CKN Series
Evolved Networking: the AI/ML Challenge
Cisco Silicon One

Rakesh Chopra
Cisco Fellow
December 1, 2022

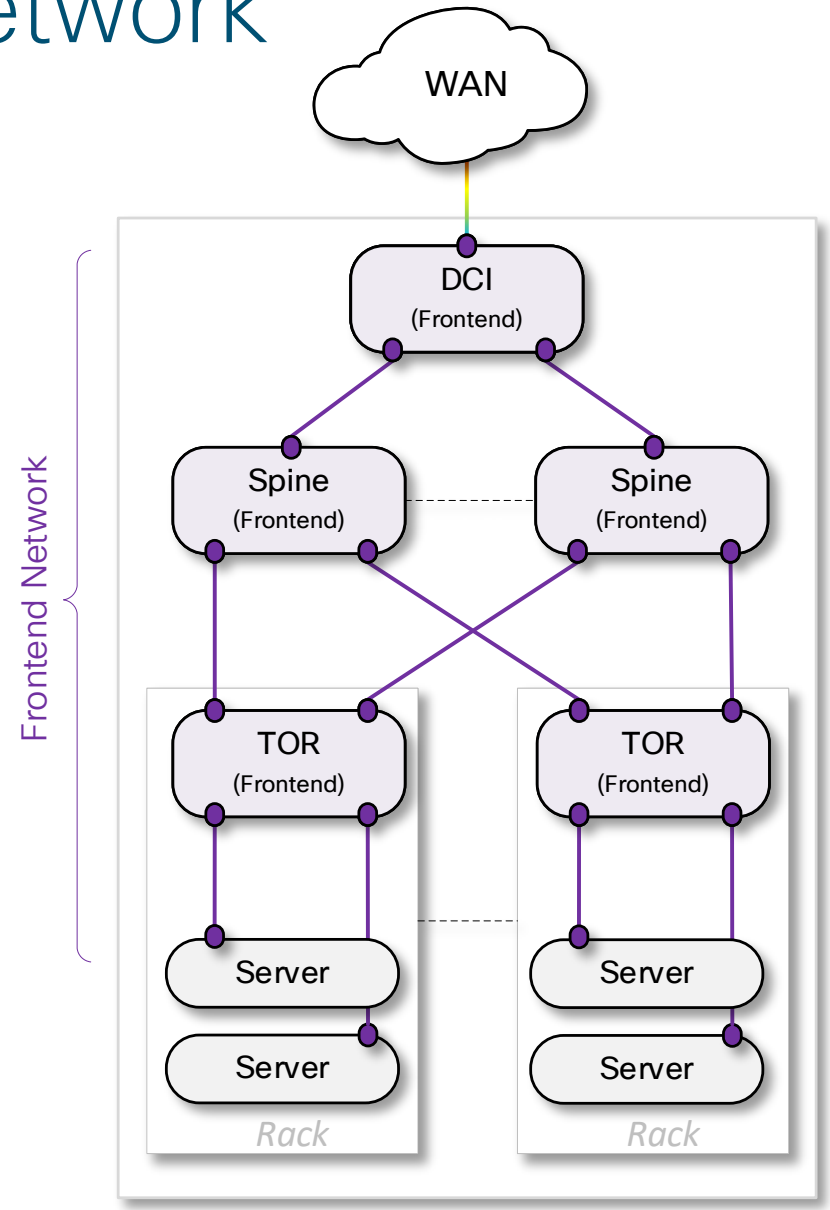
rakchopr@cisco.com
www.linkedin.com/in/rakesh-chopra/
@Rakesh_Chopra1

Special thanks to Nadav Chachmon & Ofer Iny for their incredible contributions to this presentation

Web Scale Frontend (FE) Network

Front End Network

Network's Purpose
Connecting Servers (x86, ARM, etc…) Connecting Servers to Internet
Network Bandwidth Drivers
Migration to the Cloud
Meeting the Bandwidth Needs
Y/Y CPU & Network Increase Scale-out within a generation
Interconnect
Ethernet (Massive Investment)



Network topology drawn for simplicity over accuracy

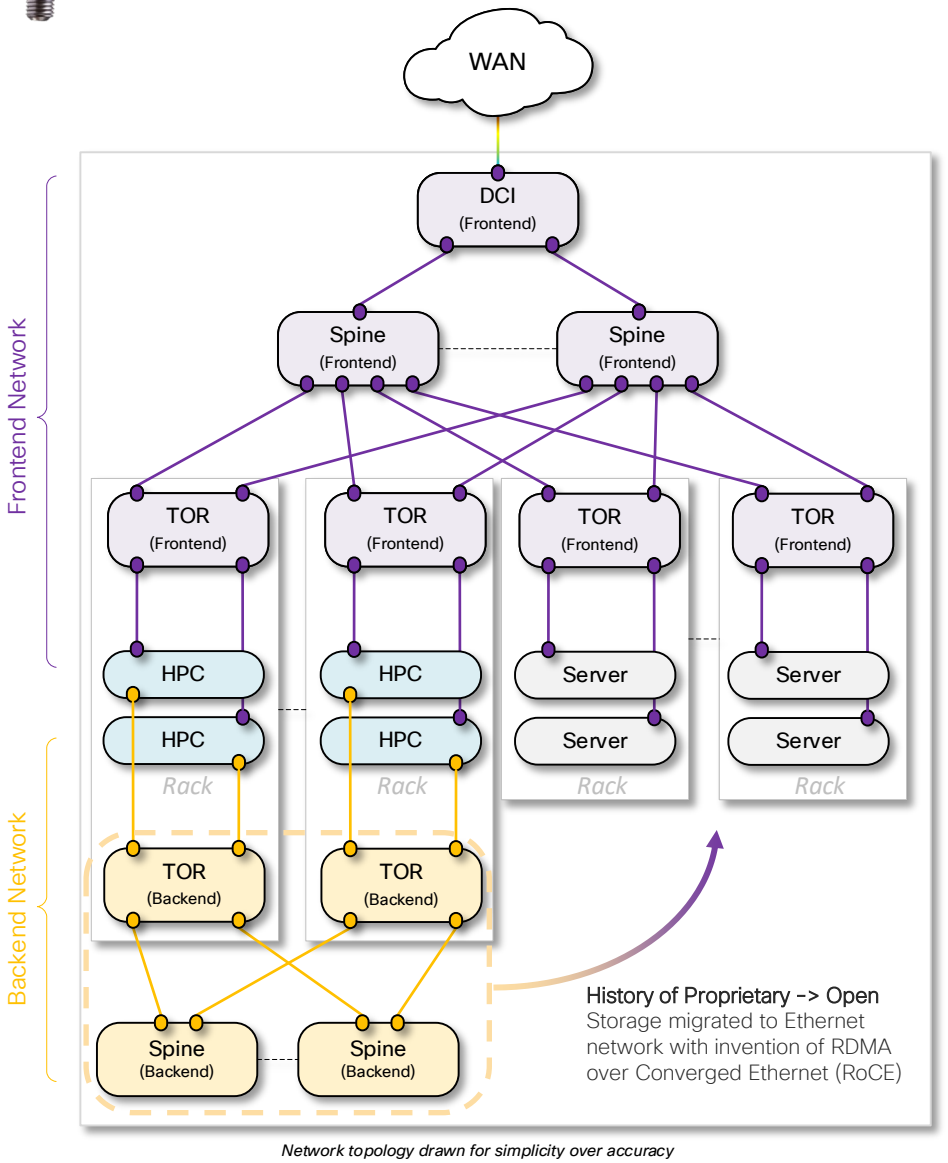
Ethernet has a huge investment and install base. Bandwidth doubles every 18-24 months

Web Scale Frontend (FE) & Backend (BE) Network

Front End Network	Back End Network
Network's Purpose	
Connecting Servers (x86, ARM, etc…) Connecting Servers to Internet	Connecting Specialized End-Points (GPUs, Storage*, etc…) No connection to the Internet
Network Bandwidth Drivers	
Migration to the Cloud	HPC & Storage ⬇ Explosion of AI/ML ⬇ Step function increase in bandwidth
Meeting the Bandwidth Needs	
Y/Y CPU & Network Increase Scale-out within a generation	Y/Y GPU & Network Increase Massive new build-outs
Interconnect	
Ethernet (Massive Investment)	Proprietary** (Limited Investment) ⬇ Ethernet (Massive Investment)

* - Many networks use RDMA over Ethernet to move storage to the frontend network
** - Some networks use InfiniBand

AI/ML needs a different solution then HPC



Network topology drawn for simplicity over accuracy

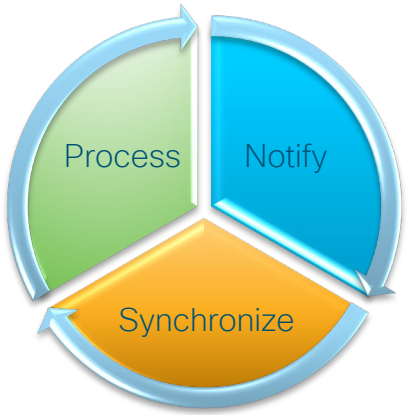
AI/ML Workload Challenge

Different from traditional data center traffic

The network is fundamental.
Making one wrong path selection stalls the entire AI job

Execute instructions on GPU

High bandwidth compute can saturate network links



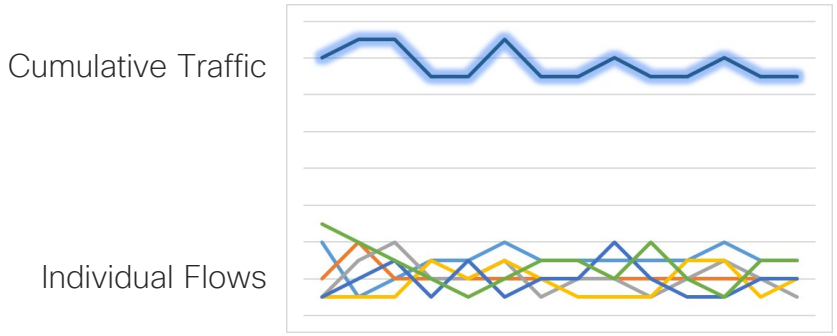
Send results of computation

Several methods, we'll focus just on one
All-to-All Collective (Everyone sends to everyone)

Wait for everyone to complete

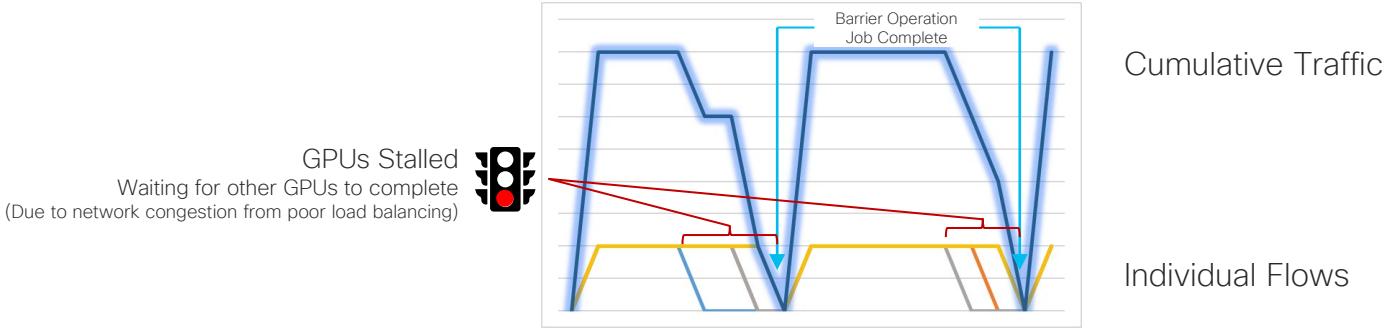
Creates synchronization between GPUs
Computation stalls waiting for the slowest path
Job Completion Time (JCT) is based on the *worst-case* tail latency

Traditional DC Traffic Pattern



Many asynchronous small BW flows
Chaotic pattern averages out to **consistent load**

AI (All-to-all Collective) Traffic Pattern



Few synchronous high BW flows
Synchronization **magnifies long tail latency & bad load balancing decisions**

AI/ML Workload Challenge

Different from traditional HPC

Tools for HPC don't scale to solve AI/ML needs

HPC

One Job

AI/ML

Many Jobs

Interference between jobs slows all jobs down

Since all GPUs stall based on the slowest path, this interface is very visible

Increasing Importance of the Network

Traditional DC
(Front End Network)

Compute bound

HPC
(Back End Network)

Mostly compute bound

AI/ML
(Back End Network)

Synchronization stalls compute
Mostly Network Bound

Load Balancing Decisions in AI/ML is Critical

New

AI/ML Load Balancing Options

Ethernet

- Good – Stateless Flow Placement (ECMP)
  - Hash based selection
  - To avoid polarization hazards
    - Flexible field selection
    - Multiple Hash functions
  - Effective load balancing database (WCMP) maybe helpful
  - Effectiveness depends on traffic load characteristics*

Ethernet with Telemetry

- Better – Stateful flow/flowlet placement
  - Telemetry based selection
  - Effectiveness depends on traffic load characteristics*

Fully Scheduled

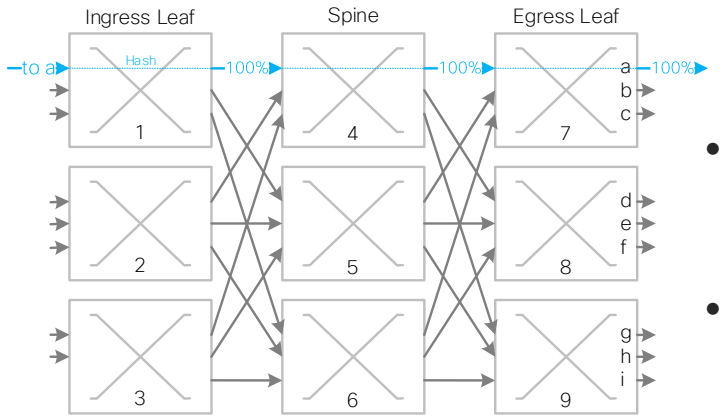
- Best – Fully Scheduled fabric with Spray & Re-order
  - Combination of end to end scheduled with packet spraying
  - Traffic load characteristics independent performance

*- Flow bandwidth, number of flows, duration of flows, gaps in flow, traffic spread/locality, hash functions

AI/ML Load Balancing Options

Example : Ethernet (Good) – 1 Flow

Good | Ethernet
Better | Ethernet with Telemetry
Best | Scheduled Fabric

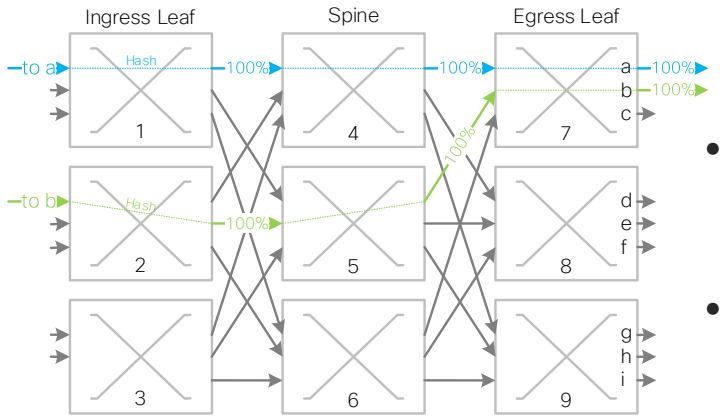


- Ingress Leaf (1)
  - Receives traffic on an input port
  - Looks up the destination of the packet to identify eligible ports
    - All leaf to spine ports can reach egress port (A)
  - Selects an output port with a hash on info in the packet
    - example; {Src IP, Src Port, Dest IP, Dest Port, Protocol}
    - In this case it selects the link to Spine 4
- Spine (4)
  - Looks up the destination of the packet to identify eligible ports
  - Only one port can reach egress port (A)
- Egress Leaf (7)
  - Looks up the destination of the packet to identify eligible ports
  - Egress port (A) is directly connected
- 100% of the traffic passes through the network
  - To-A is 100%

AI/ML Load Balancing Options

Example : Ethernet (Good) – 2 Flows

Good | Ethernet
Better | Ethernet with Telemetry
Best | Scheduled Fabric



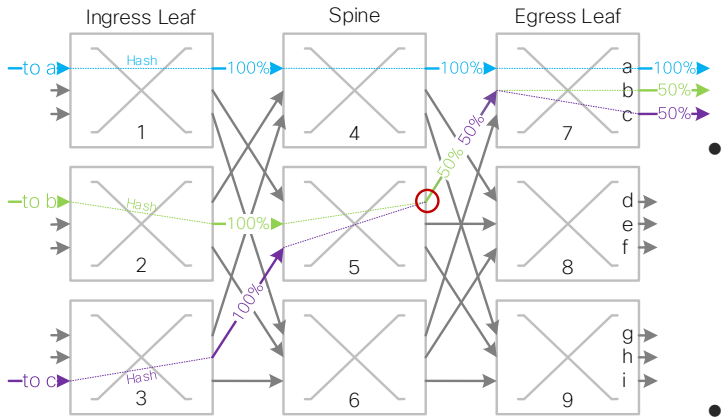
- Ingress Leaf (2)
  - Receives traffic on an input port
  - Looks up the destination of the packet to identify eligible ports
    - All leaf to spine ports can reach egress port (B)
  - Selects an output port with a hash on info in the packet
    - example; {Src IP, Src Port, Dest IP, Dest Port, Protocol}
    - In this case it selects the link to Spine 5
- Spine (5)
  - Looks up the destination of the packet to identify eligible ports
  - Only one port can reach egress port (B)
- Egress Leaf (7)
  - Looks up the destination of the packet to identify eligible ports
  - Egress port (B) is directly connected

- 100% of the traffic passes through the network
  - To-A is 100%
  - To-B is 100%

AI/ML Load Balancing Options

Example : Ethernet (Good) – 3 Flows

Good | Ethernet
Better | Ethernet with Telemetry
Best | Scheduled Fabric



Congestion on Spine 5 to Egress Leaf 7 link
means traffic to port b and port c are impacted

- Ingress Leaf (3)
  - Receives traffic on an input port
  - Looks up the destination of the packet to identify eligible ports
    - All leaf to spine ports can reach egress port (C)
  - Selects an output port with a hash on info in the packet
    - example; {Src IP, Src Port, Dest IP, Dest Port, Protocol}
    - In this case it selects the link to Spine 5
- Spine (5)
  - Looks up the destination of the packet to identify eligible ports
  - Only one port can reach egress port (C)
  - Port is already sending line-rate traffic to egress port (B)
  - Can only send 50% towards egress port(B) and 50% to egress port (C)
- Egress Leaf (7)
  - Looks up the destination of the packet to identify eligible ports
  - Egress port (C) is directly connected

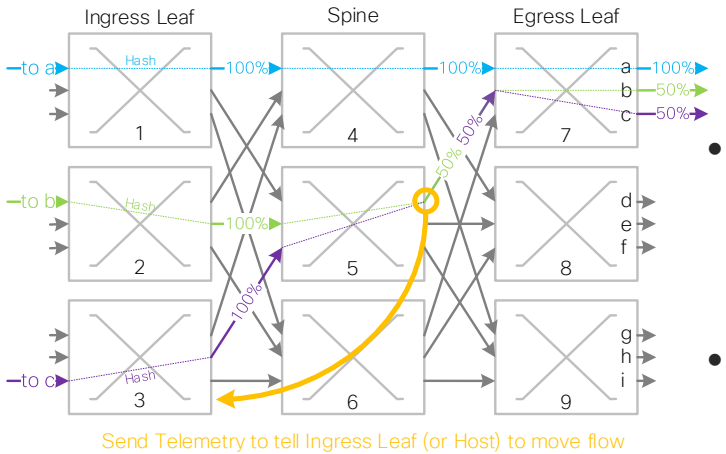
- 100% of the traffic passes through the network
  - To-A is 100%
  - To-B is 50%
  - To-C is 50%

Performance depends on flow characteristics

AI/ML Load Balancing Options

Example : Ethernet with Telemetry (Better) – 3 Flows

Good Ethernet
Better Ethernet with Telemetry
Best Scheduled Fabric

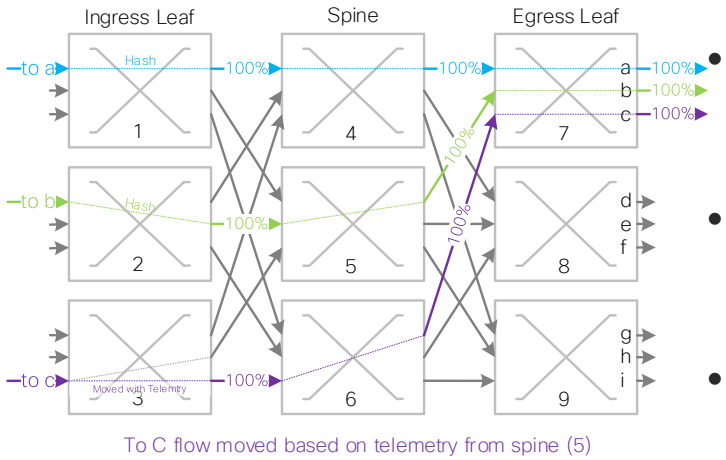


- Ingress Leaf (3)
  - Receives traffic on an input port
  - Looks up the destination of the packet to identify eligible ports
    - All leaf to spine ports can reach egress port (C)
  - Selects an output port with a hash on info in the packet
    - example; {Src IP, Src Port, Dest IP, Dest Port, Protocol}
    - In this case it selects the link to Spine 5
- Spine (5)
  - Looks up the destination of the packet to identify eligible ports
  - Only one port can reach egress port (C)
  - Detects Congestion and sends telemetry to ingress Leaf (3) or Host
- Ingress Leaf (3)
  - Receives information from Spine (5) that the path to Egress Leaf (7) is congested

AI/ML Load Balancing Options

Example : Ethernet with Telemetry (Better) – 3 Flows

Good | Ethernet
Better | Ethernet with Telemetry
Best | Scheduled Fabric



To C flow moved based on telemetry from spine (5)

- Ingress Leaf (3)
  - Receives traffic on an input port
  - Looks up the destination of the packet to identify eligible ports
    - All leaf to spine ports can reach egress port (C)
  - Based on telemetry received from Spine 5, selects link to Spine6 for this flow
- Spine (6)
  - Looks up the destination of the packet to identify eligible ports
  - Only one port can reach egress port (C)
- Egress Leaf (7)
  - Looks up the destination of the packet to identify eligible ports
  - Egress port (C) is directly connected
- 100% of the traffic passes through the network
  - To-A is 100%
  - To-B is 100%
  - To-C is 100%

Telemetry can improve Ethernet performance

Effectiveness depends on flow characteristics, buffer sizing, response times, state size, …

Reminder: AI has Few synchronous high BW flows ; This helps…

In more complex examples…

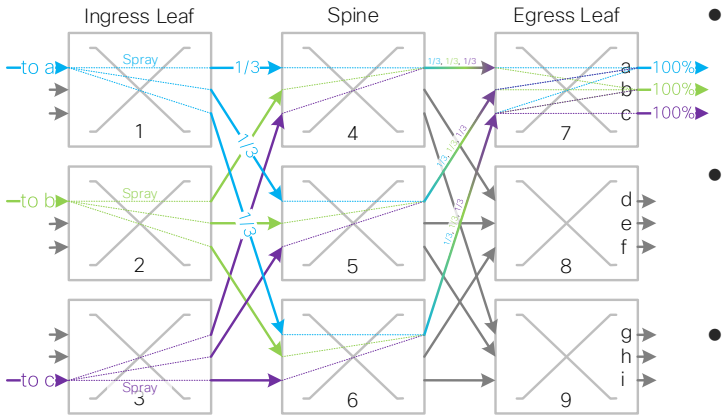
Re-balancing may cause other issues

Making globally impacting decisions based on local information

AI/ML Load Balancing Options

Example : Fully Scheduled (Best) – 3 Flows

Good	Ethernet
Better	Ethernet with Telemetry
Best	Scheduled Fabric



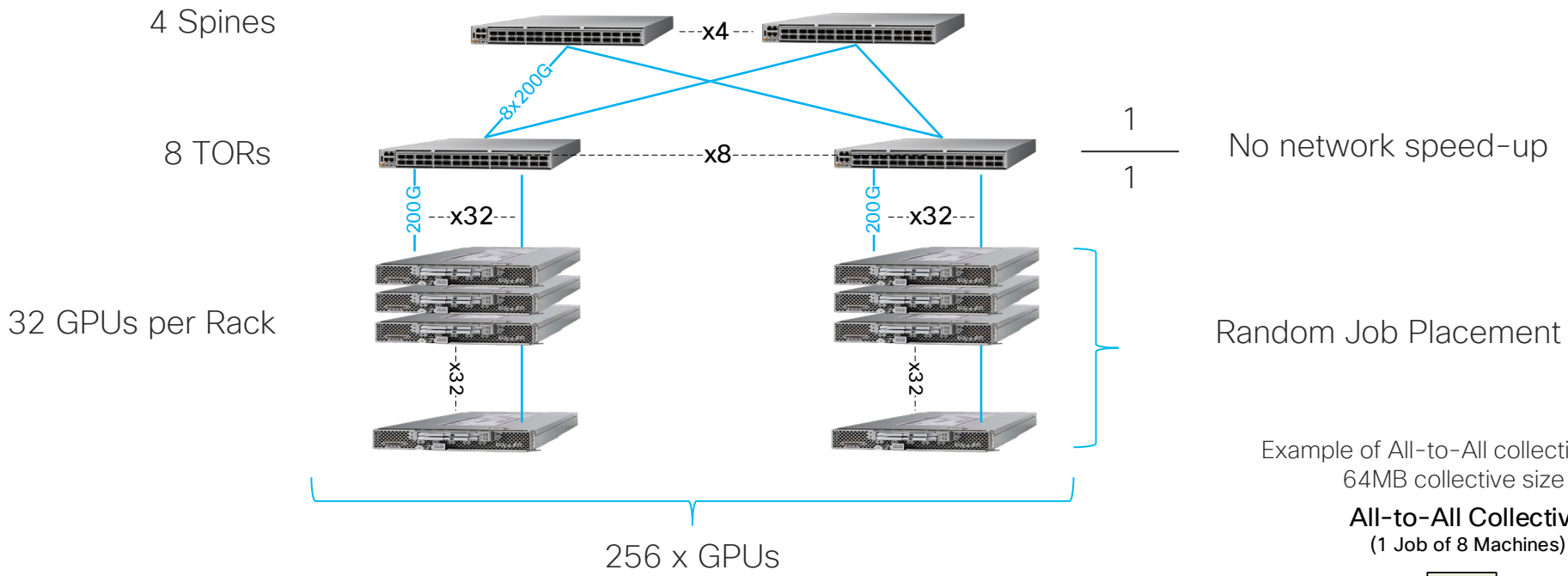
- Ingress Leafs
  - Receives traffic on an input port
  - Sprays packets across all eligible spine ports
    - In this example; 1/3 BW per link
- Spines
  - Sprays packets across all eligible ports towards egress leaf
  - Only one port can reach egress ports (A, B, C)
- Egress Leaf (7)
  - Looks up the destination of the packet to identify eligible ports
  - Egress ports (A, B, C) are directly connected
- 100% of the traffic passes through the network
  - To-A is 100%
  - To-B is 100%
  - To-C is 100%

Flow independent performance, all available links used

All available bandwidth is used, regardless flow

AI/ML Workload Study

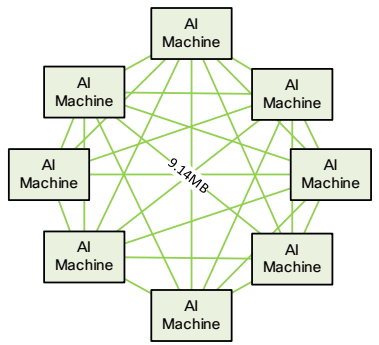
Basic Topology & Methodology



- Run All-to-All Collective traffic pattern
  - Each machine sends to all other machines in the job
  - Traffic is interleaved to all destinations in round robin fashion
  - Transmission starts at the same time

- Measure Job Completion Time (JCT)
  - Job is complete after all data is received for the Job
  - JCT is based on slowest path

Example of All-to-All collective with 64MB collective size



Total Data Transfer 8*(7*9.14MB)=512MB

AI/ML Workload Study Test – 1

Increasing the number of simultaneous jobs

How does ECMP and a Fully scheduled fabric perform as we increase the **number of jobs** running on the cluster?

vs

Normalized to

1mSec

ECMP
(Hash to select link)

Fully Scheduled
(Ingress VOQ, Req-Grant,
Spray across all links and re-order at Egress)
*[Note: InfiniBand is **not** a fully scheduled fabric]*

Ideal
(Transmission time of bits over 200GE,
No fiber latency, no switch latency
Impossible to achieve)

1 Job with
256 Machines

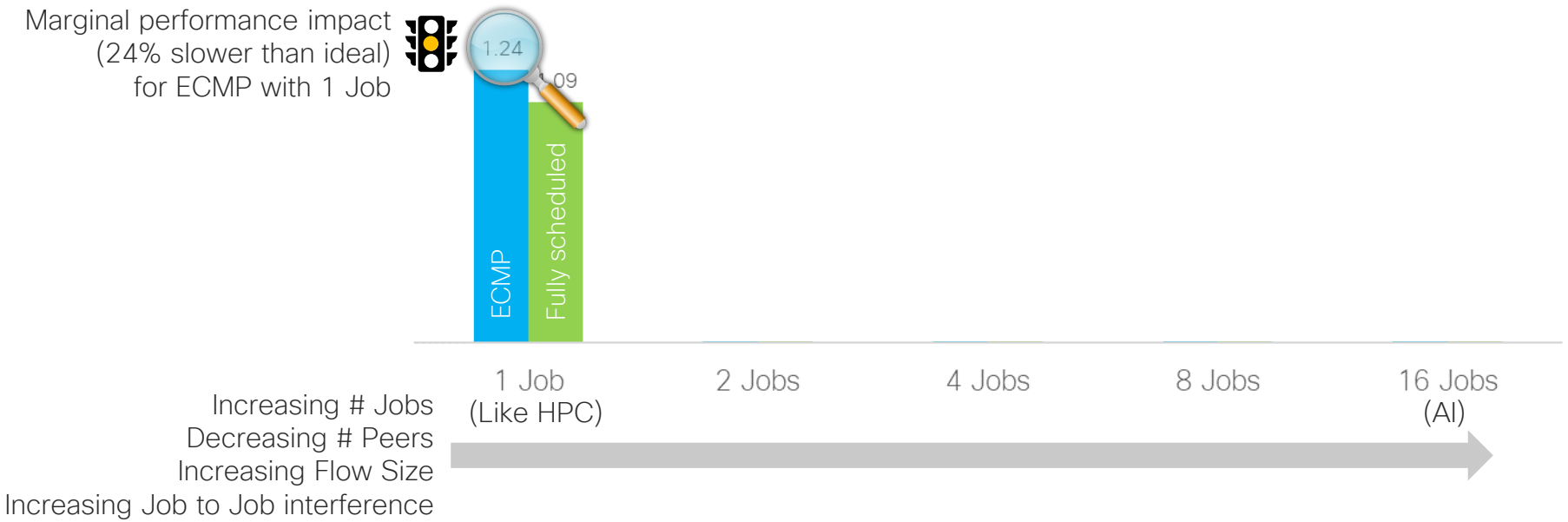
4 Jobs each with
64 Machines

16 Jobs each with
16 Machines

AI/ML Workload Study Test – 1

Increasing the number of simultaneous jobs

All 256 Machines Active
Normalized to Ideal JCT of 1mSec

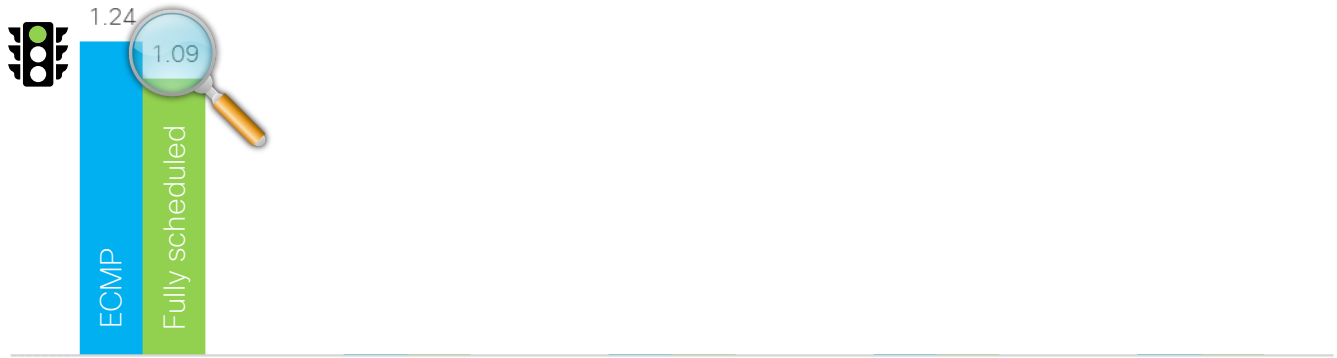


AI/ML Workload Study Test – 1

Increasing the number of simultaneous jobs

All 256 Machines Active
Normalized to Ideal JCT of 1mSec

Near perfect performance with full scheduling and ideal load balancing

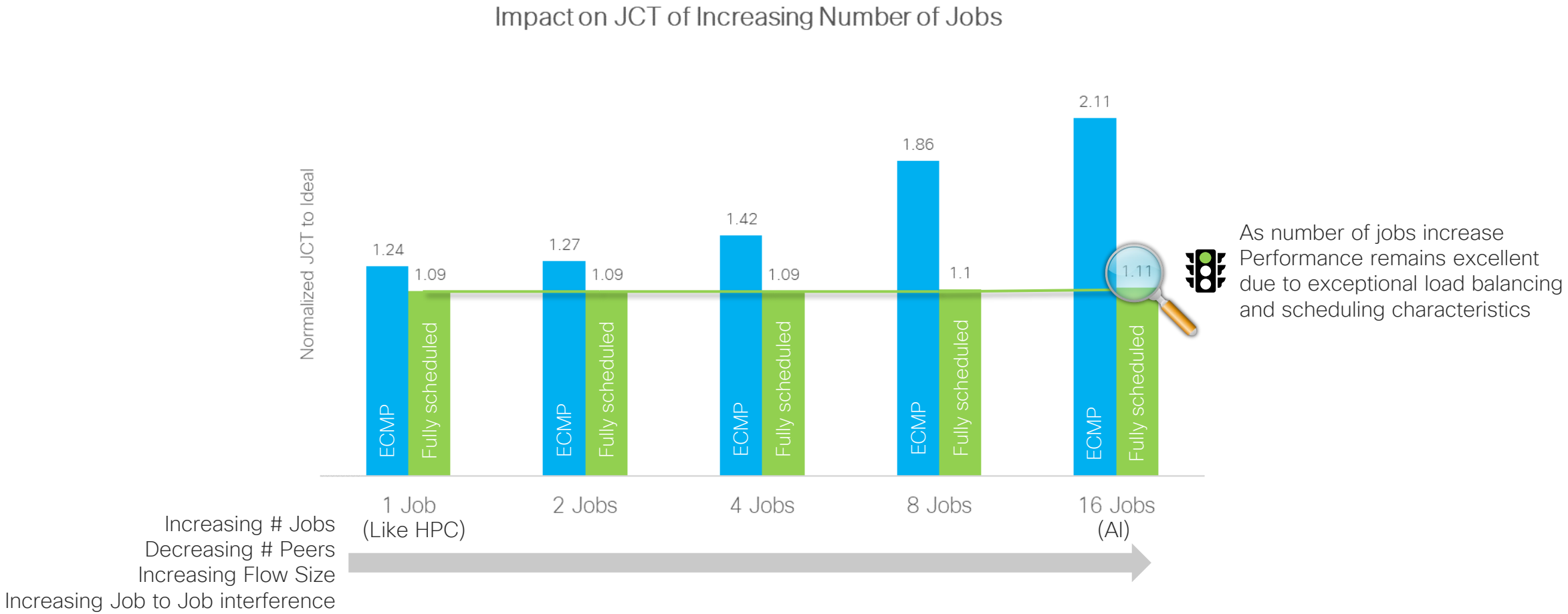


Increasing # Jobs
Decreasing # Peers
Increasing Flow Size
Increasing Job to Job interference

AI/ML Workload Study Test – 1

Increasing the number of simultaneous jobs

All 256 Machines Active
Normalized to Ideal JCT of 1mSec

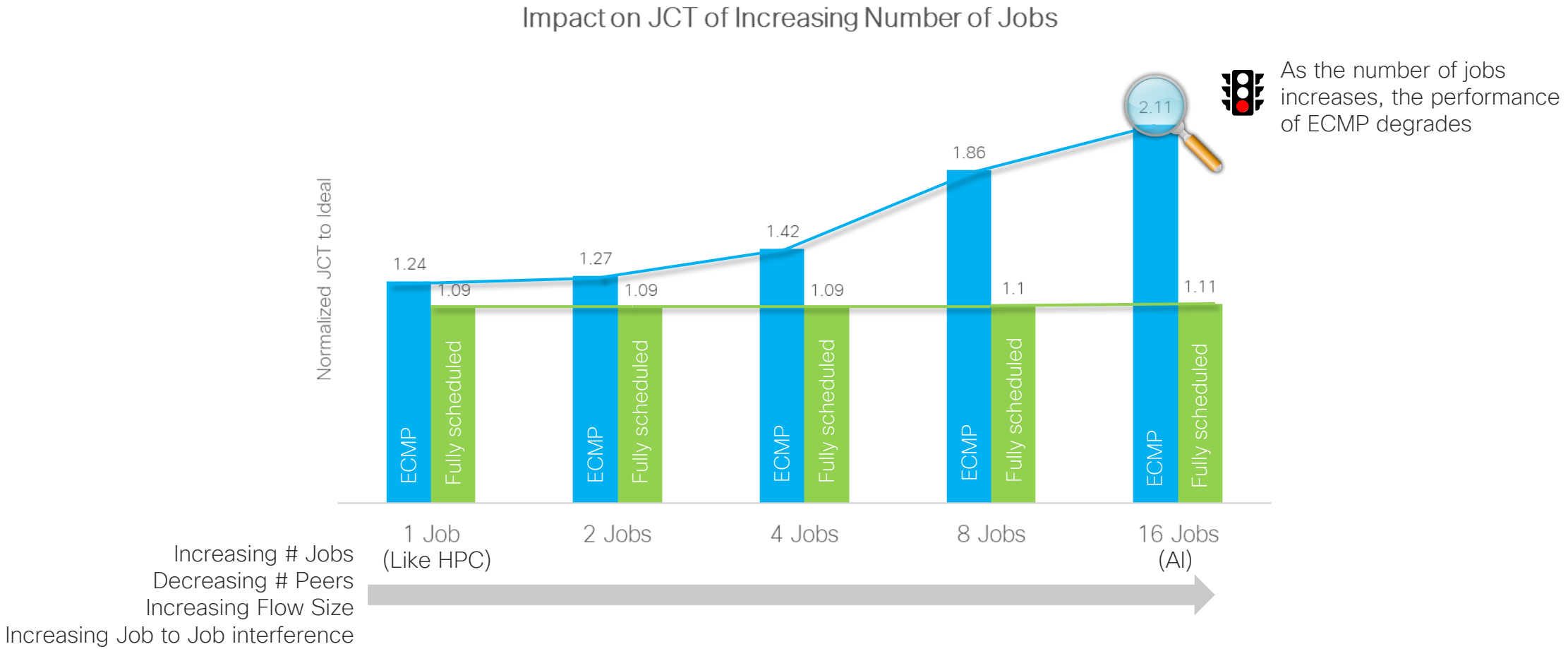


As number of jobs increase
Performance remains excellent
due to exceptional load balancing
and scheduling characteristics

AI/ML Workload Study Test – 1

Increasing the number of simultaneous jobs

All 256 Machines Active
Normalized to Ideal JCT of 1mSec

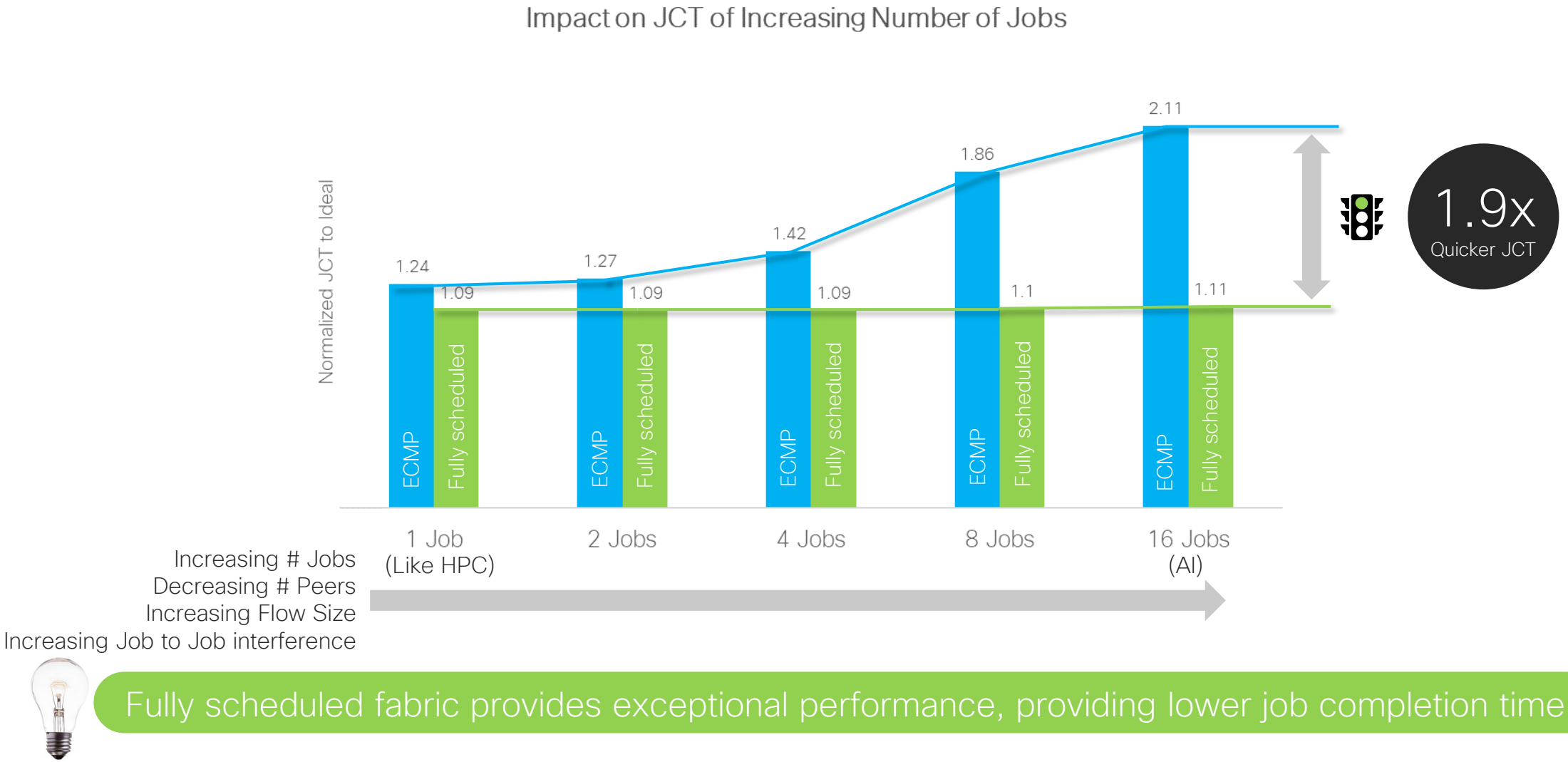


As the number of jobs increases, the performance of ECMP degrades

AI/ML Workload Study Results – 1

Conclusion: Scheduled fabric enables 1.9x better JCT

All 256 Machines Active
Normalized to Ideal JCT of 1mSec

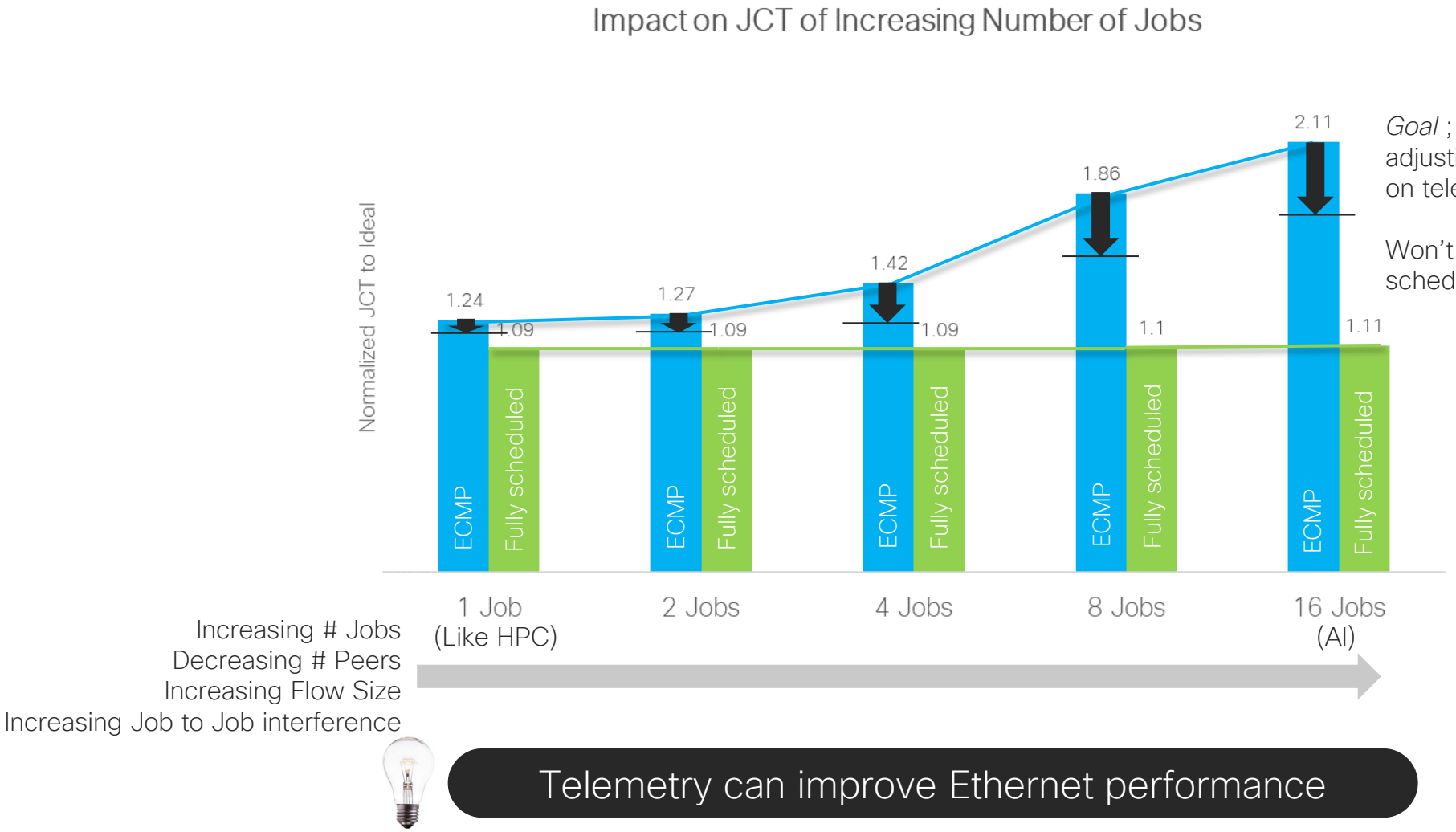


Fully scheduled fabric provides exceptional performance, providing lower job completion time

AI/ML Workload Study Results – 1

Conclusion: Use Telemetry to improve Ethernet Performance

All 256 Machines Active
Normalized to Ideal JCT of 1mSec



Goal ; Improve Ethernet performance by adjusting load balancing decisions based on telemetry information

Won't achieve same performance as fully scheduled fabric, but can help

AI/ML Workload Study Test - 2

Add Network Speed-up

How much **network speed-up** do we need to add to improve ECMP performance

vs

Normalized to

1mSec

ECMP
(Hash to select link)

Fully Scheduled
(Ingress VOQ, Req-Grant,
Spray across all links and re-order at Egress)
*[Note: InfiniBand is **not** a fully scheduled fabric]*

Ideal
(Transmission time of bits over 200GE,
No fiber latency, no switch latency
Impossible to achieve)

No Speedup
(Baseline)

1.33x Speedup
(25% AI Machines Idle)

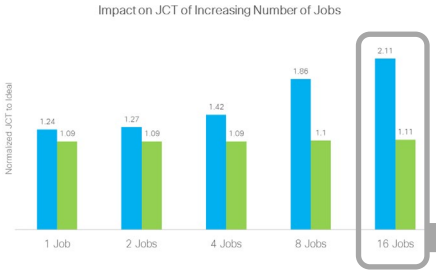
2x Speedup
(50% AI Machine Idle)

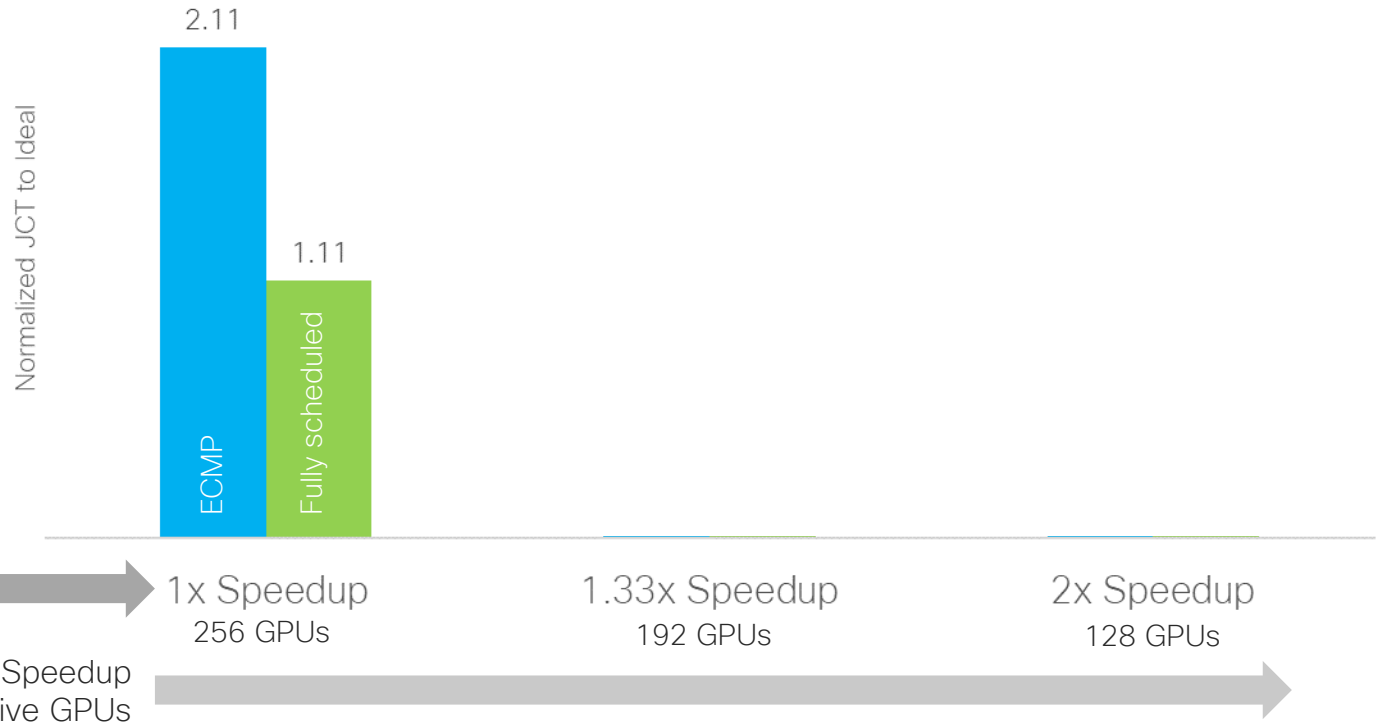
AI/ML Workload Study Test - 2

Add Network Speed-up

All jobs have 16 machines
Normalized to Ideal JCT of 1mSec

Test1 Results
16 Jobs each on 16 Machines

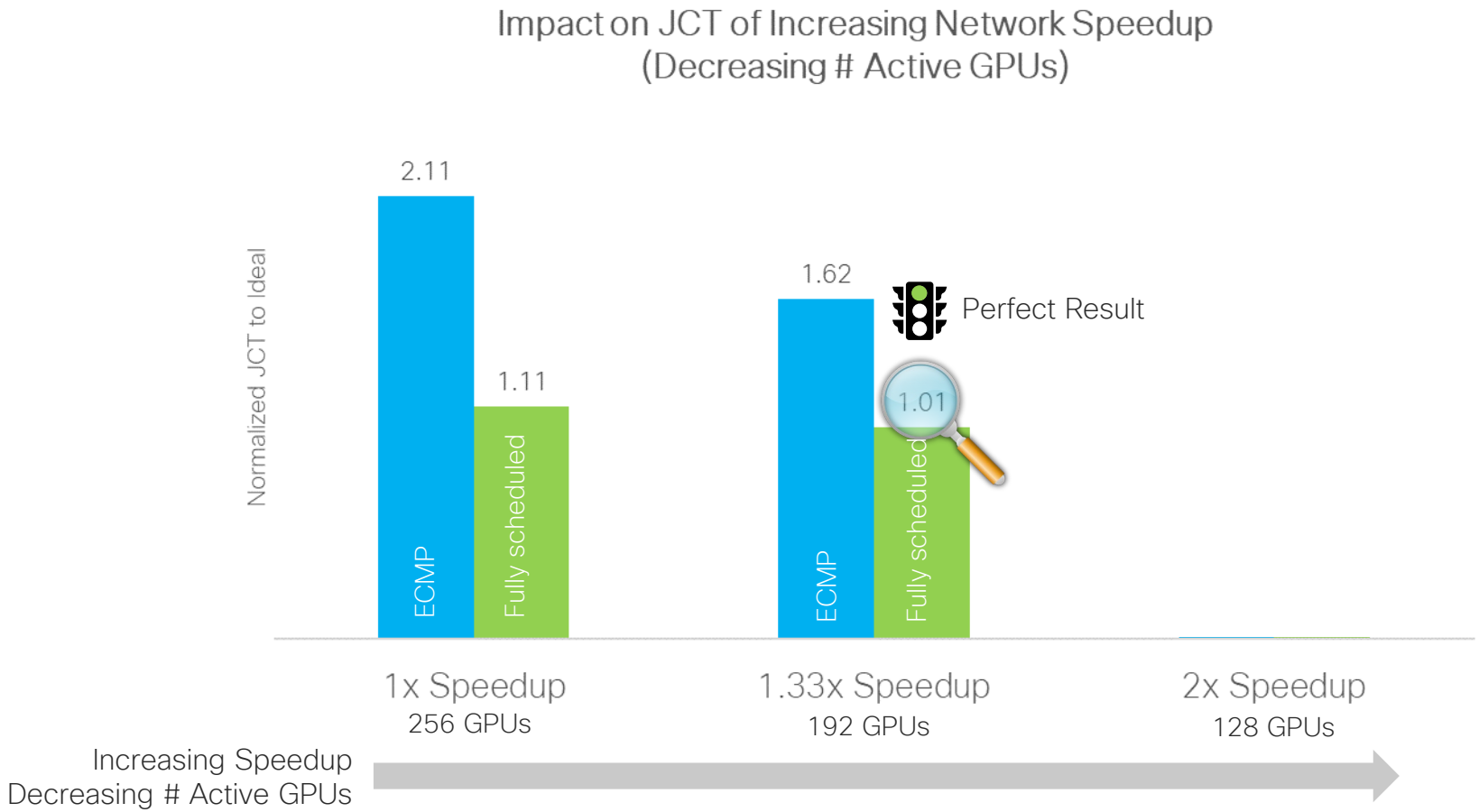




AI/ML Workload Study Test - 2

Add Network Speed-up

All jobs have 16 machines
Normalized to Ideal JCT of 1mSec

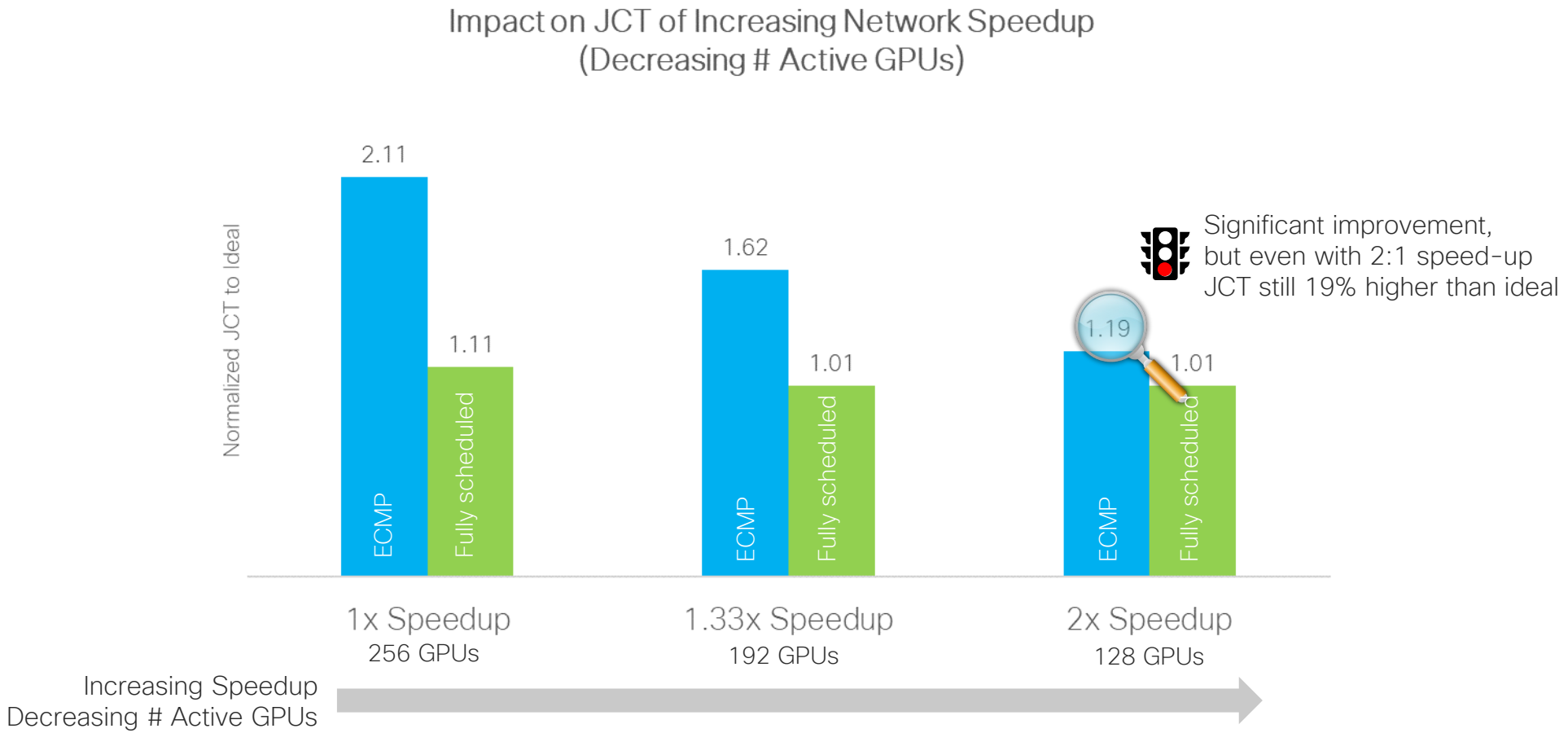


Note : Fully scheduled fabric would achieve perfect results with less than 1.33x speed-up.
Using 1.33x to simplify the take-aways

AI/ML Workload Study Test – 2

Add Network Speed-up

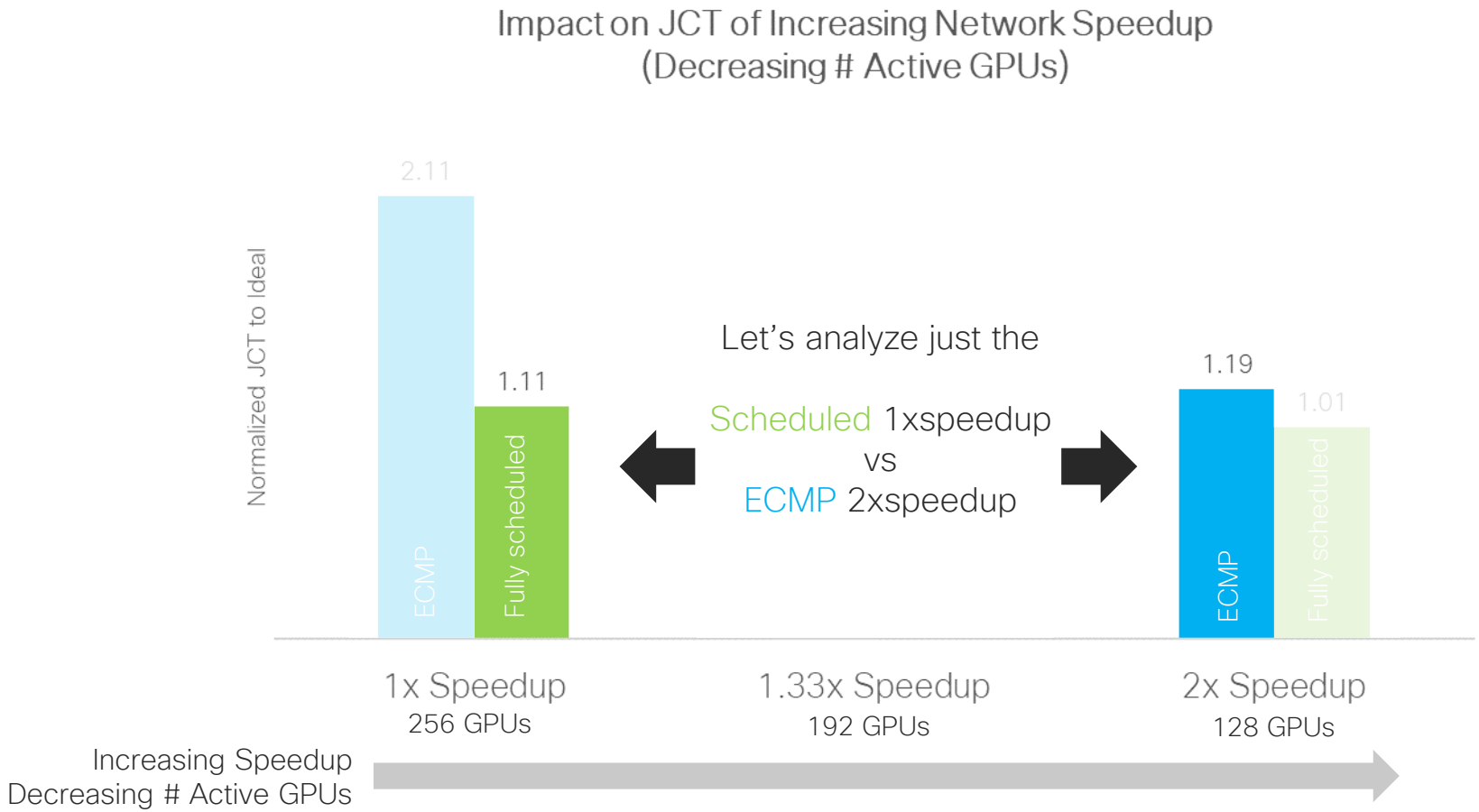
All jobs have 16 machines
Normalized to Ideal JCT of 1mSec



AI/ML Workload Study Test – 2

Add Network Speed-up

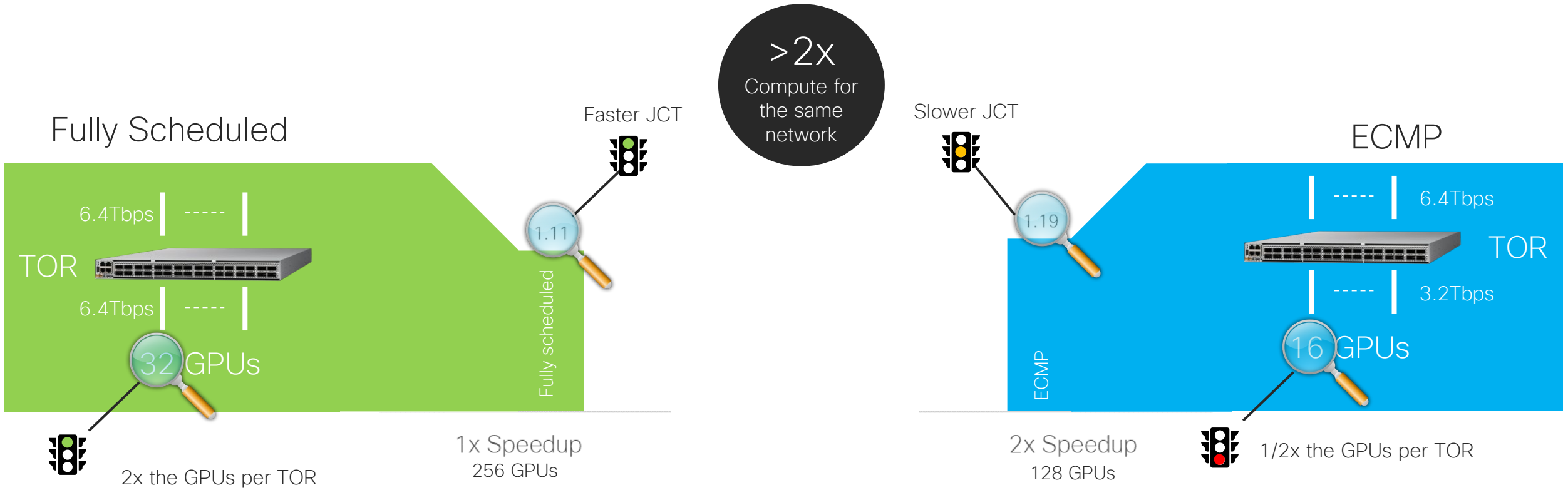
All jobs have 16 machines
Normalized to Ideal JCT of 1mSec



AI/ML Workload Study Test - 2

Add Network Speed-up

All jobs have 16 machines
Normalized to Ideal JCT of 1mSec



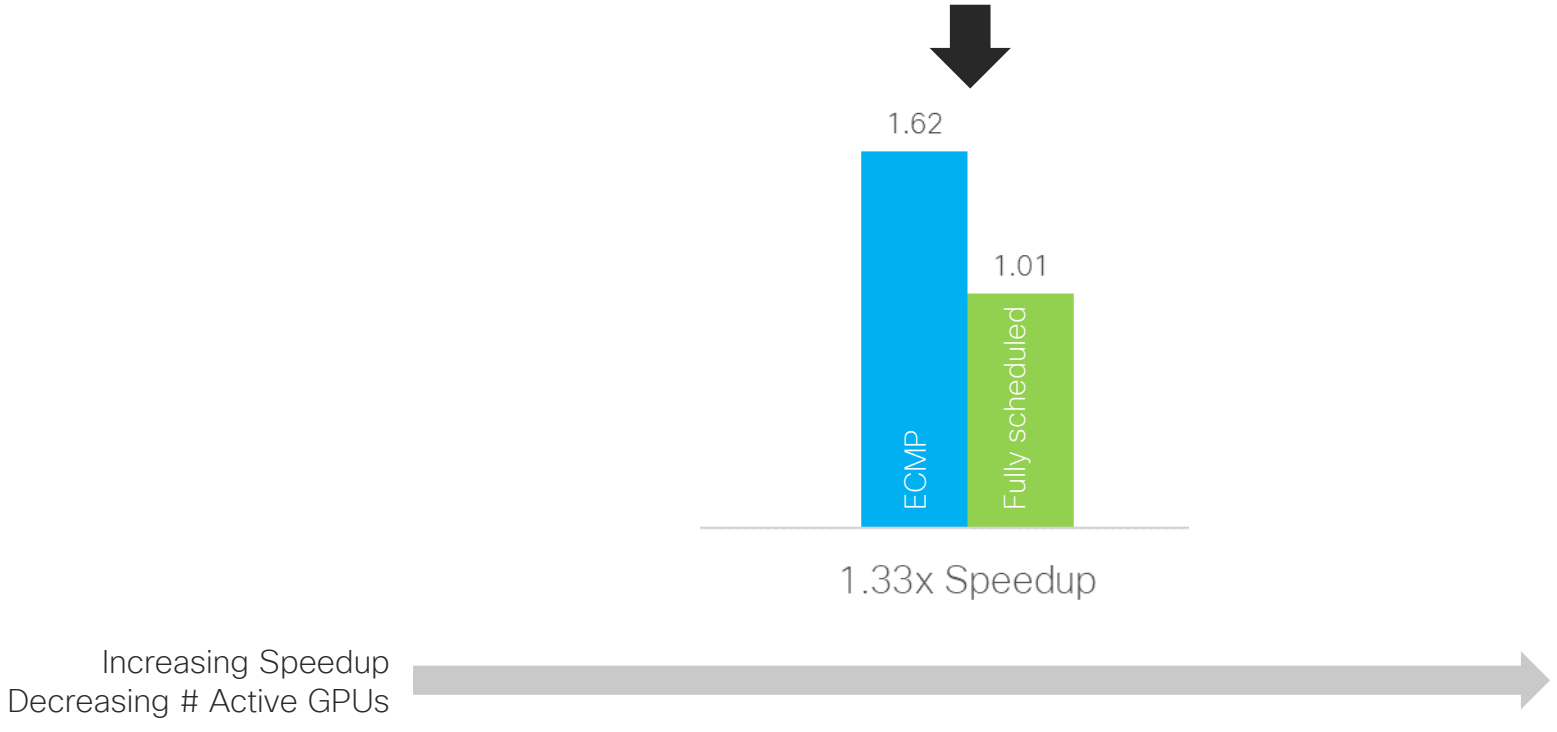
Fully scheduled fabric provides exceptional performance, even with 2x higher load

AI/ML Workload Study Test – 2

Add Network Speed-up

All jobs have 16 machines
Normalized to Ideal JCT of 1mSec

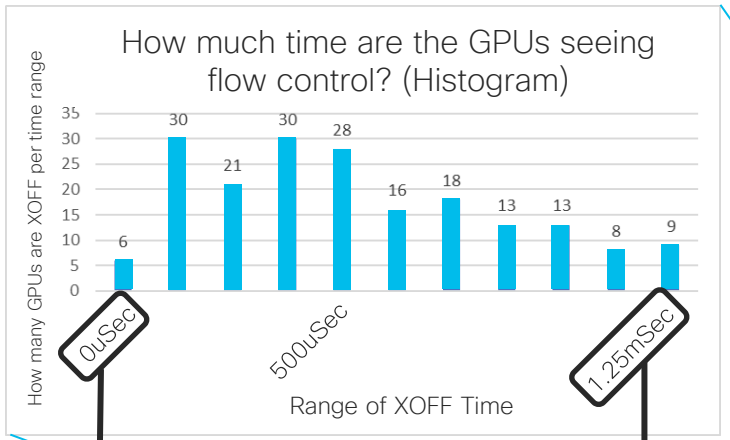
Let's analyze why the JCT is higher for
ECMP vs Scheduled fabric



AI/ML Workload Study Test - 2

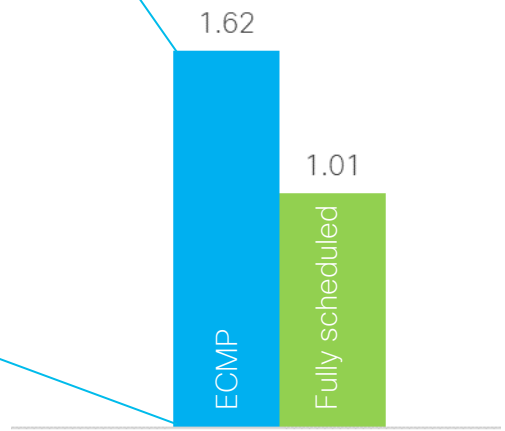
Add Network Speed-up

All jobs have 16 machines
Normalized to Ideal JCT of 1mSec



6/192
3% GPUs never Received an XOFF

9/192
5% GPUs were XOFF 1.25mSec during the test



1.33x Speedup

Because of synchronization
5% of the GPUs being slow stall the other 95%

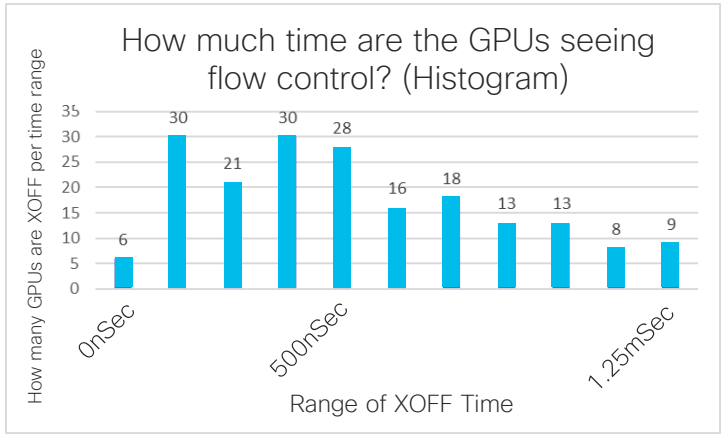
Increasing Speedup
Decreasing # Active GPUs

Any-to-Any collective is a non-blocking traffic pattern. XOFFs are from bad network load balancing choices

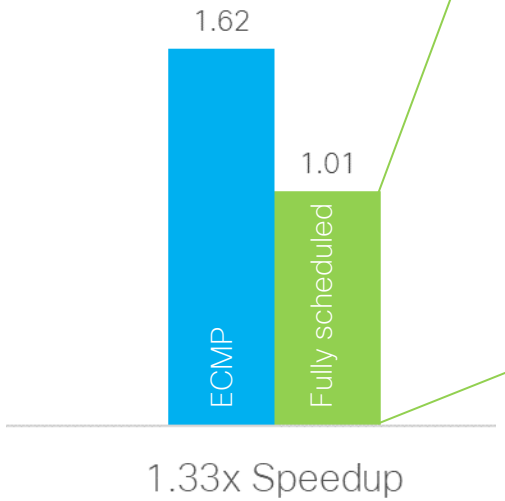
AI/ML Workload Study Test - 2

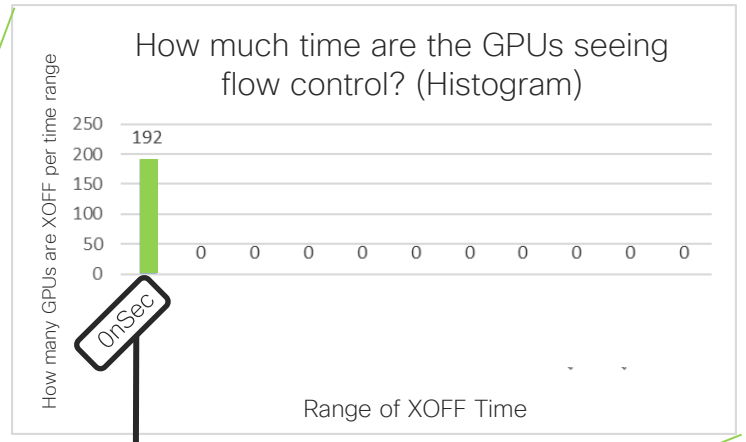
Add Network Speed-up

All jobs have 16 machines
Normalized to Ideal JCT of 1mSec



Job completion time (JCT) slowed down by 1.25mSec due to tail latency for the slowest flow





Scheduled fabric **never** slowed down the GPUs

100%
Non-Blocking

Increasing Speedup
Decreasing # Active GPUs

Fully scheduled fabric provides fully non-blocking performance and ideal JCT

Solution Summary

	InfiniBand	Ethernet ECMP for load balancing	Telemetry assisted Ethernet Use Telemetry to improve load balancing	Fully scheduled Spray & Re-order
Pace of BW Increase How quickly does the technology evolve?				
Port Bandwidth What is the maximum port speed?				
System Radix How many ports per system?				
Network Cost How much does the network cost for a certain size?				
Single Job Performance How does the network perform with 1 job?				
Multi-Job Performance How does the network perform with many jobs? Driven by load balancing characteristics				
Multi-Vendor Support How many vendors support the technology				
Support for customer-built AI Machines Can the technology support other GPU types?				

Solution Summary

	InfiniBand	Ethernet ECMP for load balancing	Telemetry assisted Ethernet Use Telemetry to improve load balancing	Fully scheduled Spray & Re-order
Pace of BW Increase How quickly does the technology evolve?				
Port Bandwidth What is the maximum port speed?				
System Radix How many ports per system?				
Network Cost How much does the network cost for a certain size?				
Single Job Performance How does the network perform with 1 job?				
Multi-Job Performance How does the network perform with many jobs? Driven by load balancing characteristics				
Multi-Vendor Support How many vendors support the technology				
Support for customer-built AI Machines Can the technology support other GPU types?				

Solution Summary

	InfiniBand	Ethernet ECMP for load balancing	Telemetry assisted Ethernet Use Telemetry to improve load balancing	Fully scheduled Spray & Re-order
Pace of BW Increase How quickly does the technology evolve?				
Port Bandwidth What is the maximum port speed?				
System Radix How many ports per system?				
Network Cost How much does the network cost for a certain size?				
Single Job Performance How does the network perform with 1 job?				
Multi-Job Performance How does the network perform with many jobs? Driven by load balancing characteristics				
Multi-Vendor Support How many vendors support the technology				
Support for customer-built AI Machines Can the technology support other GPU types?				

Use telemetry to improve load balancing decisions

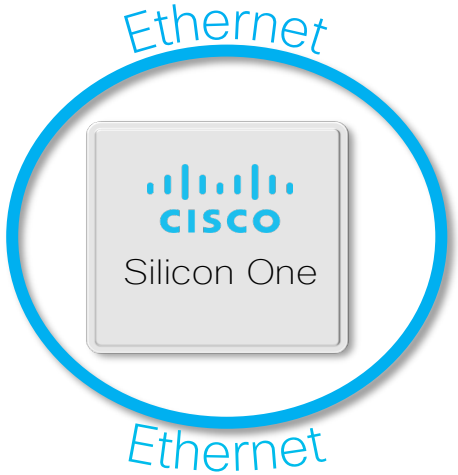
Solution Summary

	InfiniBand	Ethernet ECMP for load balancing	Telemetry assisted Ethernet Use Telemetry to improve load balancing	Fully scheduled Spray & Re-order
Pace of BW Increase How quickly does the technology evolve?				Vendor implementation dependent
Port Bandwidth What is the maximum port speed?				
System Radix How many ports per system?				Vendor implementation dependent
Network Cost How much does the network cost for a certain size?				Vendor implementation dependent
Single Job Performance How does the network perform with 1 job?				
Multi-Job Performance How does the network perform with many jobs? Driven by load balancing characteristics			Use telemetry to improve load balancing decisions	
Multi-Vendor Support How many vendors support the technology				Multiple vendors build spray & re-order fabrics
Support for customer-built AI Machines Can the technology support other GPU types?				

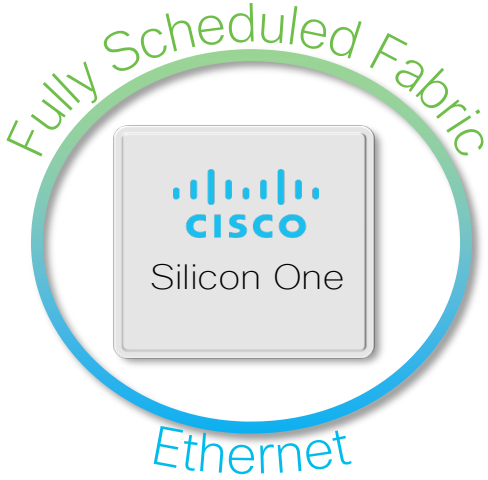
Cisco Silicon One

One Architecture, Three Modes

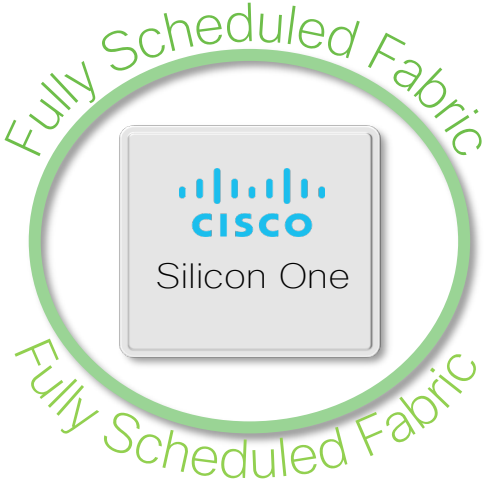
Standalone Mode



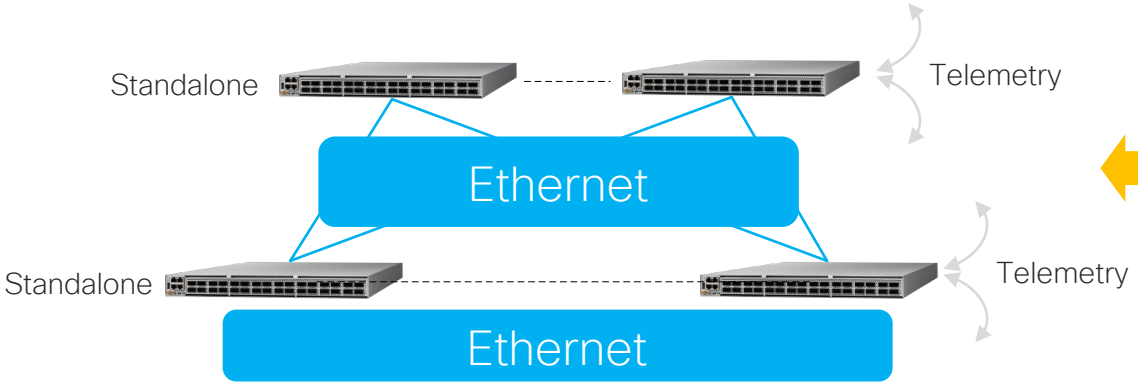
Linecard Mode



Fabric Element Mode

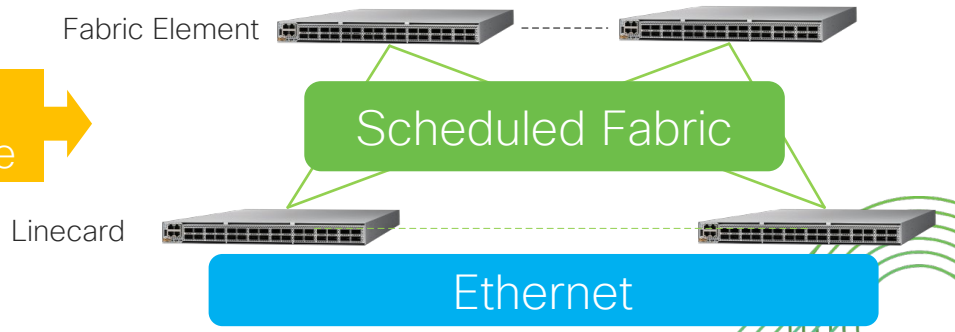


Ethernet ECMP



Same Hardware

Fully Scheduled Fabric



Cisco Silicon One

One Architecture for AI/ML

Silicon One

	InfiniBand	Ethernet ECMP for load balancing	Telemetry assisted Ethernet Use Telemetry to improve load balancing	Fully scheduled Spray & Re-order
Pace of BW Increase How quickly does the technology evolve?				
Port Bandwidth What is the maximum port speed?				
System Radix How many ports per system?				
Network Cost How much does the network cost for a certain size?				
Single Job Performance How does the network perform with 1 job?				
Multi-Job Performance How does the network perform with many jobs? Driven by load balancing characteristics				
Multi-Vendor Support How many vendors support the technology				
Support for customer-built AI Machines Can the technology support other GPU types?				

Deploy a Cisco Silicon One Network now. **Evolve** from Ethernet to Fully scheduled as needed

Conclusions & Key Messages

The past doesn't dictate the future

The network is fundamental for AI/ML due to unique traffic characteristics

Data Center

Many small asynchronous flows
Small penalty for wrong path selection

Compute bound

AI/ML

Few large synchronous flows
Large penalty for wrong path selection based on synchronization of algorithms

Synchronization stalls compute
Mostly Network Bound

! InfiniBand is not the correct interconnect for AI/ML

Slow pace of innovation, proprietary interconnect, expensive
Optimized for 1 Job results in poor performance for AI/ML workloads

Conclusions & Key Messages

Evolve with Cisco Silicon One

Which customers should deploy Ethernet AI/ML Networks?

Customer's who want to …

- Enjoy the heavy investment, open standards & favorable cost-bandwidth dynamics of Ethernet
- Use telemetry to improve the performance of Ethernet
- Willing to minimize network load by restricting the placement of AI jobs

Which customers should deploy Fully Scheduled AI/ML Networks?

Customer's who want to …

- Enjoy the performance benefits of an Ingress VOQ, fully scheduled, spray and re-order fabric
- Maximize AI/ML compute performance [1.9x better JCT]
- Optimize the power and cost efficiency of their network [2x more compute for the same network]

What is special about Cisco Silicon One for AI/ML Networks?

Only Cisco Silicon One…

- Provides the same innovation pace and cost-bandwidth dynamics as Ethernet, but with Scheduled fabric
- Has **one** architecture which can evolve from Ethernet to Scheduled fabric
- Uses P4 programming to evolve telemetry semantics over time, adding features and ensuring interop across vendors
- Provides excellent burst absorption with a fully shared and unified packet buffer

Cisco Silicon One

Evolve your network

One Architecture for AI/ML

Standard Ethernet	Telemetry Assisted Ethernet	Fully Scheduled
Maximize Interoperability	Middle Ground	Maximize Performance


CISCO