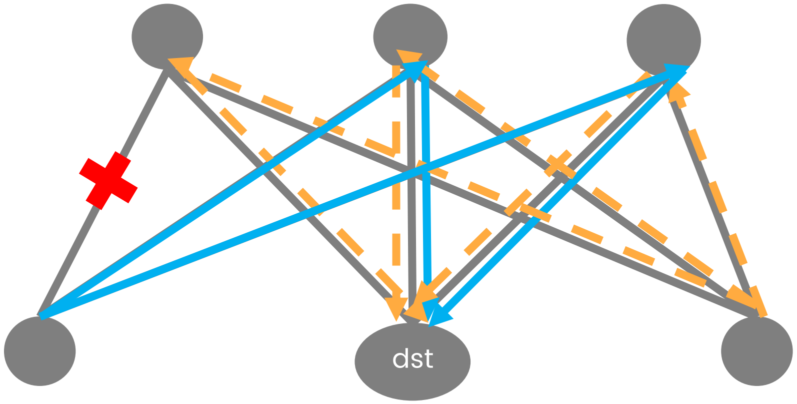
KEY IDEA:

Decompose the network into symmetric components and micro-LB inside components.

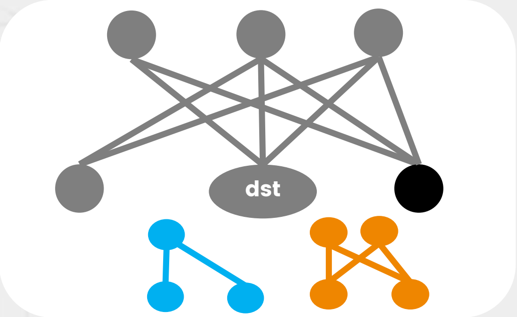


ASYMMETRY:
GRAPH DECOMPOSITION

Intuition: Two paths are symmetric if they have equal number of hops and the i^{th} queue carries the same flows on all paths for all *i*.



MICRO LOAD BALANCING IN ASYMMETRIC TOPOLOGIES

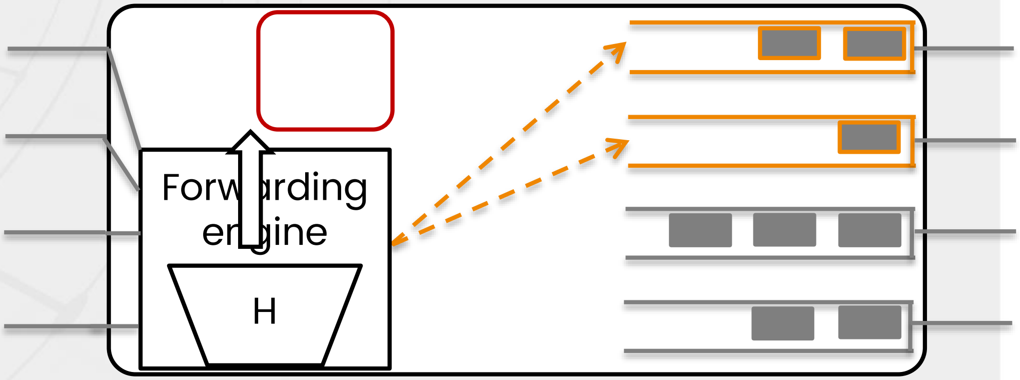


SWITCH CONTROL PLANE :

- Discover topology.
- Compute multiple shortest paths.
- Decompose the network into symmetric components.

Switch data plane:

- Hash flows to component.
- Micro load balance inside a component.



dst

EXPERIMENTAL EVALUATION

Environment

- OMNET++ simulator
- Ported Linux 2.6 TCP implementation

Topology

- 2- and 3-stage Clos
- Asymmetric topologies
- Varying failures

Workload

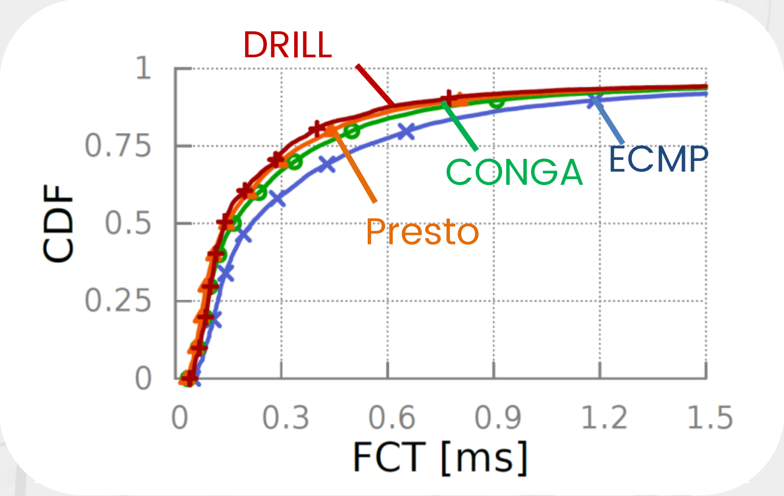
- Real measurements *[1]*
- Synthetic
- Incast patterns

[1] Inside the social network's (datacenter) network, Facebook, SIGCOMM 2015.

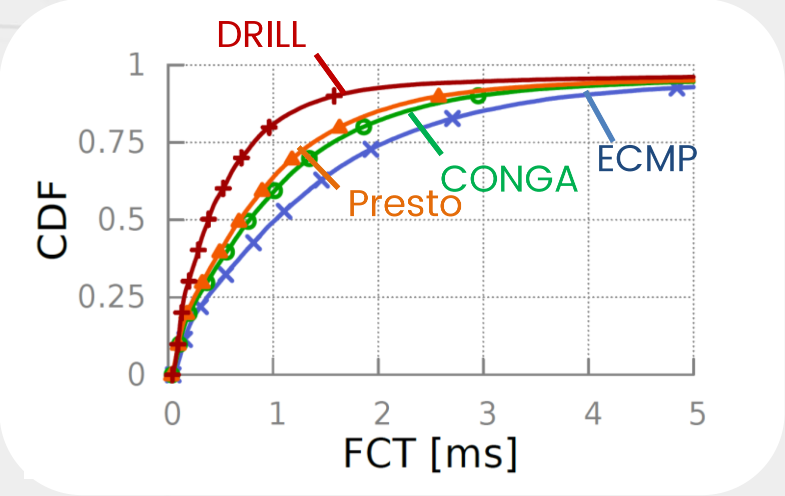
DRILL REDUCES FCT
ESPECIALLY UNDER LOAD

Clos with 16 spines, 16 leafs, each connected to 20 hosts, links: 10Gbps

30% load

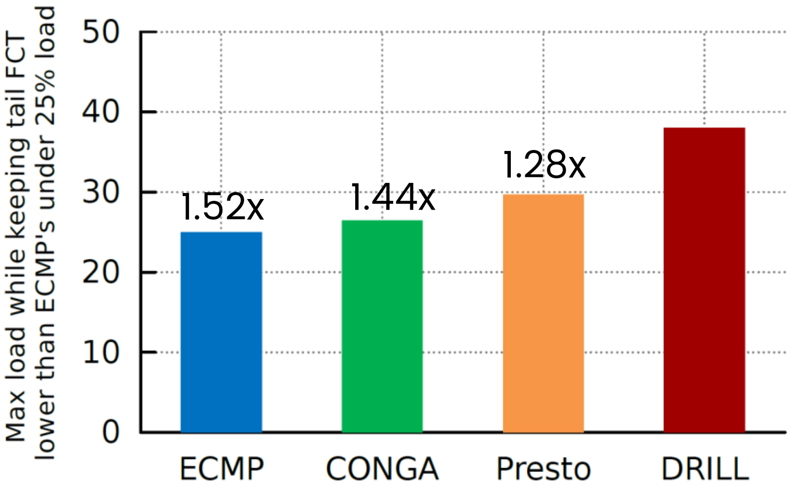


80% load



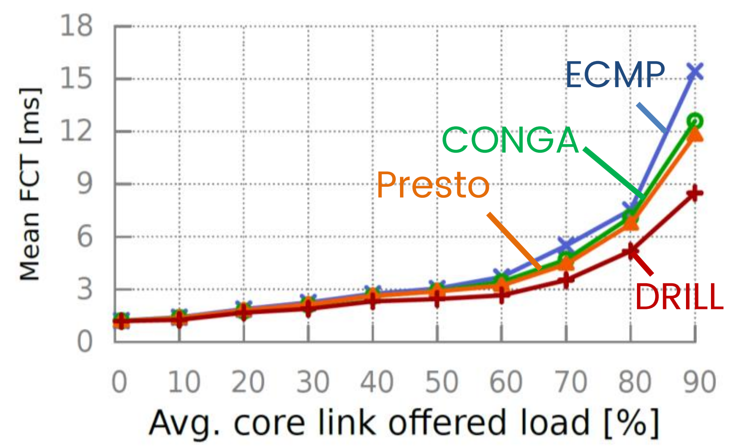
DRILL ALLOWS HIGHER UTILIZATION WITH EQUAL LATENCY

Clos with 16 spines, 16 leafs, each connected to 20 hosts, links: 10Gbps



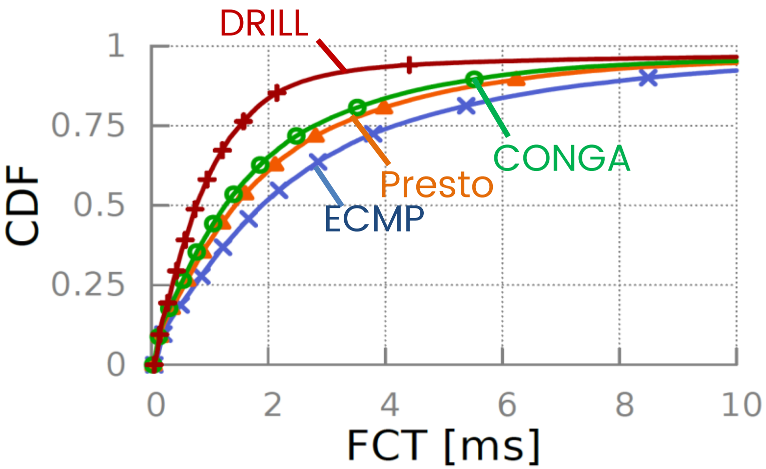
DRILL HANDLES ASYMMETRY

Clos with 4 spines and 16 leafs, each connected to 20 hosts, edge links: 10Gbps, core links: 40Gbps. 5 randomly selected leaf-spine links fail.

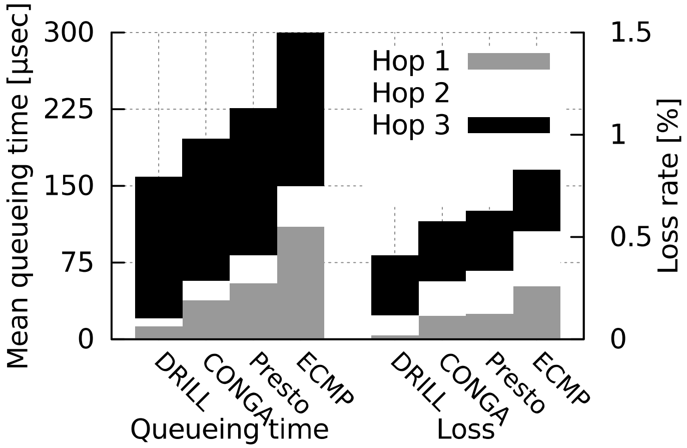


DRILL CUTS THE TAIL LATENCY IN INCAST

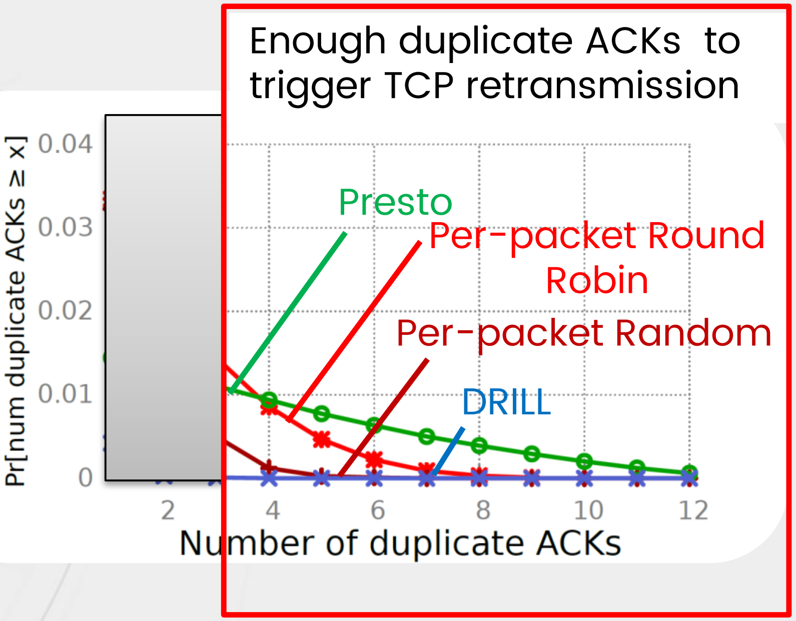
Clos with 4 spines and 16 leafs, each connected to 20 hosts, edge links: 10Gbps, core links: 40Gbps. Network is under 20% load. 10% of hosts send simultaneous requests for 10KB flows to 10% of the other hosts (all randomly selected).



UNDER THE HOOD



Leaf → Spine: big improvement
Spine → Leaf: some improvement
Leaf → Host: no improvement



Enough duplicate ACKs to trigger TCP retransmission

Fine granularity is helpful but not enough: load-awareness keeps dup ACKs under control.

DRILL’S KEY IDEAS

- Graph decomposition
- Micro-LB:
  Randomized algorithm

Key results:

- Theoretical analysis: Stable
  and 100% throughput
- Low FCT
  2.5x tail FCT reduction in incast
- Simple switch design
  - Less than 1% area overhead
  - No host changes
  - No global coordination