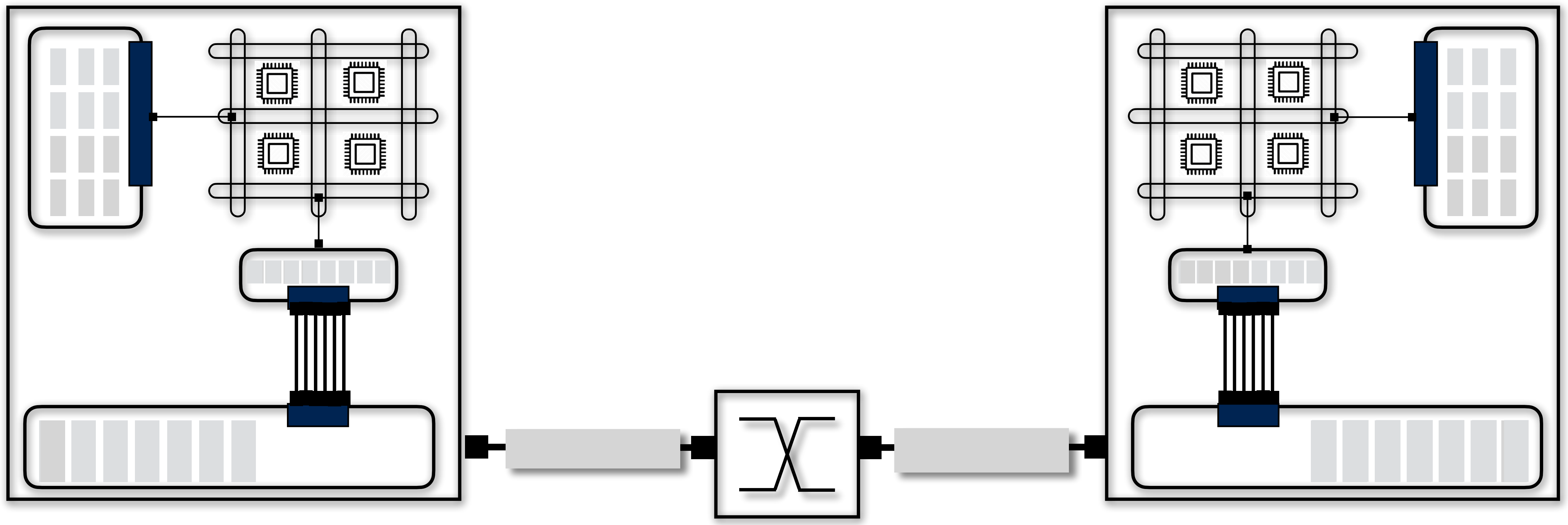
Host Congestion Control

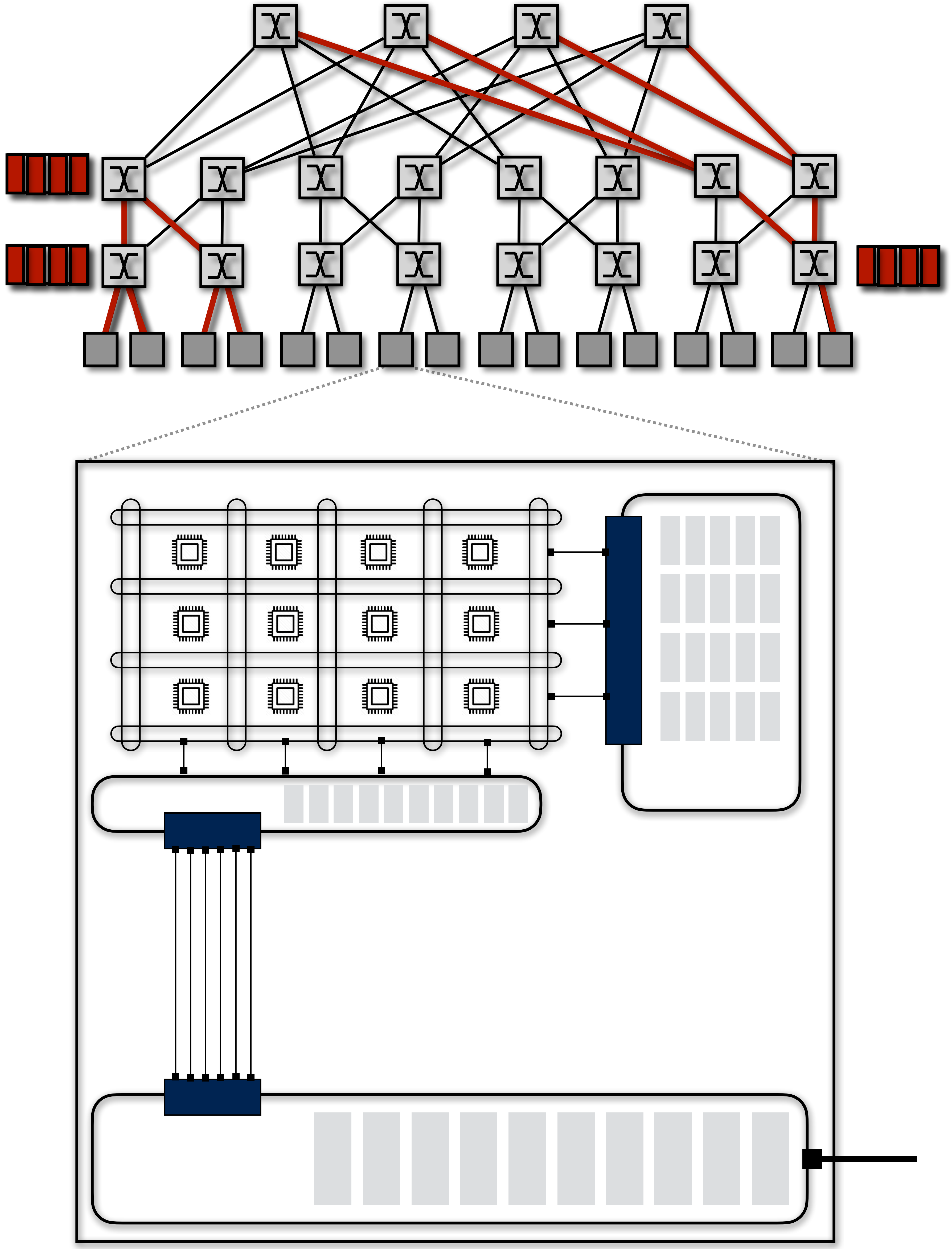


Saksham Agarwal
Cornell University

Arvind Krishnamurthy
Google &
University of Washington

Rachit Agarwal
Cornell University

Emergence of Host Congestion



Conventional wisdom: congestion happens in the network core

At switches

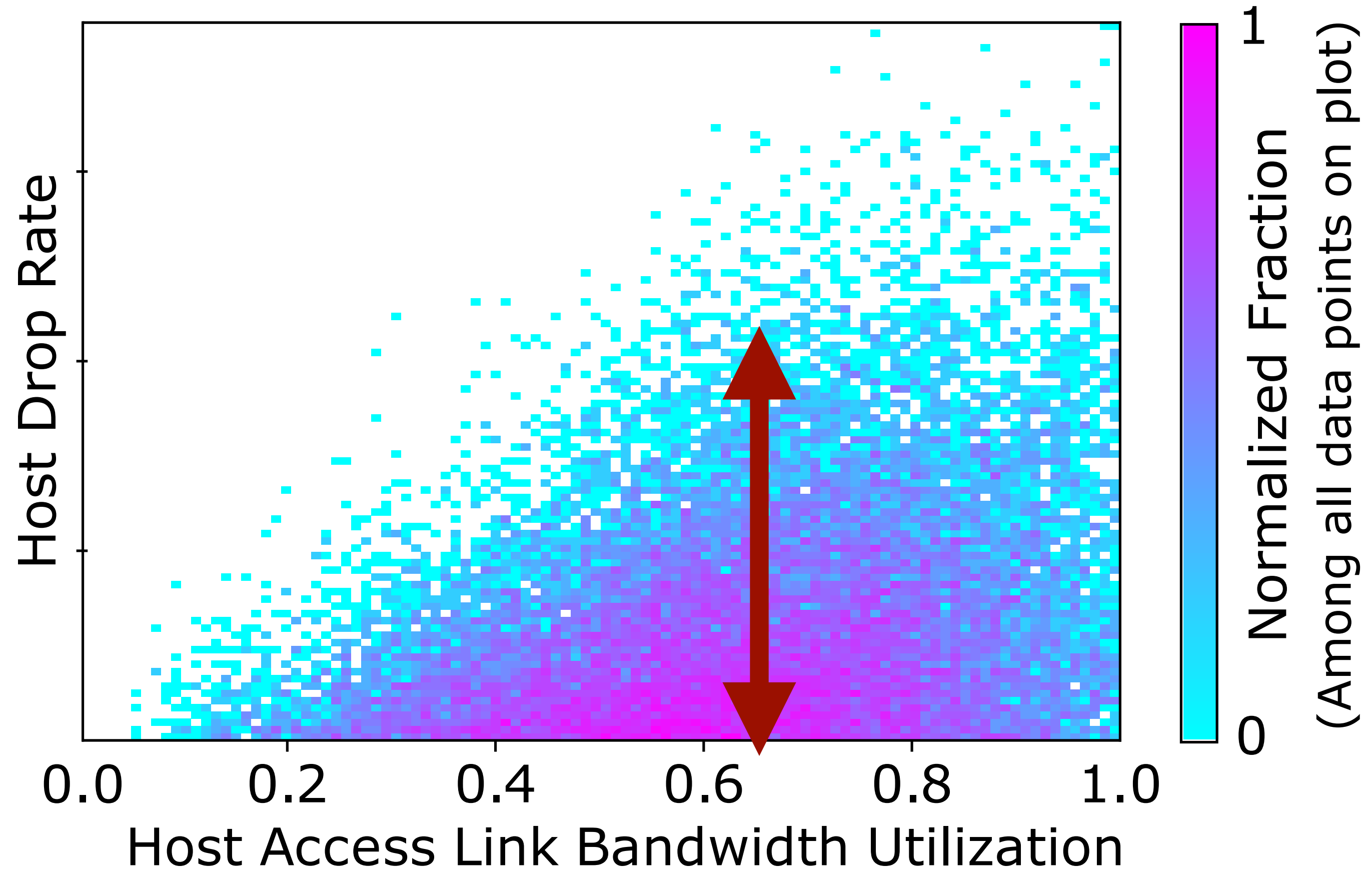
Recent technology trends: host congestion

E.g., recent studies from Google, Microsoft, Alibaba, etc.

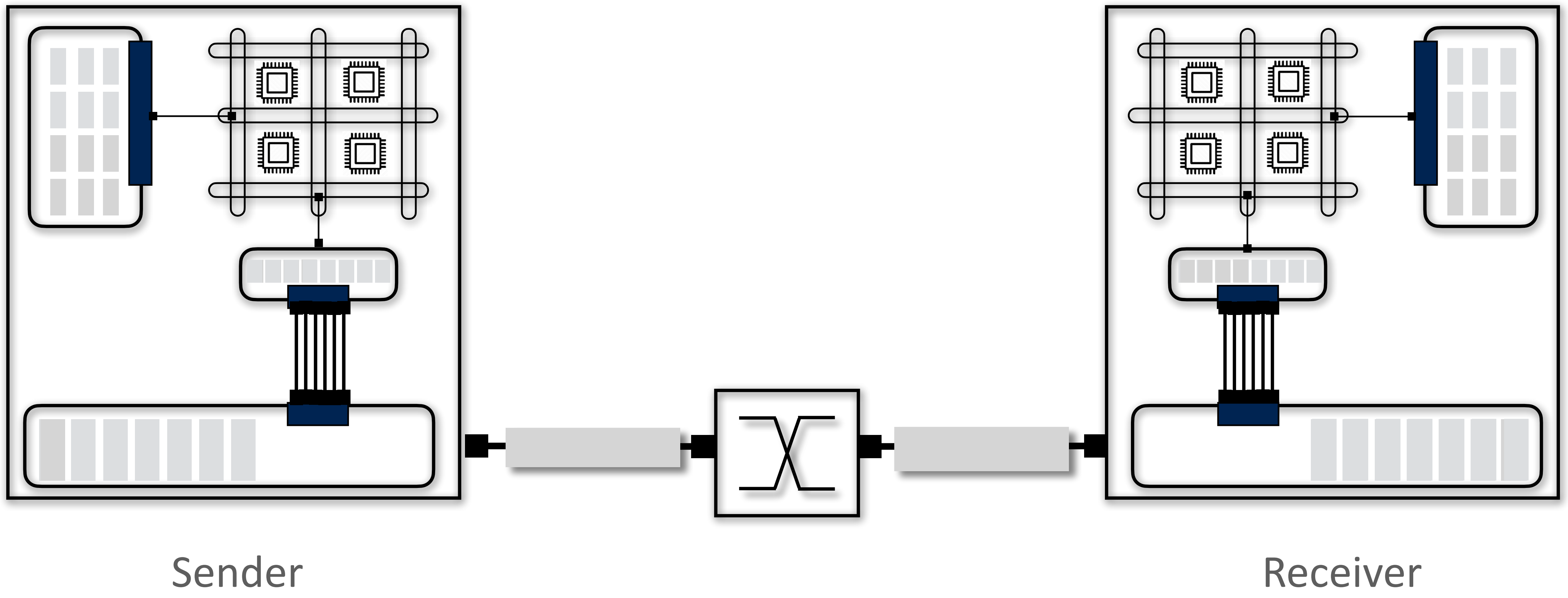
Host Congestion: Impact on Application Performance

Host congestion in Google production cluster

Source: Understanding Host Interconnect Congestion, HotNets 2022



We reproduced host congestion phenomenon using an open sourced stack: Linux + DCTCP



Topology: single sender, single receiver, 100Gbps access links

- No network fabric congestion

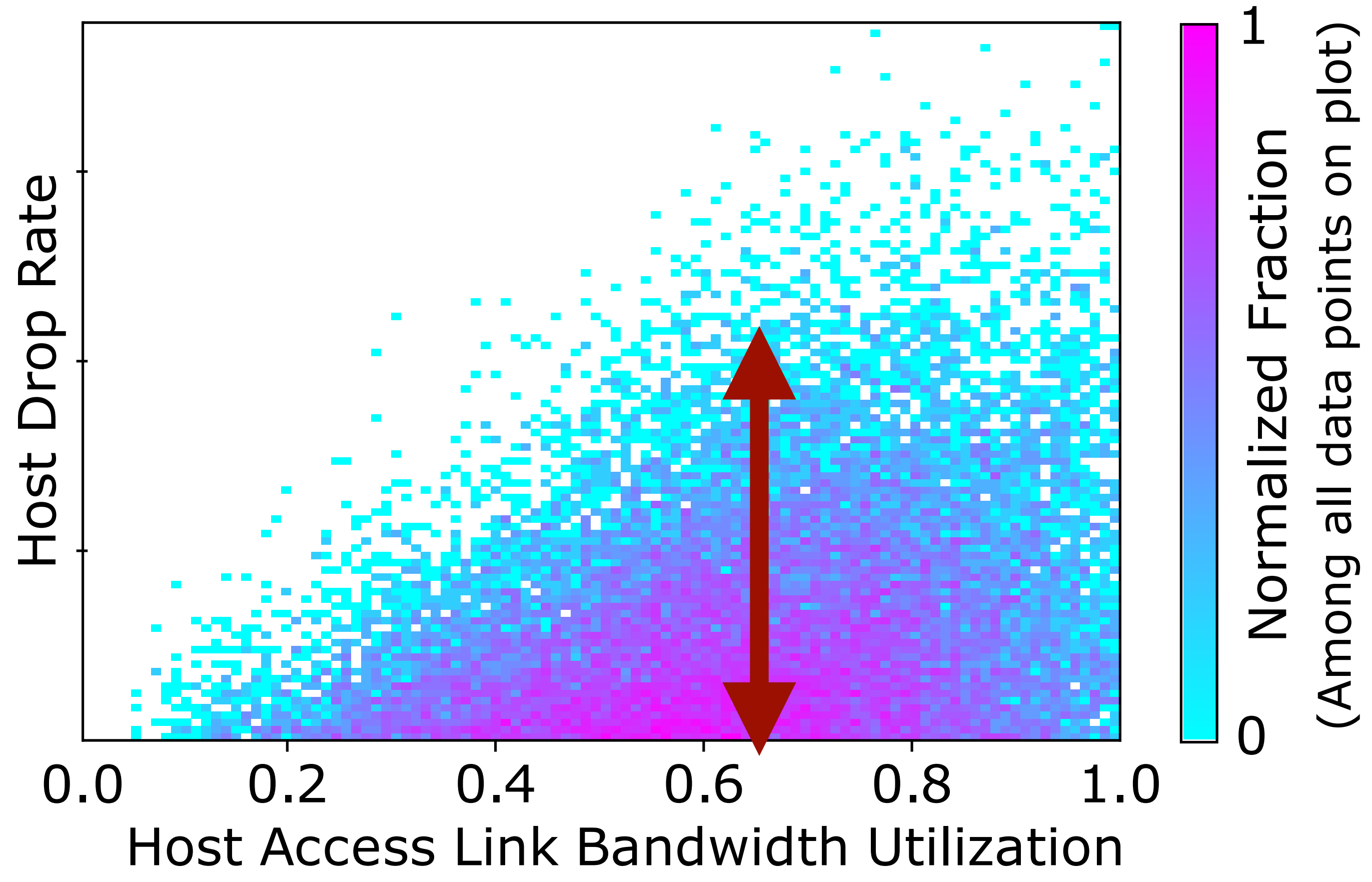
Workload: Multi-tenant scenario

- iperf: **Throughput-intensive** network app
- netperf: **Latency-sensitive** network app
- MLC: **Memory-intensive** host-local app

Host Congestion: Impact on Application Performance

Host congestion in Google production cluster

Source: Understanding Host Interconnect Congestion, HotNets 2022

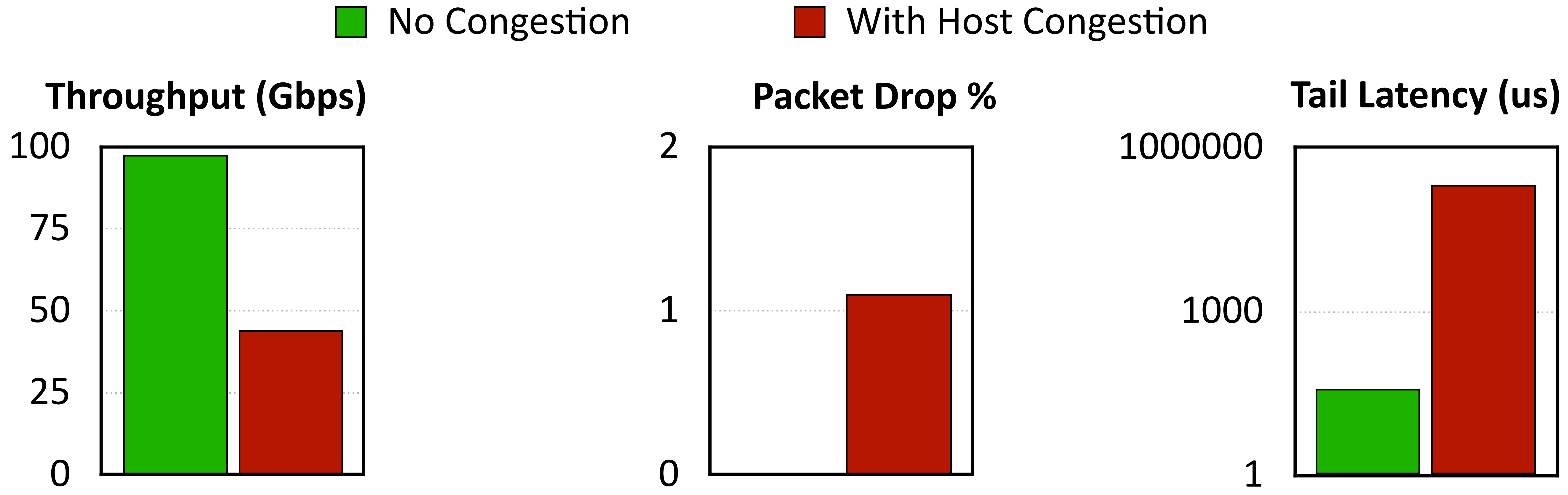


We reproduced host congestion phenomenon using an open sourced stack: Linux + DCTCP

Experiment 1
[Baseline scenario]
Up to 55% reduction in throughput

Experiment 2
[Larger #connections]
Up to 1.1% packet drops => poor isolation

Experiment 3
[Multi-tenant scenario]
Up to 5000x inflation in tail latency

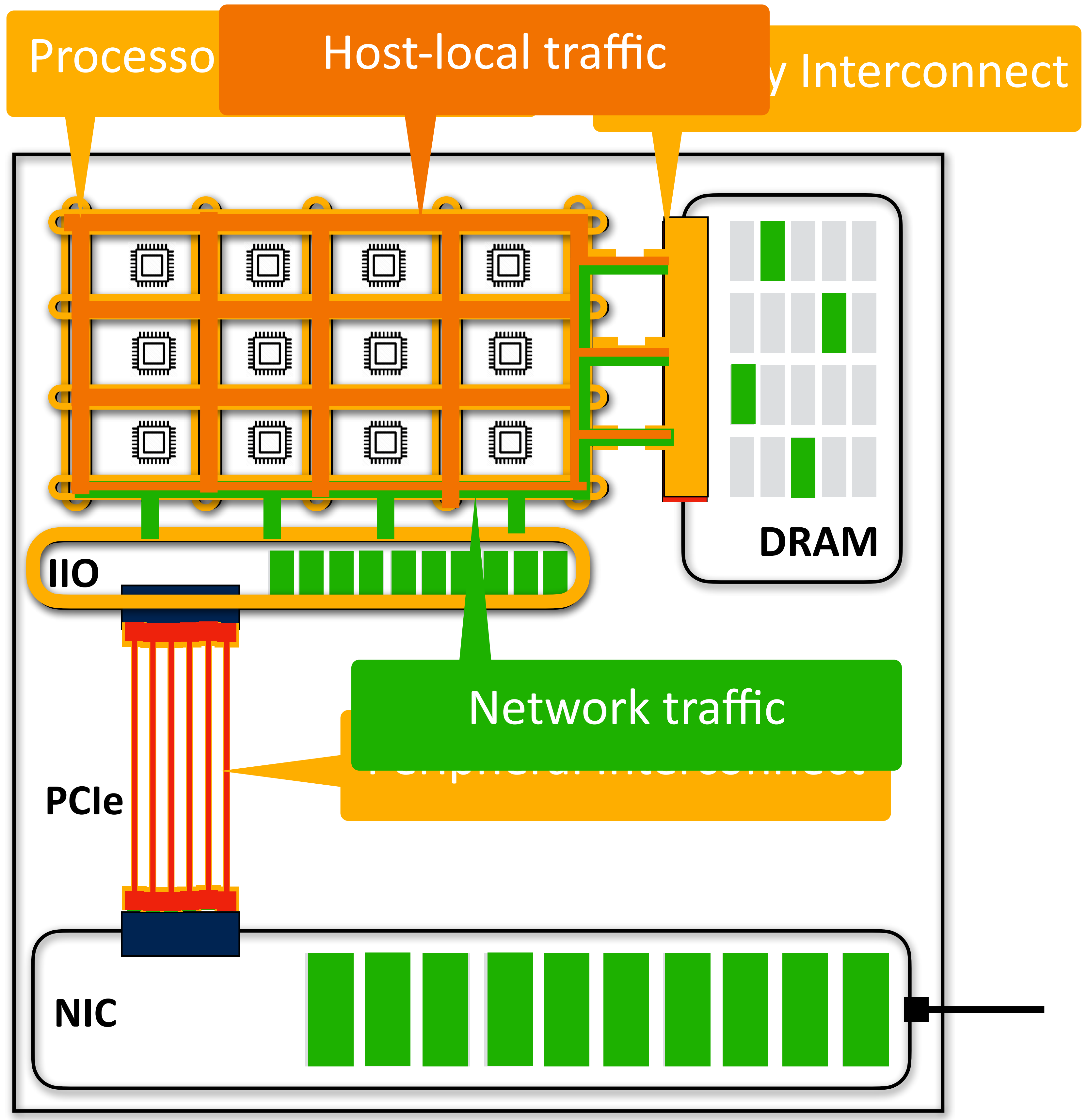


Paper provides workload details and additional results

Our **GitHub repo** provides workloads and infrastructure required to reproduce our results:

https://www.github.com/Terabit-Ethernet/hostCC

Understanding Host Congestion



Host interconnect comprises of three main components

- processor, peripheral and memory interconnect
- help exchange information across NIC, CPUs and DRAM

Host interconnect: a different kind of network fabric

- hardware guarantees losslessness (no data drops)
- is shared by network applications and “host-local” applications

Host Congestion: congestion within the host interconnect

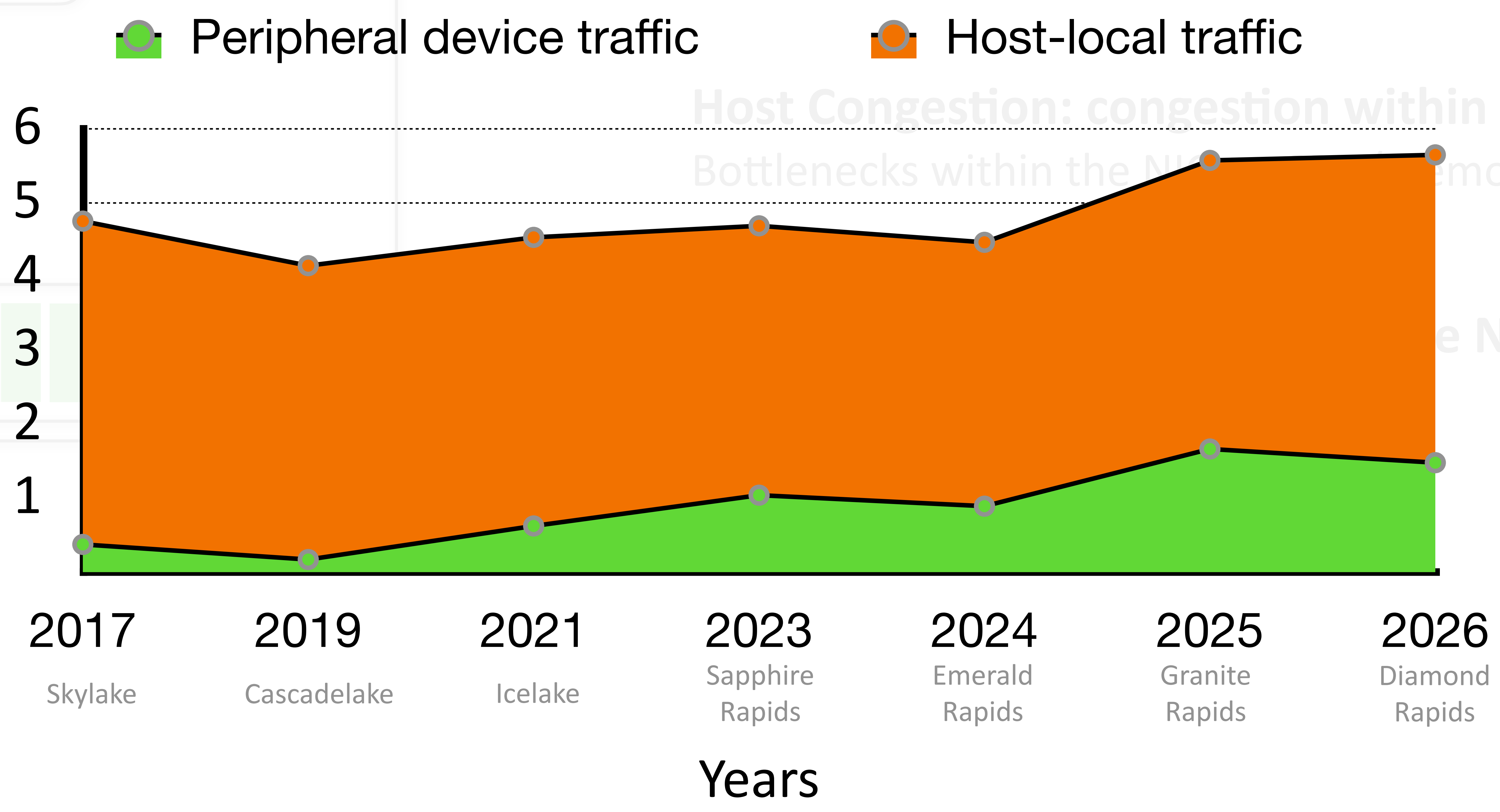
Bottlenecks within the NIC-to-CPU/memory datapath

Result: Queueing and drops at the NIC

Understanding Host Congestion

Problem likely to get even worse over time

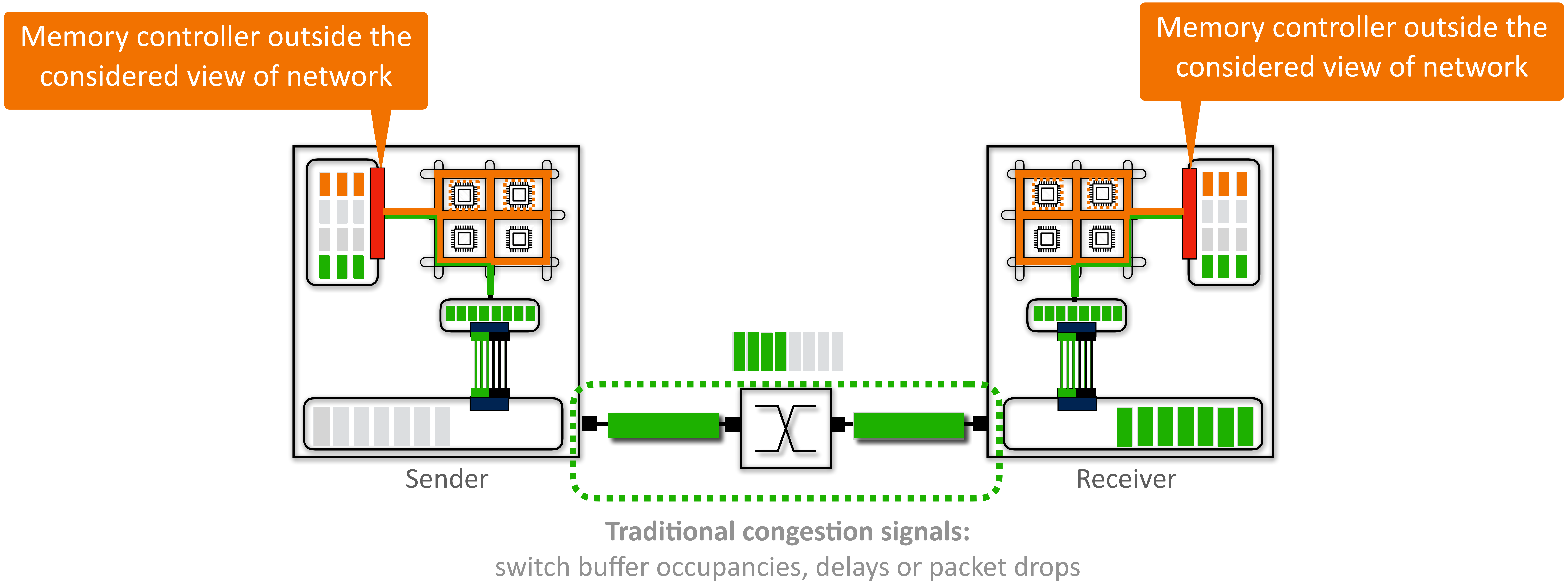
Memory Bandwidth Oversubscription



Host Congestion Control: Rethinking CC Architecture

Rethinking congestion signals

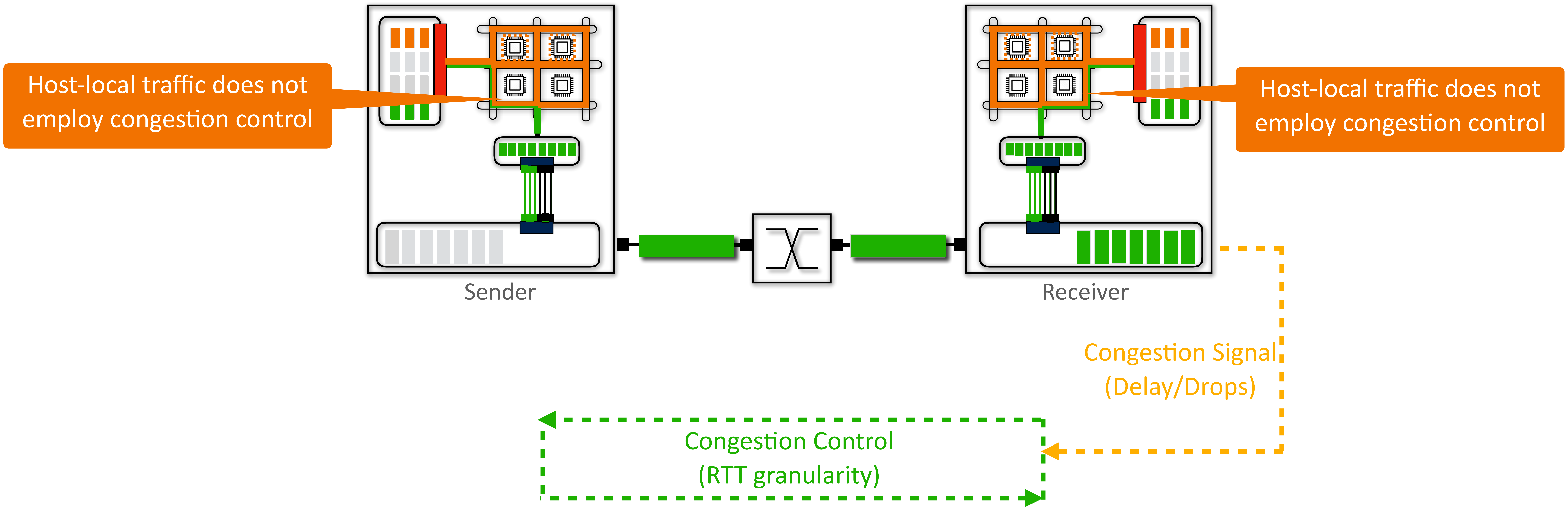
Congestion happening “outside” the network



Host Congestion Control: Rethinking CC Architecture

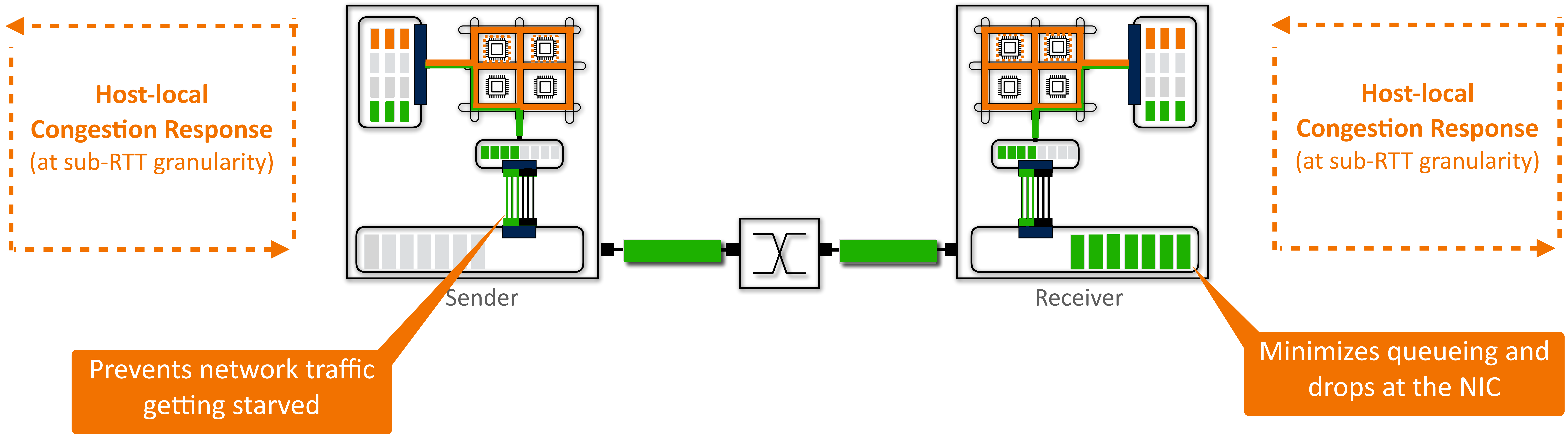
Rethinking congestion response

- **Host-local traffic does not employ CC**
- **CC performed at RTT granularity**



hostCC: A new CC Architecture for Host and Network Congestion

Key idea: Host-local congestion response, at sub-RTT granularity

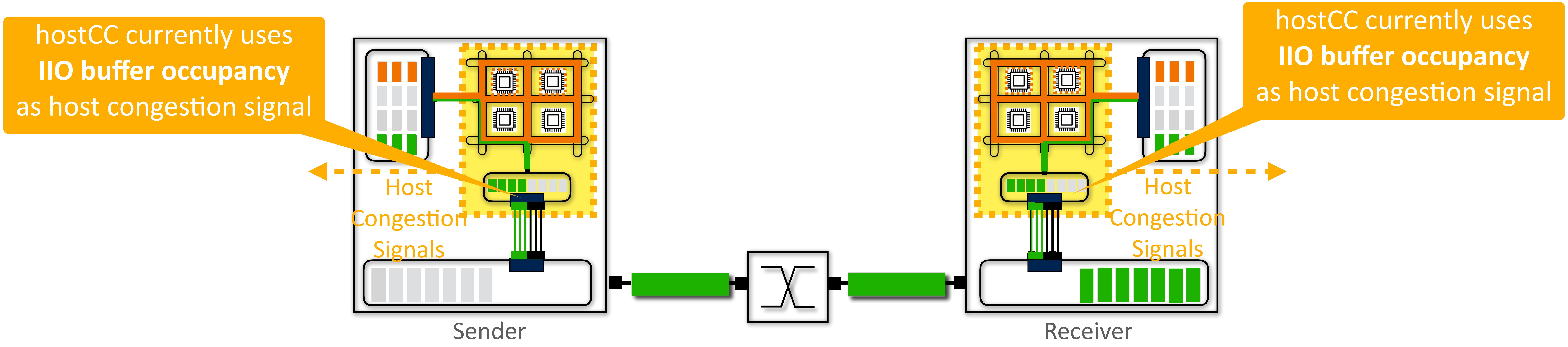


hostCC: End-to-end Overview

1. Host Congestion Signals

At sub-μs granularity

Using commodity hardware



hostCC: End-to-end Overview

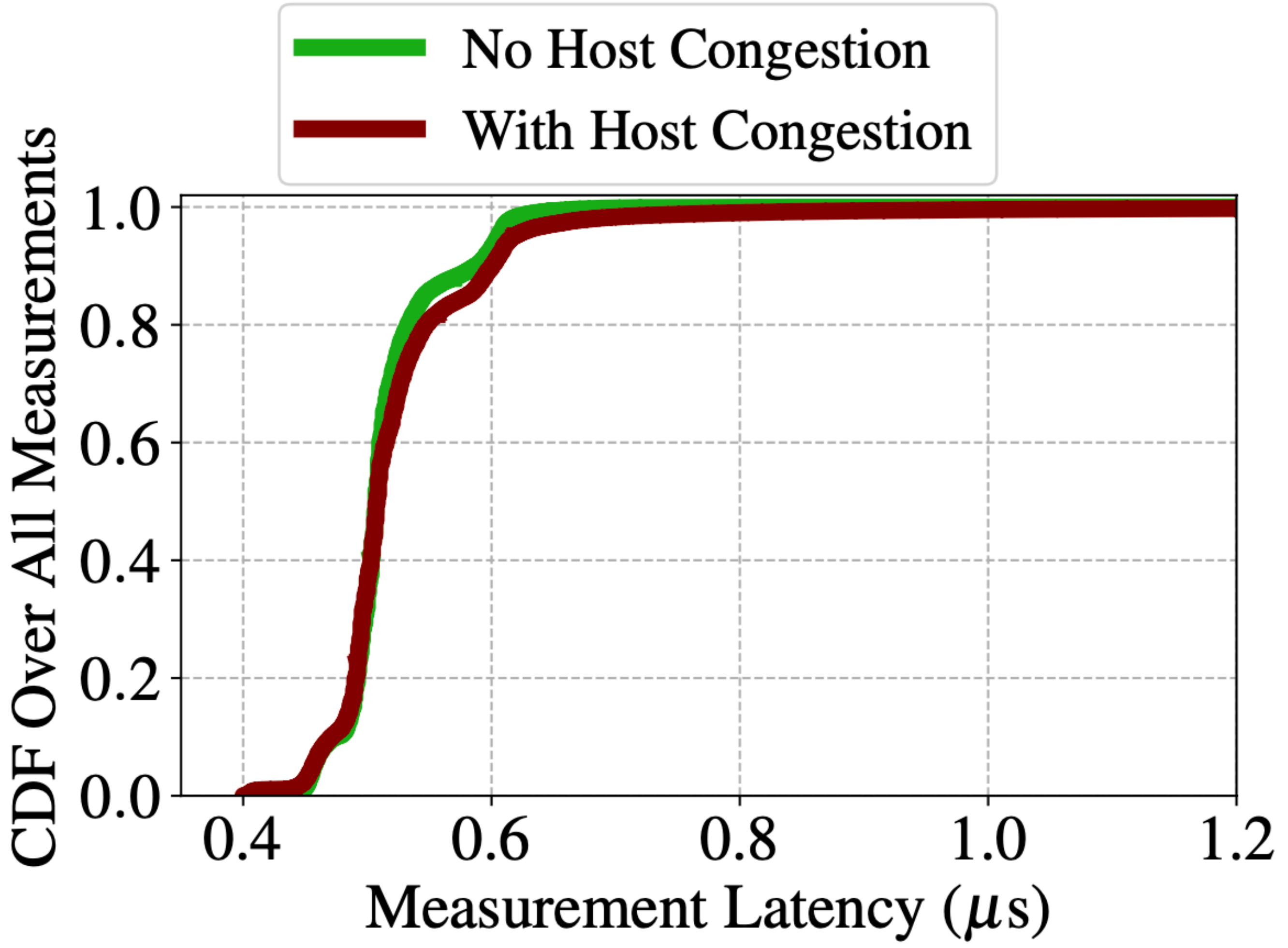
1. Host Congestion Signals

At sub-µs granularity

Using commodity hardware

µs-scale Behavior of IIO Occupancy

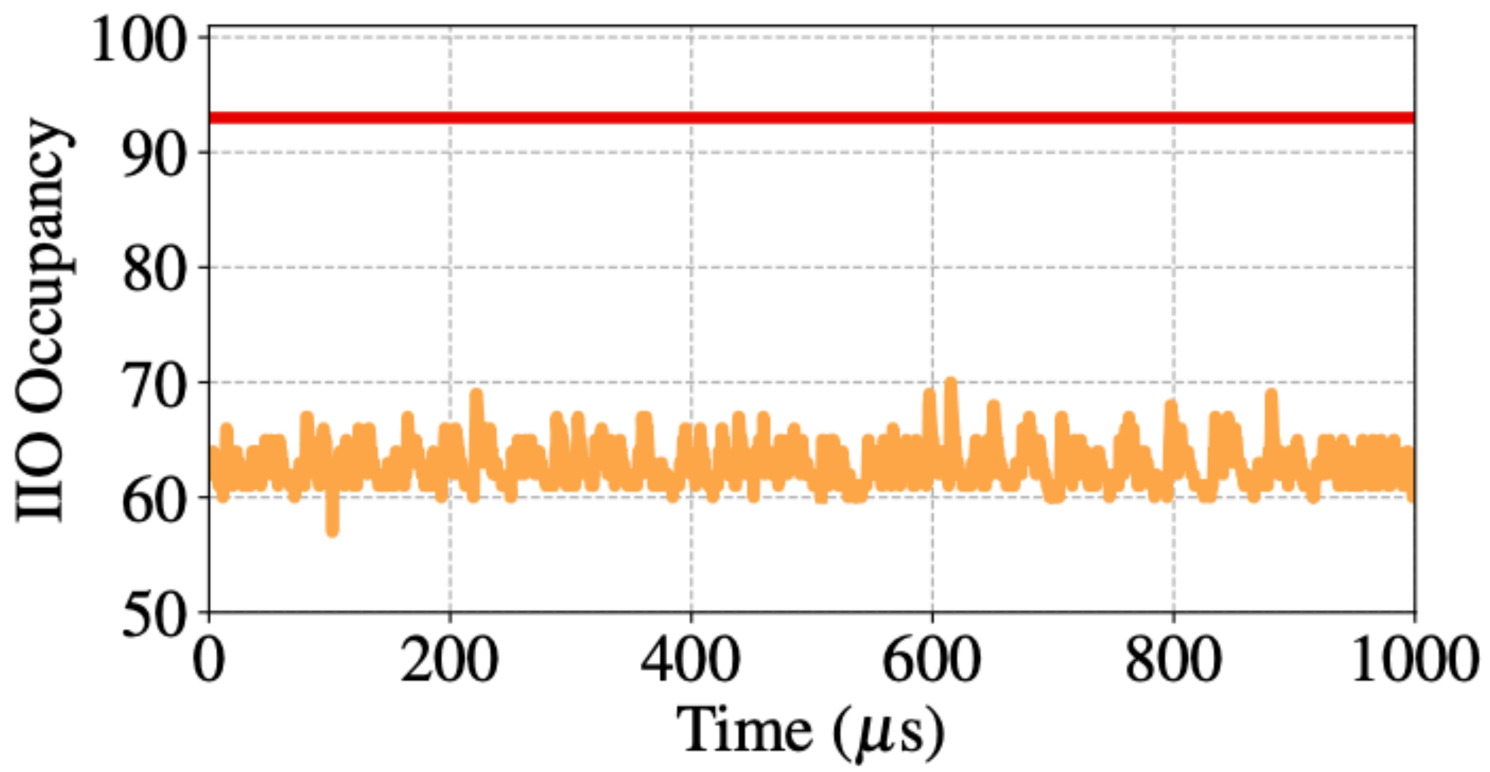
CDF for IIO Occupancy Measurement Latency



Measurement latency <~600ns, **Independent** of host congestion

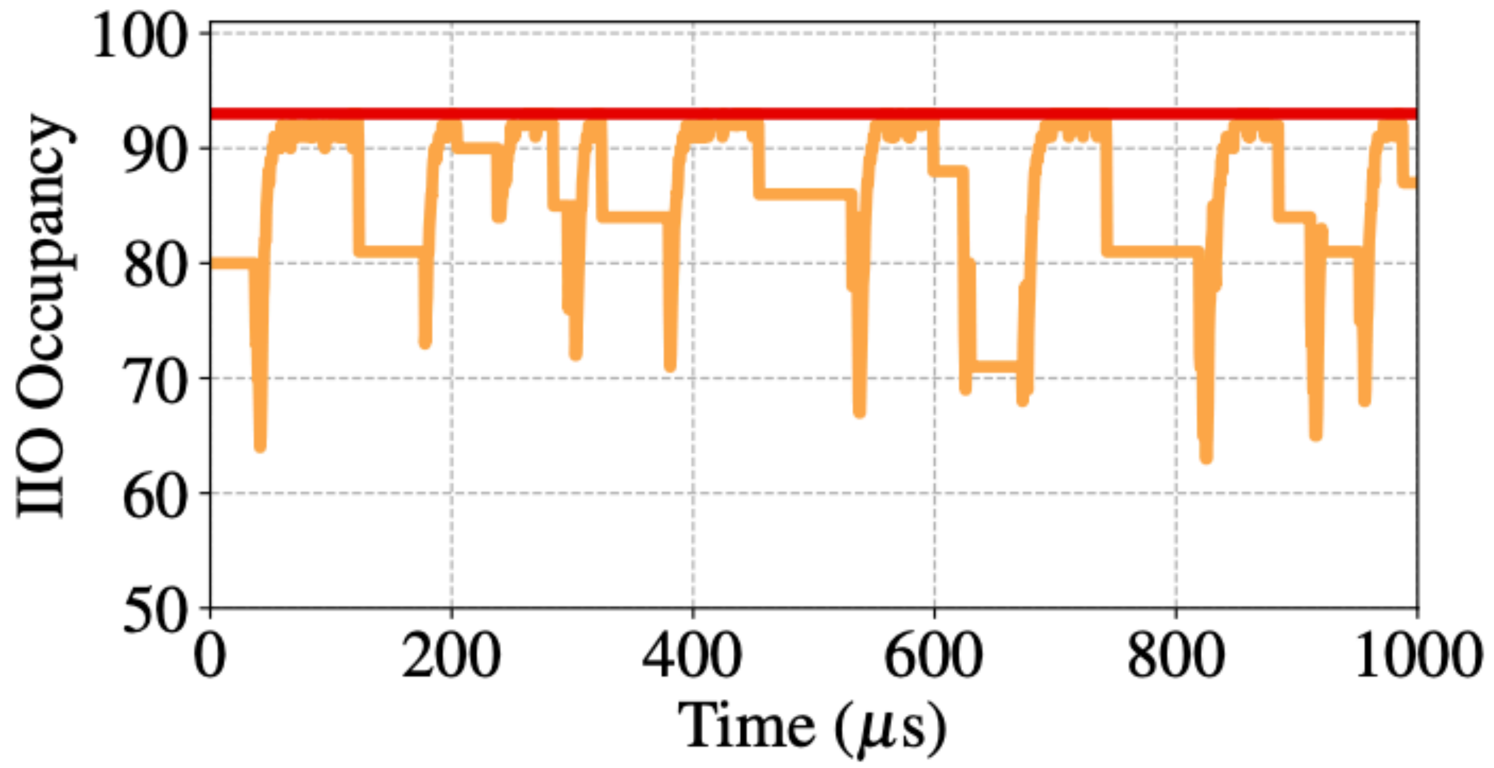
IIO Occupancy Behavior

No Host Congestion



IIO occupancy ~65 cachelines under no host congestion scenario

With Host Congestion



IIO occupancy saturates to max value of ~92 cachelines

hostCC: End-to-end Overview

1. Host Congestion Signals

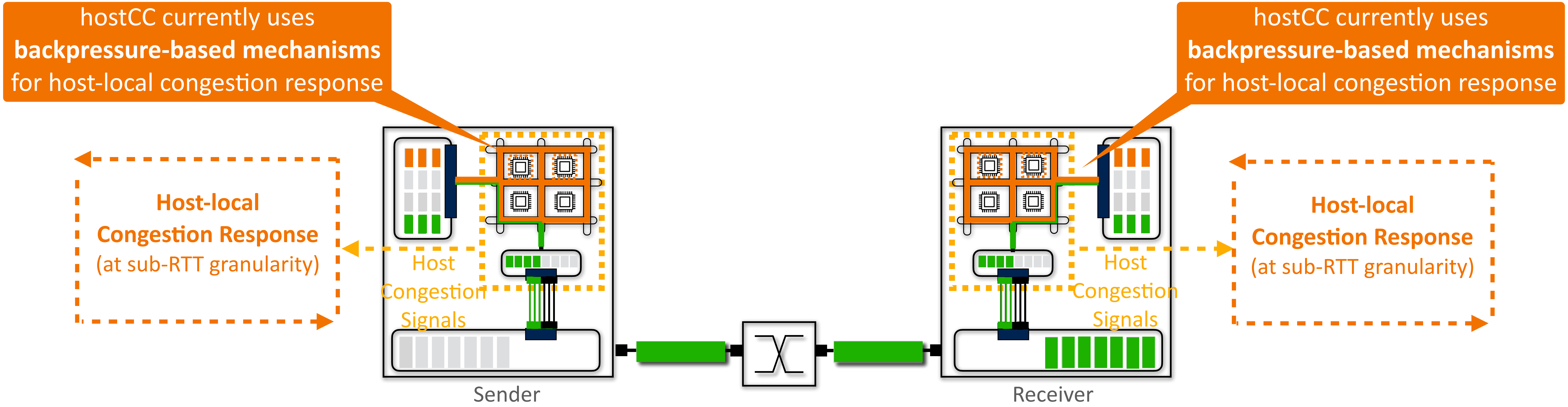
At sub-µs granularity

Using commodity hardware

2. Host-local Congestion Response

At sub-RTT granularity

No changes to applications/hardware



hostCC: End-to-end Overview

1. Host Congestion Signals

At sub-µs granularity

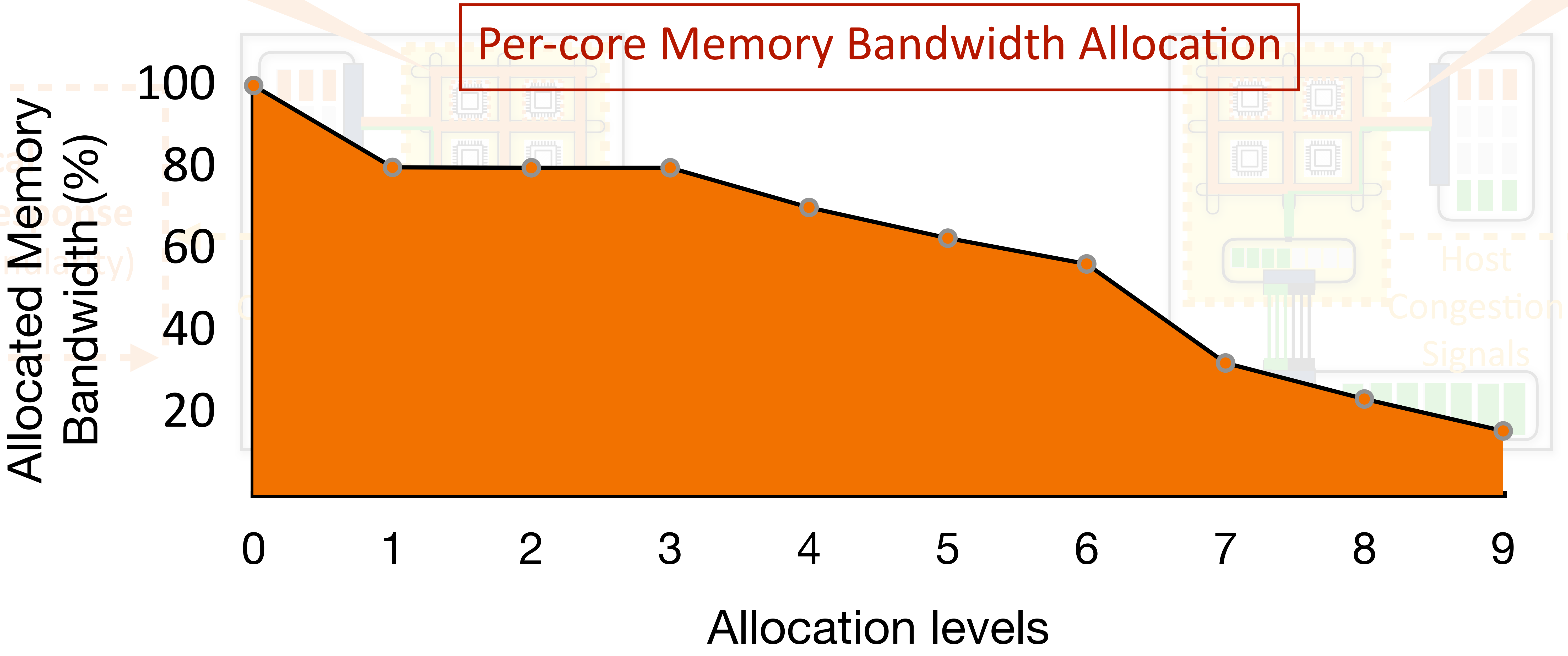
Using commodity hardware

2. Host-local Congestion Response

At sub-RTT granularity

No changes to applications/hardware

Example tool for backpressure to host-local traffic: Intel MBA



Increasing backpressure for increasing allocation levels

hostCC: End-to-end Overview

1. Host Congestion Signals

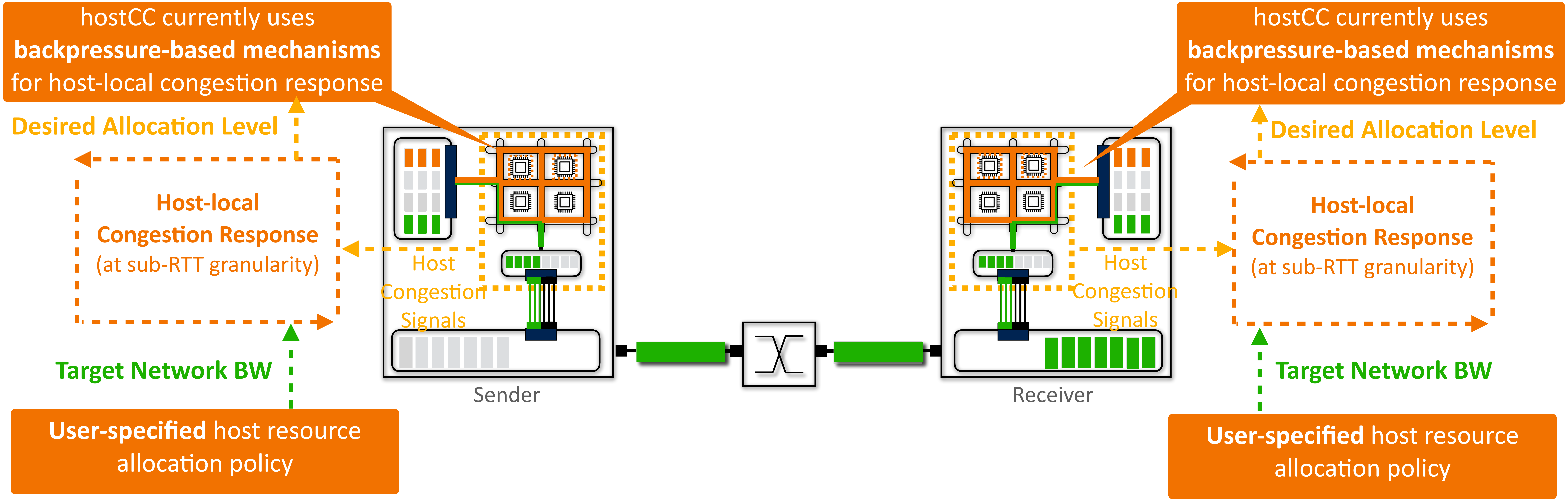
At sub-µs granularity

Using commodity hardware

2. Host-local Congestion Response

At sub-RTT granularity

No changes to applications/hardware



hostCC: End-to-end Overview

1. Host Congestion Signals

At sub-µs granularity

Using commodity hardware

2. Host-local Congestion Response

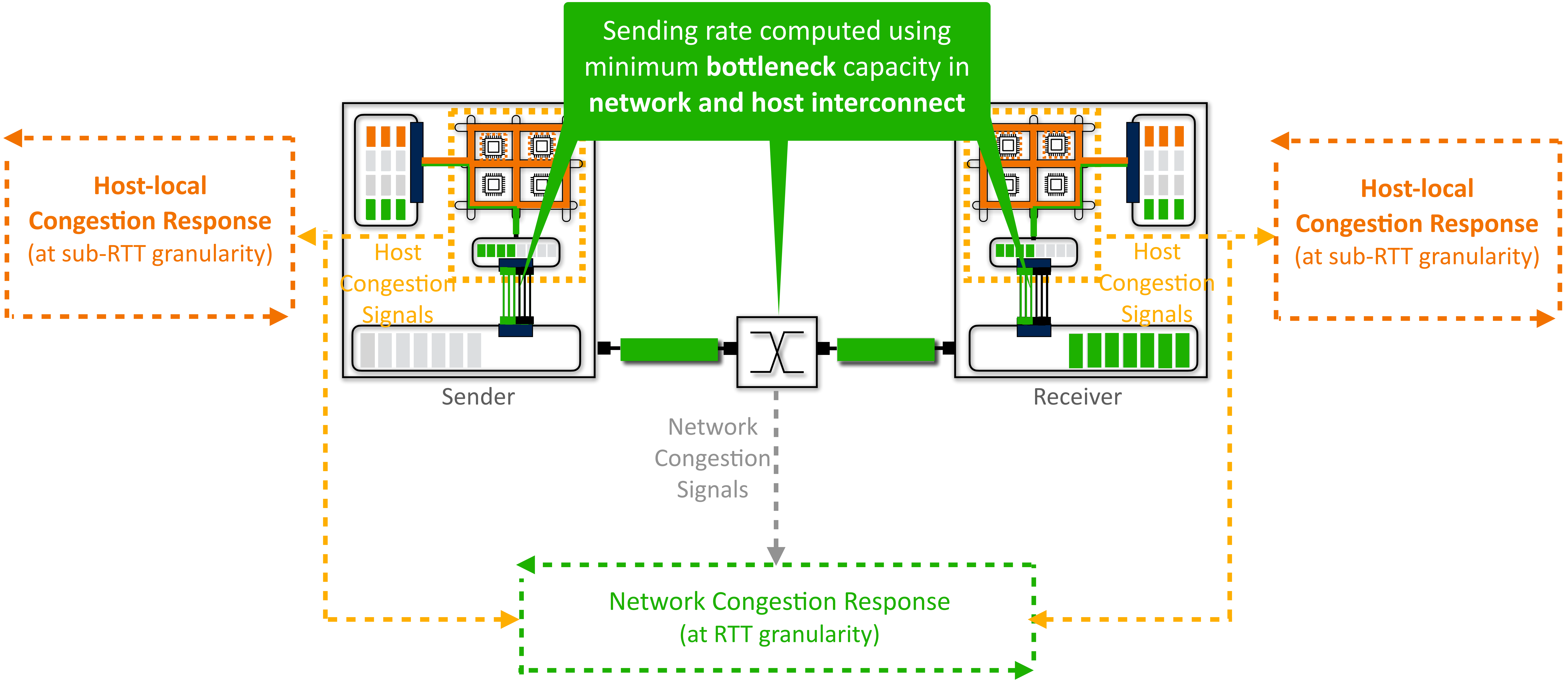
At sub-RTT granularity

No changes to applications/hardware

3. Network Congestion Response

Uses both network & host congestion signals

No changes to network CC protocols



hostCC: End-to-end Overview

1. Host Congestion Signals

At sub-µs granularity

Using commodity hardware

2. Host-local Congestion Response

At sub-RTT granularity

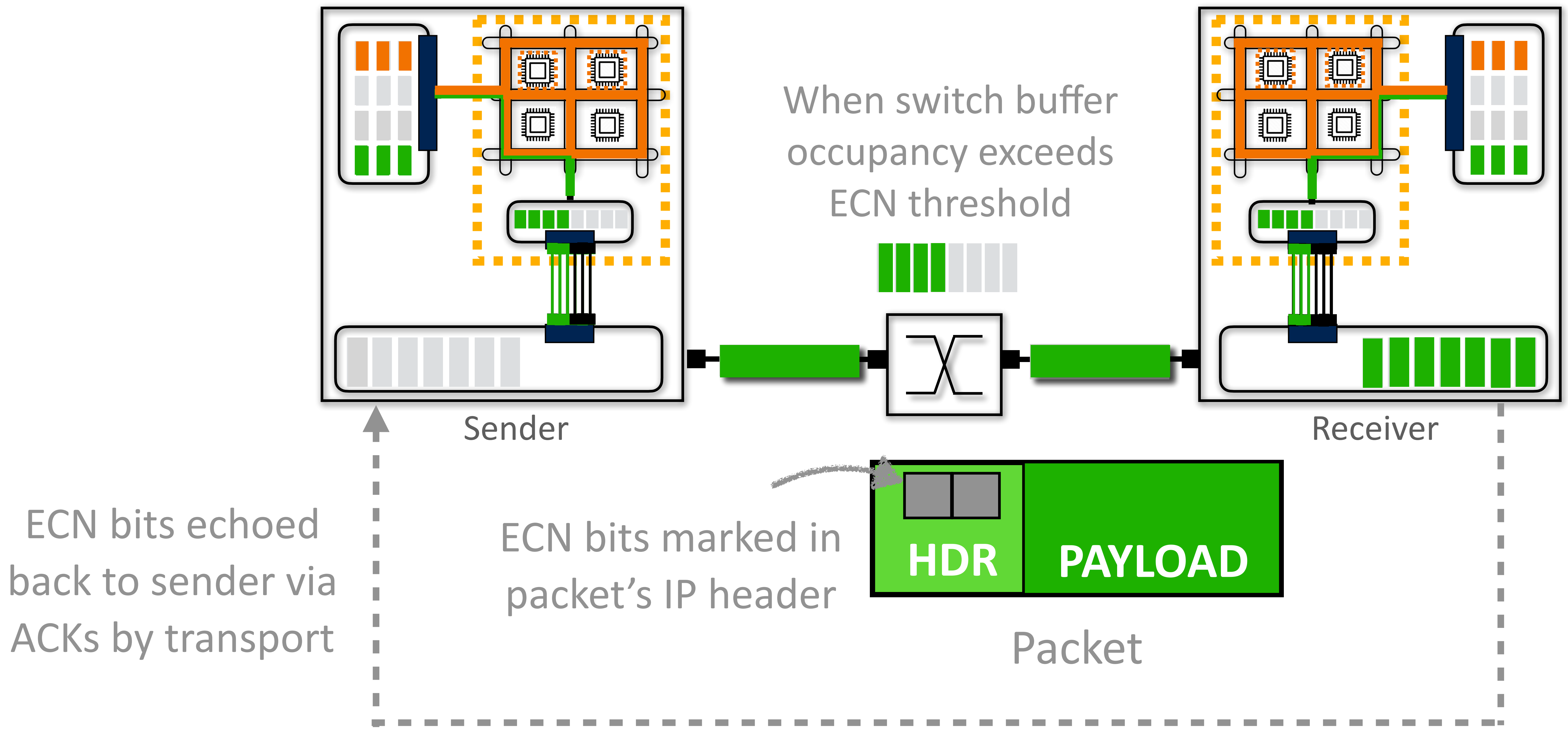
No changes to applications/hardware

3. Network Congestion Response

Uses both network & host congestion signals

No changes to network CC protocols

Example scenario: Using ECN-based network CC protocols



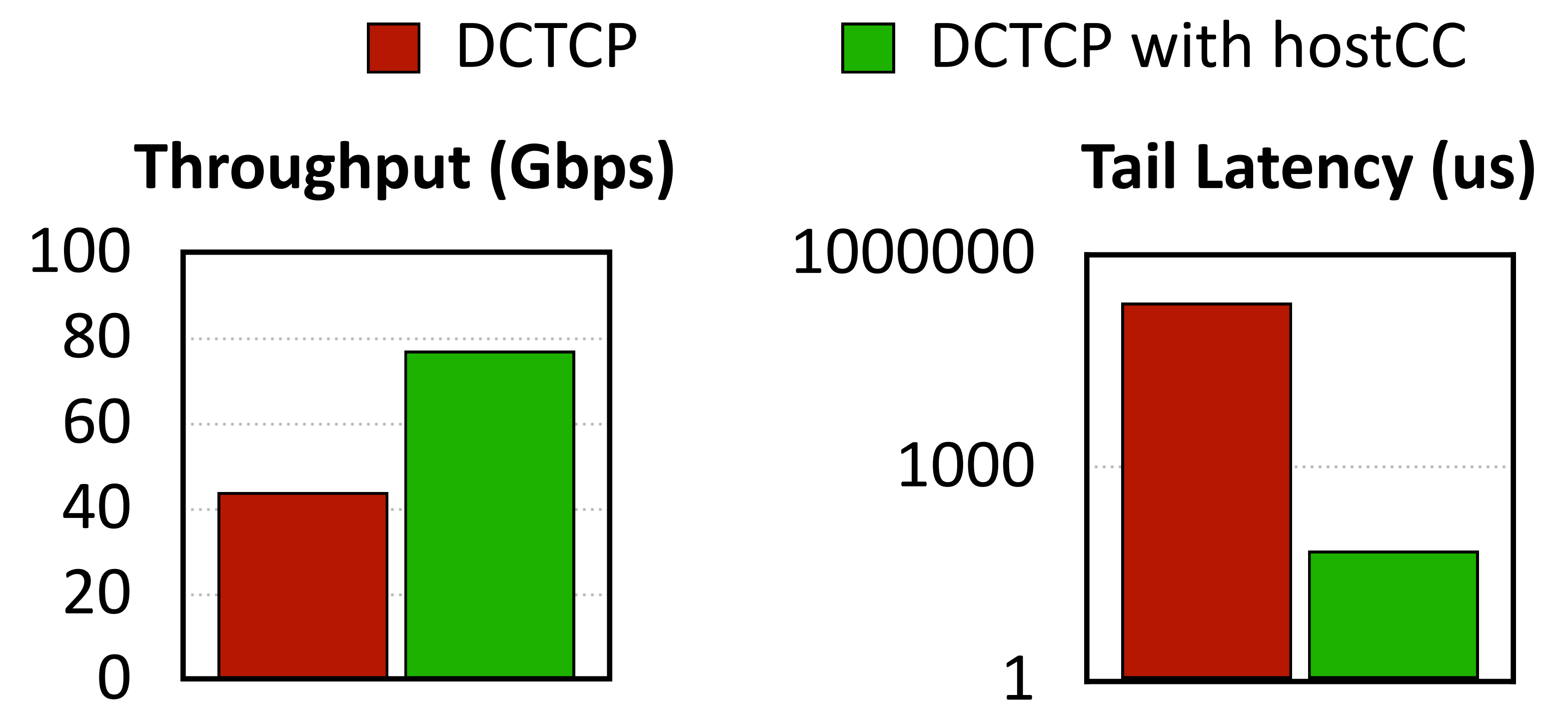
When ~~switch~~ **IIO buffer** occupancy exceeds ECN threshold

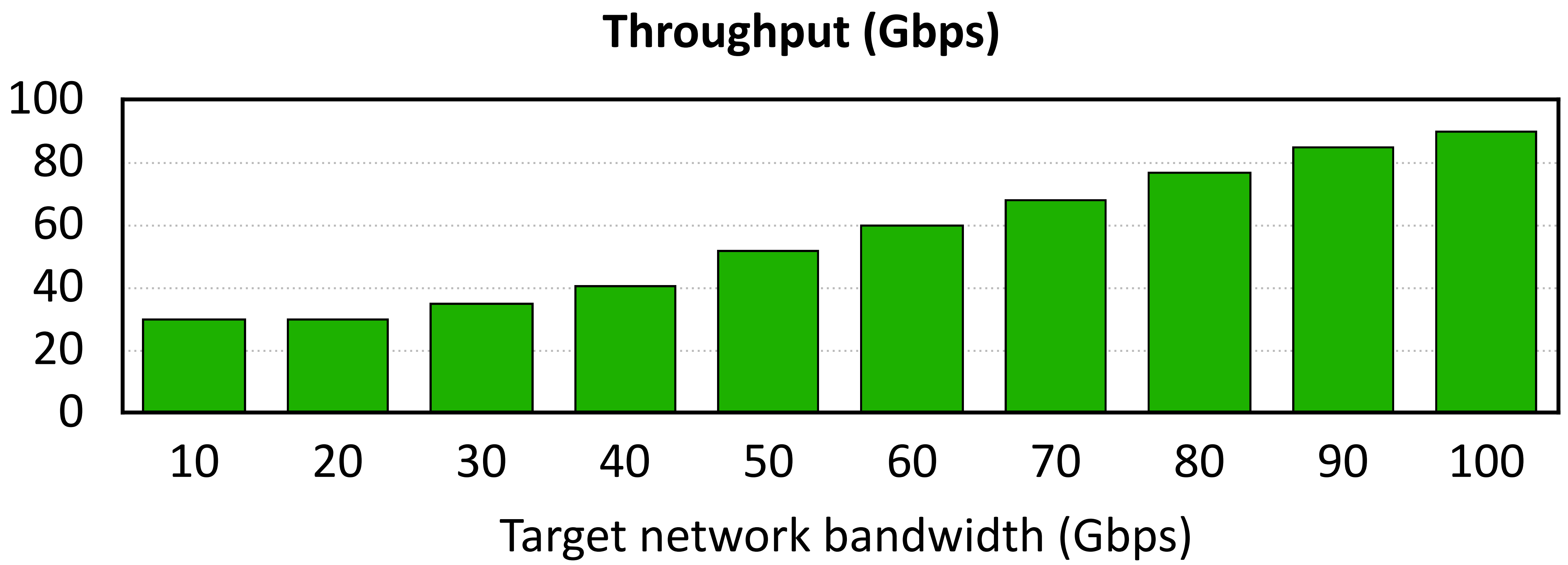
Minimal modifications to the host IP layer stack (<100 LOC using NetFilter hooks)

CC protocols react to ECNs **independent of the source of marking**

No changes required for existing ECN-based protocols (DCTCP, TCP w/ ECN, etc)

hostCC Benefits With Host Congestion





Improved performance under host congestion

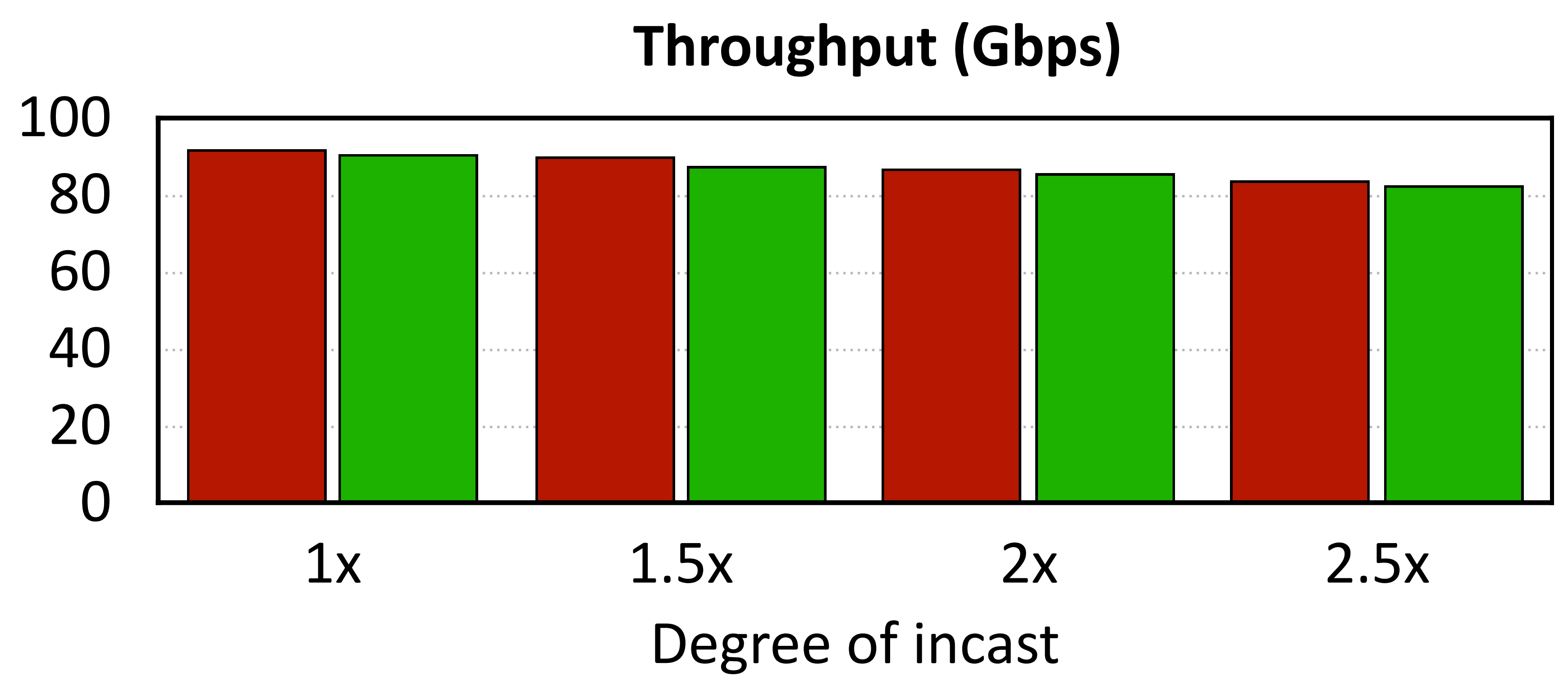
Near-optimal throughput and latency

Reduces queueing/drops to a bare minimum

Enables enforcing desired resource allocation policy

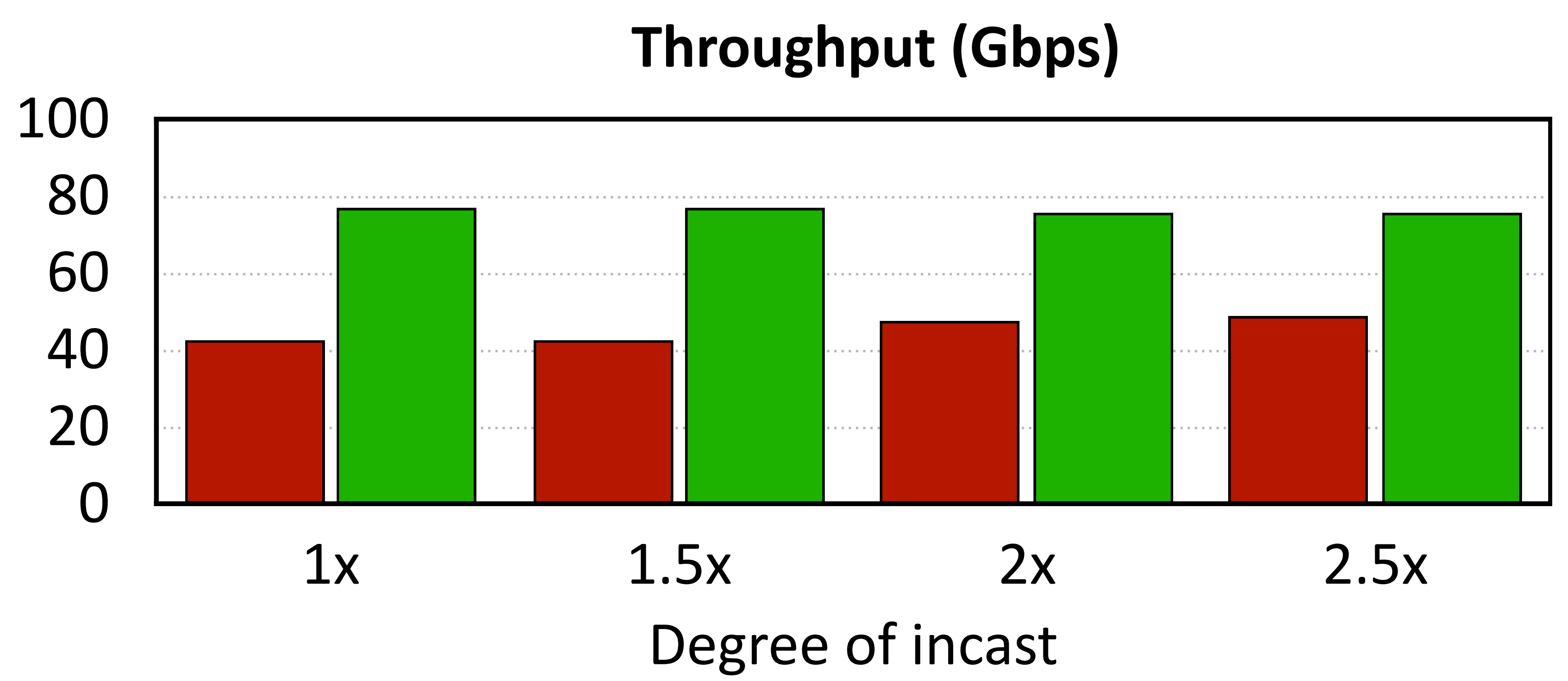
Network traffic close to user-specified target bandwidth

hostCC Benefits With Host Congestion and Network Fabric Congestion



Performance similar to network CC in presence of only network congestion

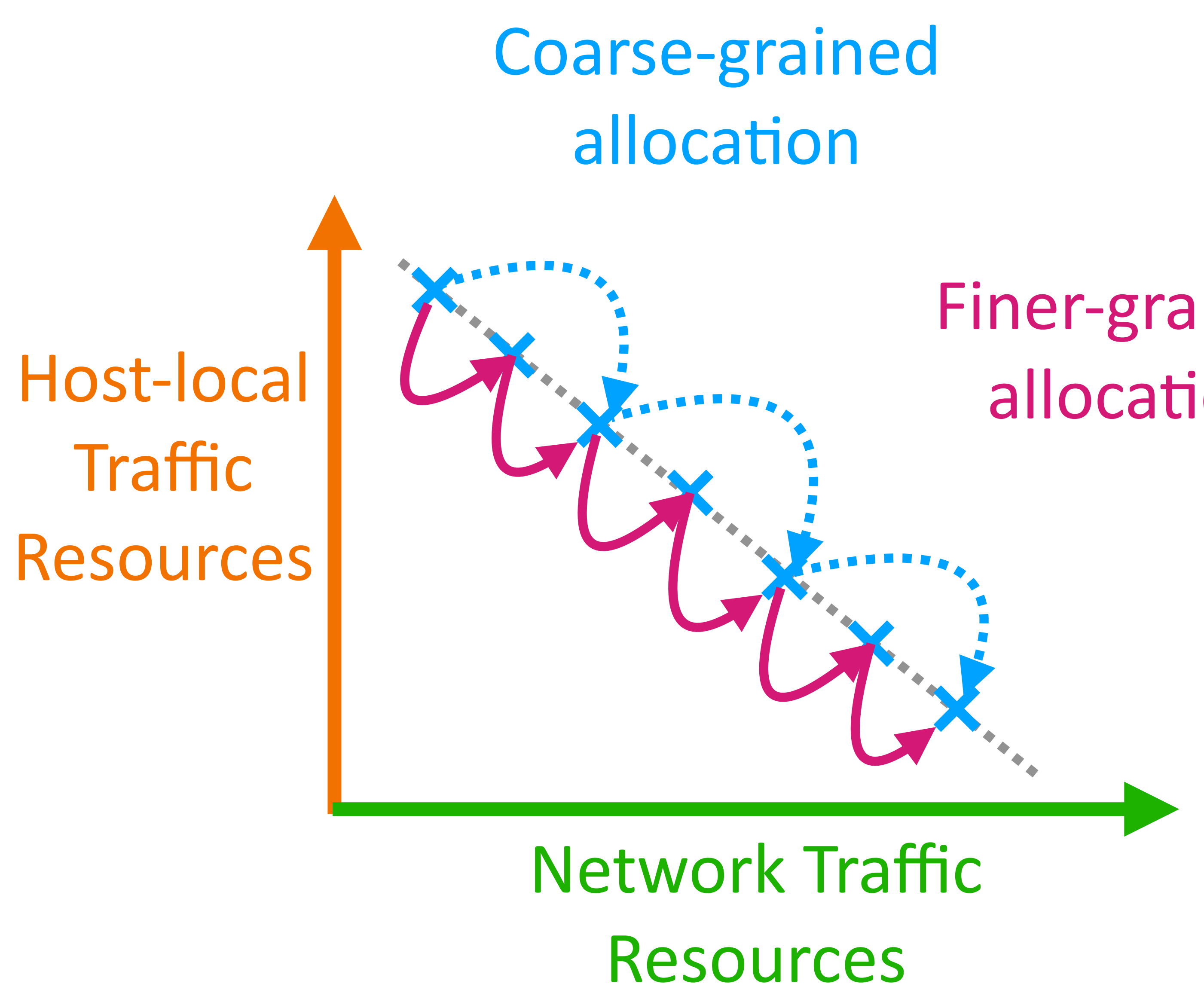
Minimal overheads of using hostCC



Maintains benefits even in presence of both network and host congestion

Interpolates well with network CC

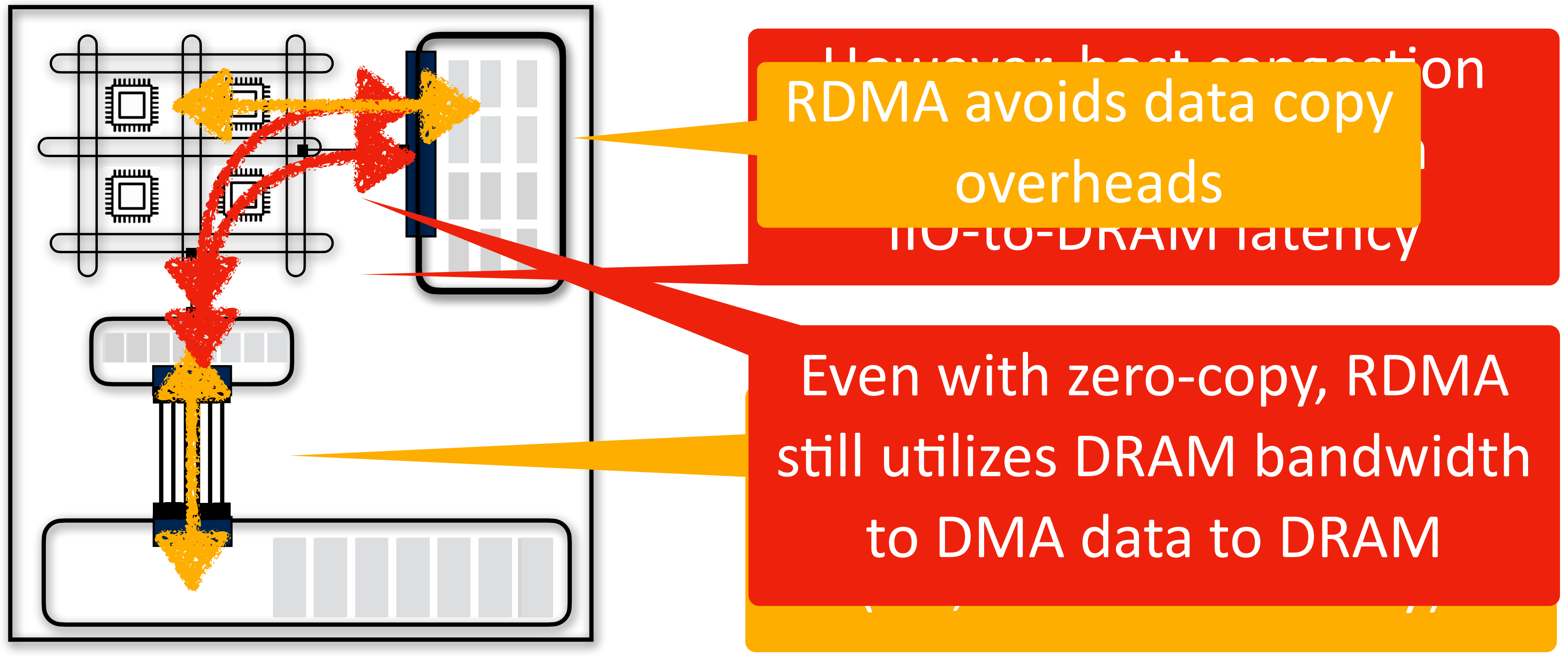
Lessons learnt and future directions



We need new tools for efficient resource allocation

Existing tools too coarse grained

Need tools for finer-grained allocation

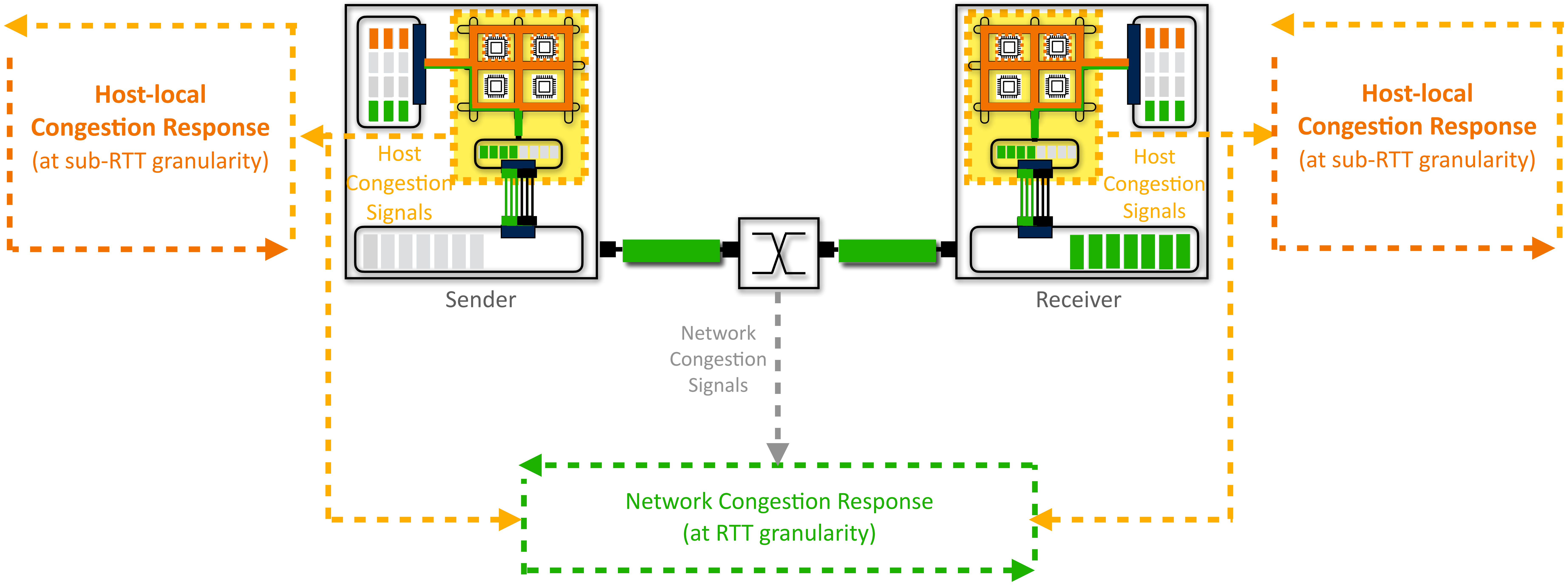


New technologies for solving host congestion

Unclear if CXL will solve the problem

RDMA may not solve the problem by itself

hostCC: A CC architecture that handles host and network fabric congestion



hostCC Linux implementation & workloads to reproduce our results are available at **www.github.com/Terabit-Ethernet/hostCC**

hostCC project webpage: **www.cs.cornell.edu/~saksham/hostcc**