

# From Switch-based to Switch-less Interconnection Network

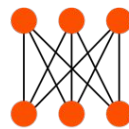
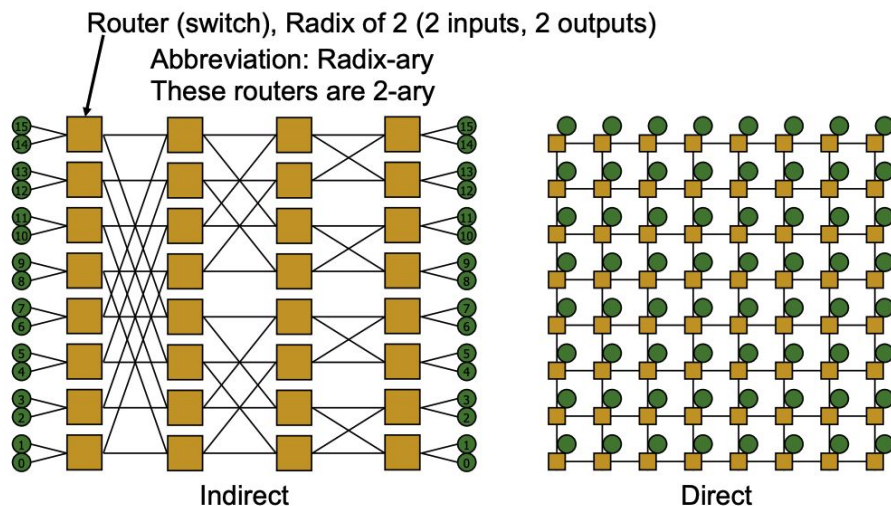
Eric Ding, 4/28

# Overview

- ❑ A brief overview of the interconnection network
- ❑ Dragonfly topology
- ❑ Switch-less Dragonfly design
- ❑ Evaluation
- ❑ Discussion

# Interconnection Networks

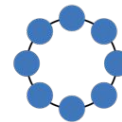
- ❑ **Topology: the way switches and nodes are wired**
- ❑ Routing
- ❑ Buffering and flow control



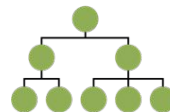
(a) Crossbar



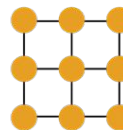
(b) Star



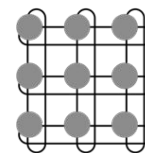
(c) Ring



(d) Tree



(e) 2D Mesh



(f) 2D Torus

Terminology: diameter, bisection bandwidth, injection bandwidth, blocking & non-blocking, radix,...

Metrics: cost, latency, contention, energy, bandwidth,...

# Interconnection Networks

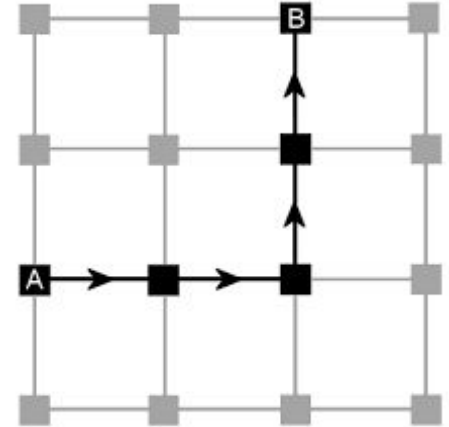
- ❑ Topology
- ❑ **Routing: how does a message get from source to destination**
- ❑ Buffering and flow control

## Mechanism

1. Arithmetic. Eg. dimension order routing
2. Source based. Route determined by source.
3. Table lookup based. Route determined along transmission

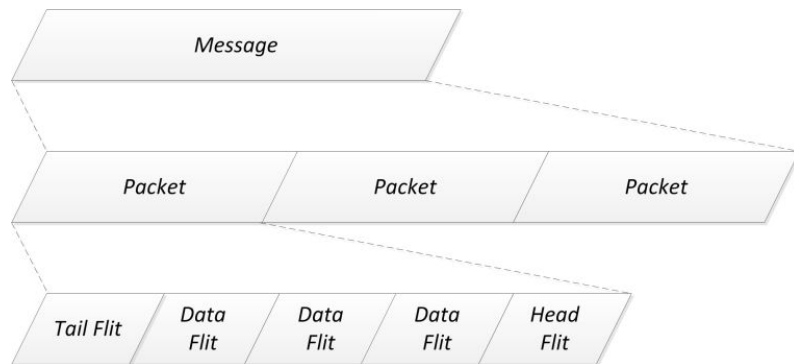
## Algorithm

1. Deterministic
2. Oblivious
3. Adaptive



# Interconnection Networks

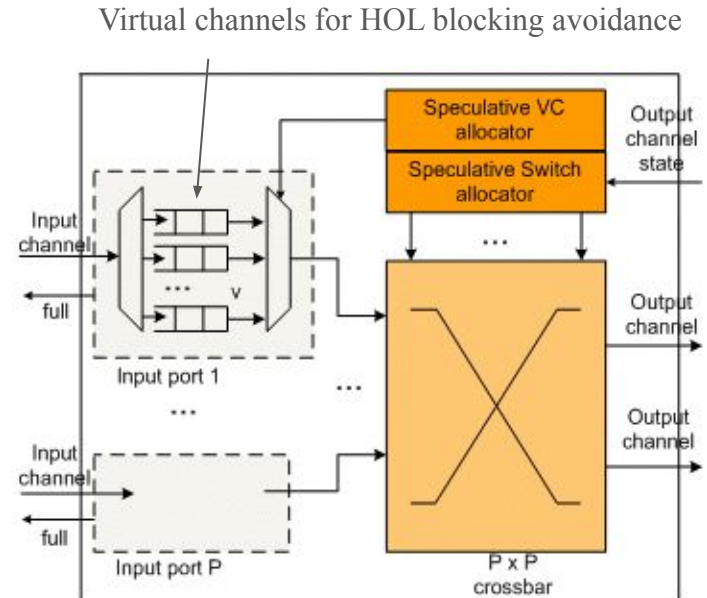
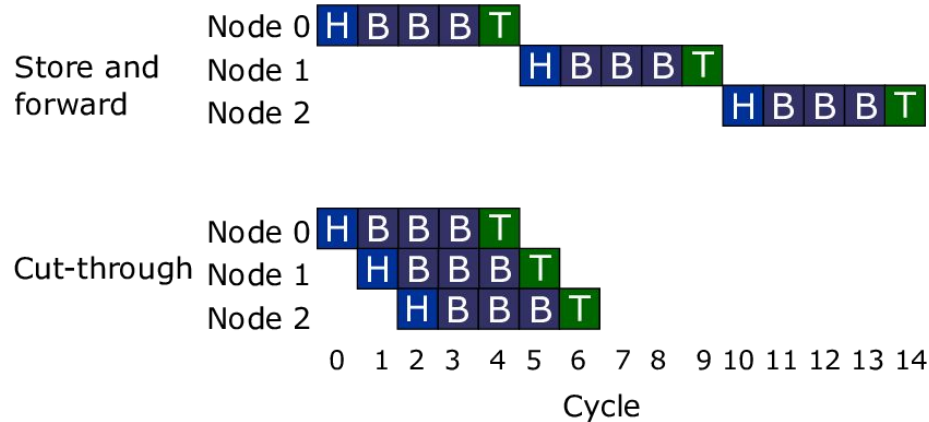
- ❑ Topology: the way switches and nodes are wired
- ❑ Routing: how does a message get from source to destination
- ❑ **Buffering and flow control**
  - ❑ Packet: a message is broken into multiple packets
  - ❑ Flit: a packet may itself be broken into flits
    - ❑ Flits do not contain additional headers
    - ❑ Flits are ordered and follow the same path



Interconnection network performance of multi-core cluster architectures. Journal of Computers. 2015

# Interconnection Networks

- ❑ Topology: the way switches and nodes are wired
- ❑ Routing: how does a message get from source to destination
- ❑ **Buffering and flow control**



# HPC Network: Dragonfly

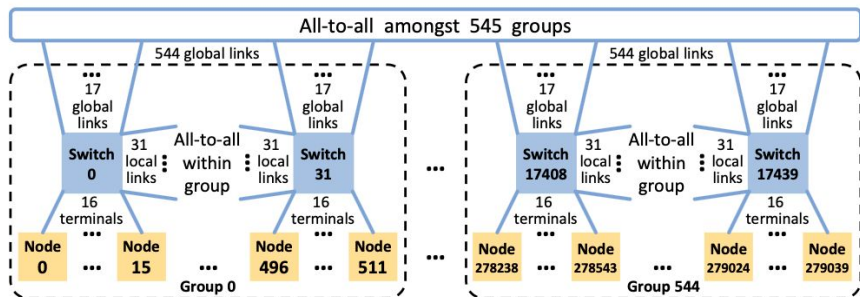
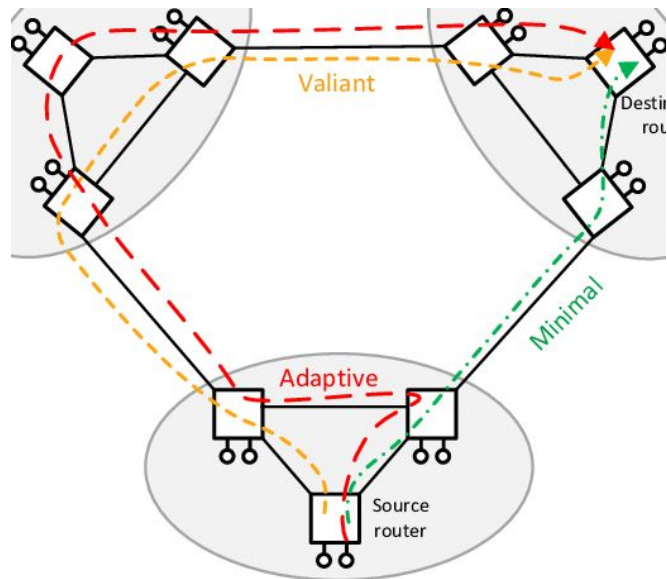


Fig. 2. The Dragonfly-based Slingshot topology. Switches are fully connected within groups, and groups are also all-to-all connected.

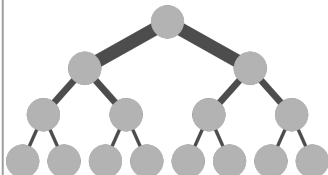
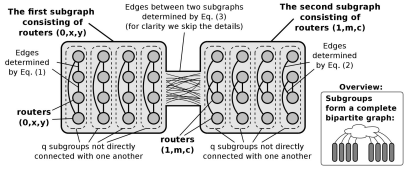
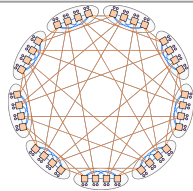
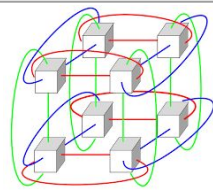


Efficient Routing Mechanisms for Dragonfly Networks. ICPP 2013

A network topology designed for large scale compute clusters

Examples: Frontier, Aurora, LUMI, Perlmutter

# HPC Network

	Fat-Tree	Slimfly	Dragonfly	3D Torus
Description				
Diameter	$2 \log_2(N)$	2	3, 4	$3 (N^{1/3})/2$
Bisection bandwidth	$n/2$ , maximal	Near maximal	Very high	$2 N^{2/3}$
Pros	Full bandwidth	Optical latency, fewer switches and cables	Scale well, fewer switches and cables	Simple cabling, simple routing
Cons	Expensive, high radix requirement	Difficult cabling	Require smart routing, congestion	Poor for global communication, does not scale well



# Interconnection Networks

Packet: a message is broken into multiple packets

Flit: a packet may itself be broken into flits

- ❑ Flits do not contain additional headers
- ❑ Flits are ordered and follow the same path

For a flit to jump to the next router, it must acquire three resources:

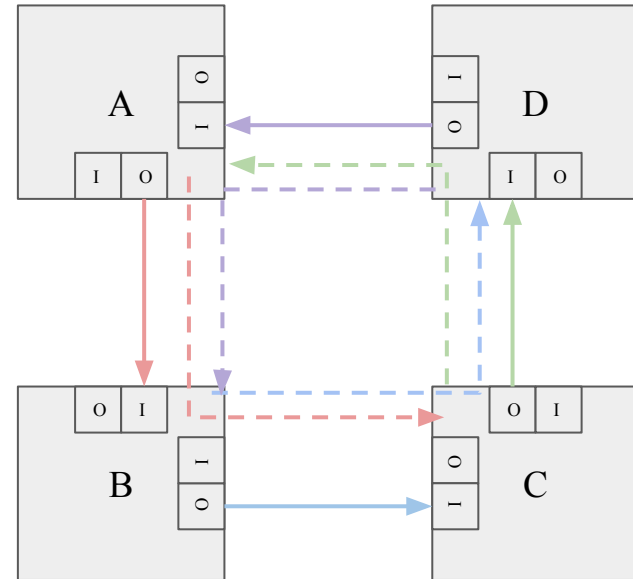
1. A free virtual channel on its intended hop
2. Free buffer entries for the virtual channel
3. A free cycle on the physical channel

# Interconnection Networks

## ❏ Deadlock

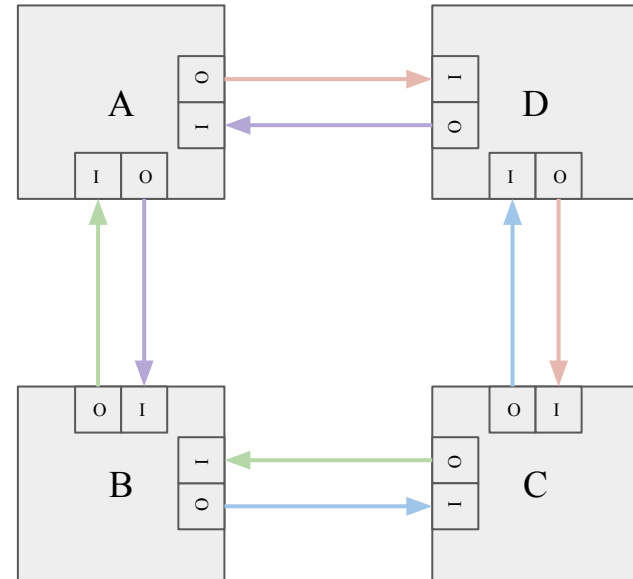
- ❏ Caused by circular dependencies on resources

- ❏ A wants to talk to C
- ❏ B wants to talk to D
- ❏ C wants to talk to A
- ❏ D wants to talk to B



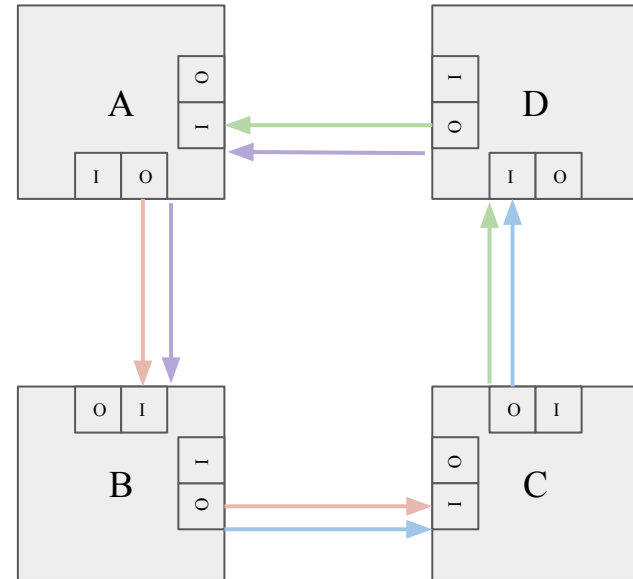
# Interconnection Networks

- ❑ Deadlock
  - ❑ Caused by circular dependencies on resources
- ❑ Avoidance
  - ❑ **Dimension-order routing or turn restriction**
  - ❑ Use virtual channels



# Interconnection Networks

- ❑ Deadlock
  - ❑ Caused by circular dependencies on resources
- ❑ Avoidance
  - ❑ Dimension-order routing or turn restriction
  - ❑ Use **virtual channels**



# Switch-Less Dragonfly on Wafers: A Scalable Interconnection Architecture based on Wafer-Scale Integration

Yinxiao Feng and Kaisheng Ma, Tsinghua University, SC24

# Motivation

- ❑ Physical channel bandwidth (400G/800G) limits per-chip injection bandwidth
- ❑ High-radix switches are expensive, introducing latency and energy overhead
  - ❑ 64-port 400G Infiniband switch cost \$40,000, 200ns port-to-port latency, 1.7KW
- ❑ Modern computing chips can provide abundant I/O and switching bandwidth

# Wafers and chips

- ❑ Traditionally, a chip is limited by the lithographic reticle area (26mm x 33mm) on a monolithic die
- ❑ Integrated-Fan-Out-System-on-Wafer (InFO-SoW)
  - ❑ Eliminates using substrates and PCBs
  - ❑ Achieves higher integration/interconnection density and energy efficiency
  - ❑ 2D-mesh on wafer interconnection
  - ❑ Difficult to scale out

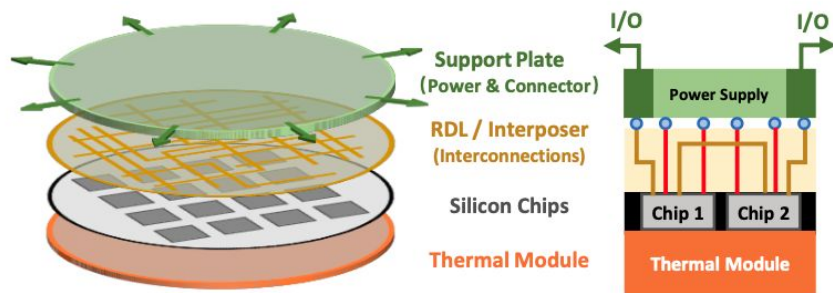
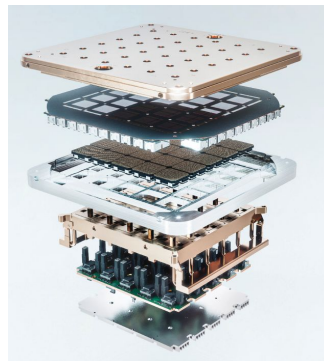
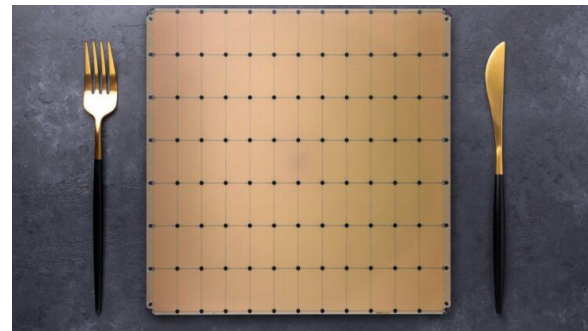


Fig. 1. Profile of the InFO-SoW integration technology. Connectors and power modules are solder-joined to the InFO wafer [17].



Tesla DOJO, 25 D1  
dies for FSD training



Cerebras WSE-2, 850,000 cores

# Wafer-scale integration

Category	Switching Chip			Computing Chip		
Specification	NVSwitch [31]	Tofino2 [32]	Rosetta [33]	H100 [14, 34]	EPYC [35, 36]	DOJO D1 [15]
Physical Lanes	128	256	256	36	128	576
Data-rate (Gbps)	100	50	50	100	32	112
Throughput (Tb/s)	12.8	12.8	12.8	3.6	4	63

- ❑ Ultra-high on/off-wafer bandwidth

- ❑ Comparable to high-end switches

- ❑ Challenges

- ❑ 2D-mesh topology is not scalable
  - ❑ Bandwidth difference between on-wafer and off-wafer
  - ❑ Interconnecting 2D-mesh introduces routing problems, requiring joint optimization on-chip and off-chip

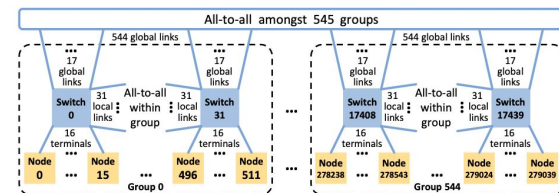
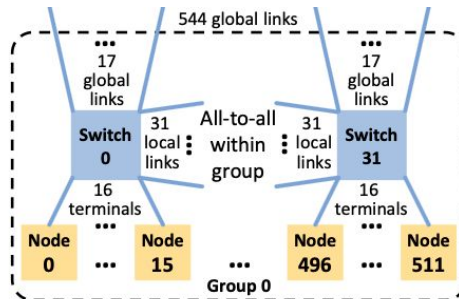
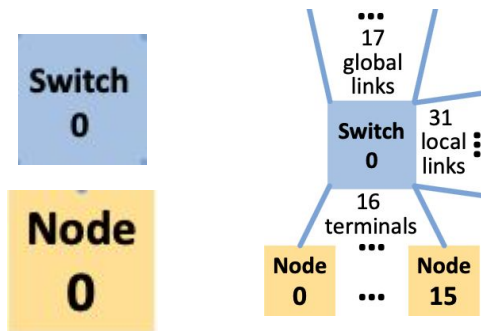


# From switch-based to switch-less

Core contribution:

- ❑ Avoid using costly high-radix switches
- ❑ Improve injection/local throughput and maintain global throughput
- ❑ Scale out 2D-mesh-on-wafer to network-of-wafers
- ❑ Minimal/non-minimal routing algorithm and a novel labeling and interconnection methods to reduce the virtual-channel number
  - ❑ Only one additional virtual channel against traditional Dragonfly is needed

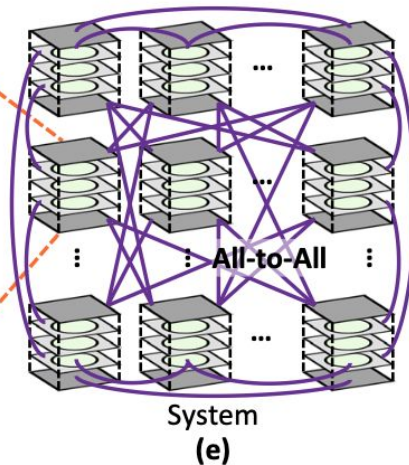
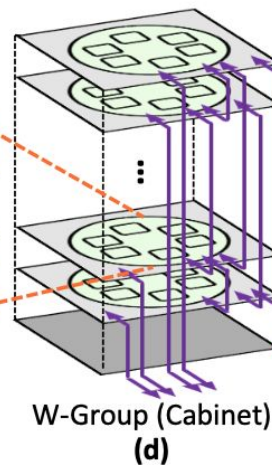
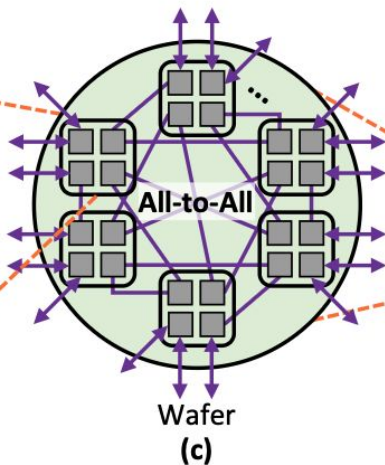
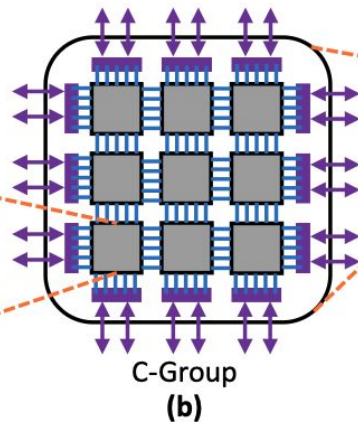
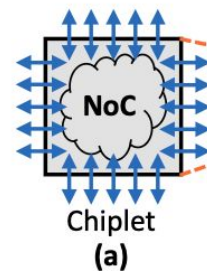
# Switch-less Dragonfly architecture



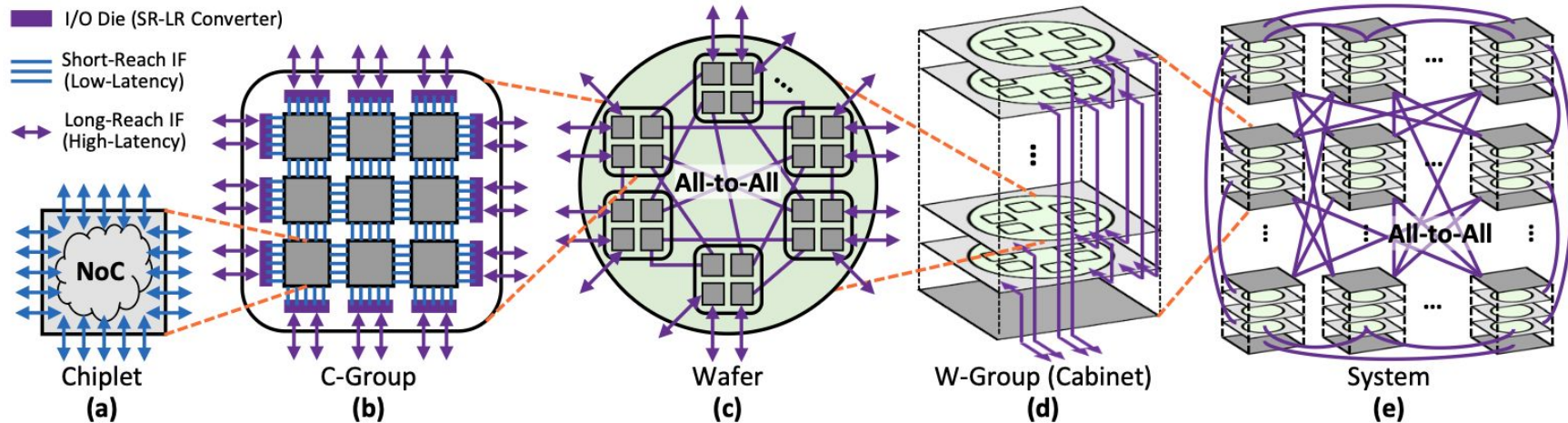
■ I/O Die (SR-LR Converter)

≡≡≡ Short-Reach IF (Low-Latency)

⇄ Long-Reach IF (High-Latency)



# Switch-less Dragonfly architecture



$n$ : # I/O ports

$m$ : scale of  
2D-mesh

$k$ : # external  
I/O

$k = mn$

$a$ : # C-groups

Off-wafer  
connection

$b$ : # wafers

$h$ : # global ports  
of a C-group

$h = k - ab + 1$

$g$ : # W-groups

$N$ : # chiplets

$g = abh + 1$

# Switch-less Dragonfly architecture

## ❑ Scalability

- ❑ 2 C-groups/wafer, 4 wafers, 2x2 chiplets/C-group, 6 interfaces/chiplet → **1k chiplet**

## ❑ Throughput

- ❑ Requirement: global / intra-group injection bandwidth = **1/2**
- ❑ Could achieve higher local throughput by having multiple physical links
- ❑ However, the mesh could introduce contention between intra-C-group and inter-C-group traffic

Injection throughput (flits/cycle/chip)	Switch-based Dragonfly	Switch-less Dragonfly
Global	1	1
Intra-group	1	2
Intra-C-group	2	3

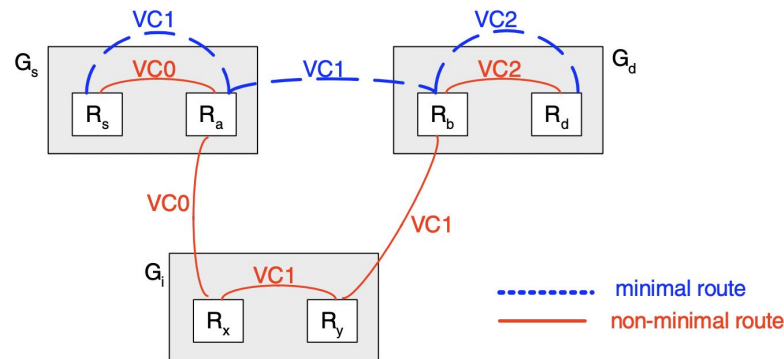
# Switch-less Dragonfly architecture

- ❑ Diameter
  - ❑ Short distance hops will not incur too much latency ( $\sim 1\text{ns}$ )
- ❑ Collective communication
  - ❑ 2D algorithm could be used to reduce latency compared to ring

# hops	Switch-based Dragonfly	Switch-less Dragonfly
Global	1	1
Intra-group	2	2
Intra-C-group	2 (hops between node to switch)	$2(m-1) * 4$

# Interconnection and routing design

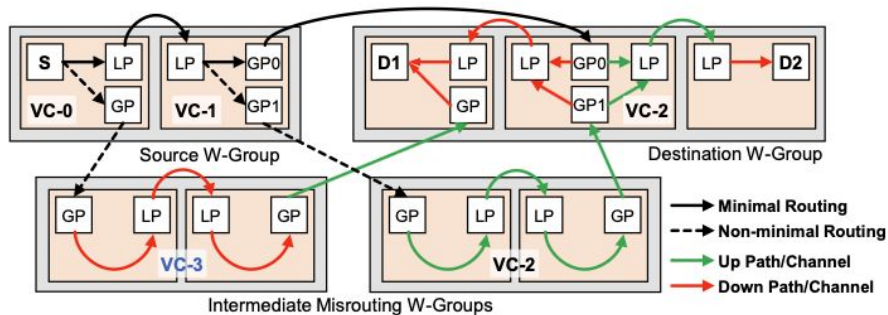
- ❑ Switch-based Dragonfly
  - ❑ Deadlock-free minimal routing: 2 virtual channels
  - ❑ Non-minimal routing: 3 virtual channels
- ❑ Switchless Dragonfly
  - ❑ Minimal routing:
    - ❑ 4 virtual channels
    - ❑ 3 inter-C-group routing steps
    - ❑ 4 intra-C-group routing steps
  - ❑ Non-minimal routing:
    - ❑ 6 virtual channels
    - ❑ 5 inter-C-group routing steps
    - ❑ 6 intra-C-group routing steps



Technology-Driven, Highly-Scalable  
Dragonfly Topology. ISCA 2008

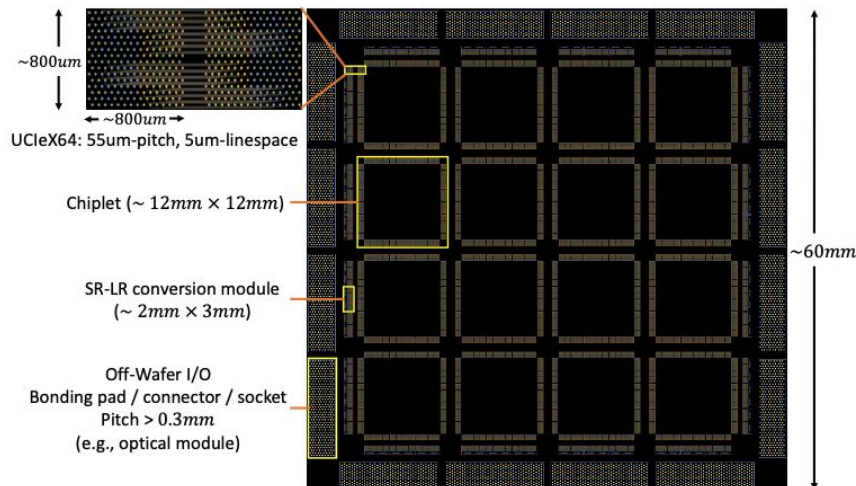
# Interconnection and routing design

- ❑ Switch-based Dragonfly
  - ❑ Deadlock-free minimal routing: 2 virtual channels
  - ❑ Non-minimal routing: 3 virtual channels
- ❑ Switchless Dragonfly
  - ❑ **VC can be reduced through up/down routing (dimension-order)**
  - ❑ Minimal routing:
    - ❑ **4 virtual channels** → 3
    - ❑ 3 inter-C-group routing steps
    - ❑ 4 intra-C-group routing steps
  - ❑ Non-minimal routing:
    - ❑ **6 virtual channels** → 4
    - ❑ 5 inter-C-group routing steps
    - ❑ 6 intra-C-group routing steps



# Evaluation method

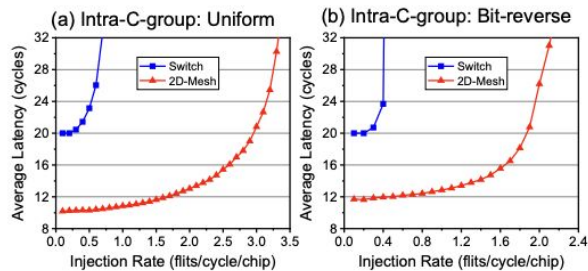
- ❑ Layout: PHYs, chiplets, IO connectors
- ❑ Intra-C-group short-reach
  - ❑ 128 lanes of UCIe per channel
  - ❑ 4096 Gbps/port
  - ❑ 6 channels per edge
  - ❑ Bisection bandwidth: 12TBps
- ❑ Off-C-group long-reach
  - ❑ 8 lanes of 112G SerDes
  - ❑ 896 Gbps/port
  - ❑ 1536 ports per C-group
- ❑ CNSim simulation
- ❑ Workloads
  - ❑ Unicast traffic
  - ❑ Adversarial
  - ❑ Collective ring algorithm



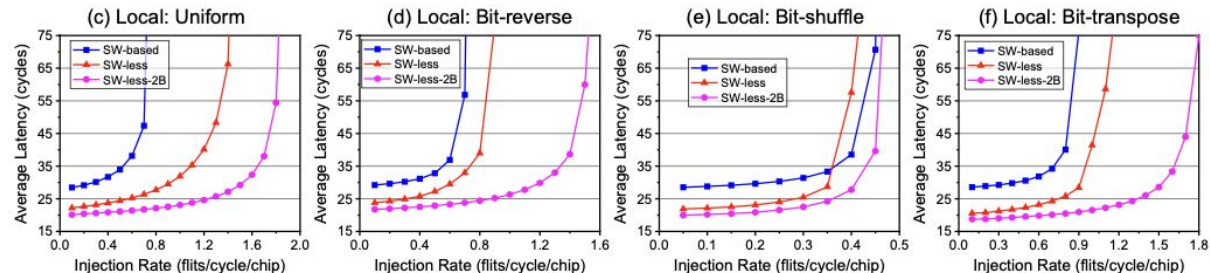


# Evaluation results

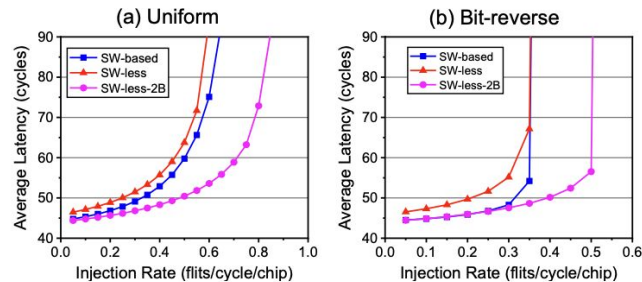
Intra-C-group, 4 chips



Intra-W-group, 32 chips



Global, 1312 chips

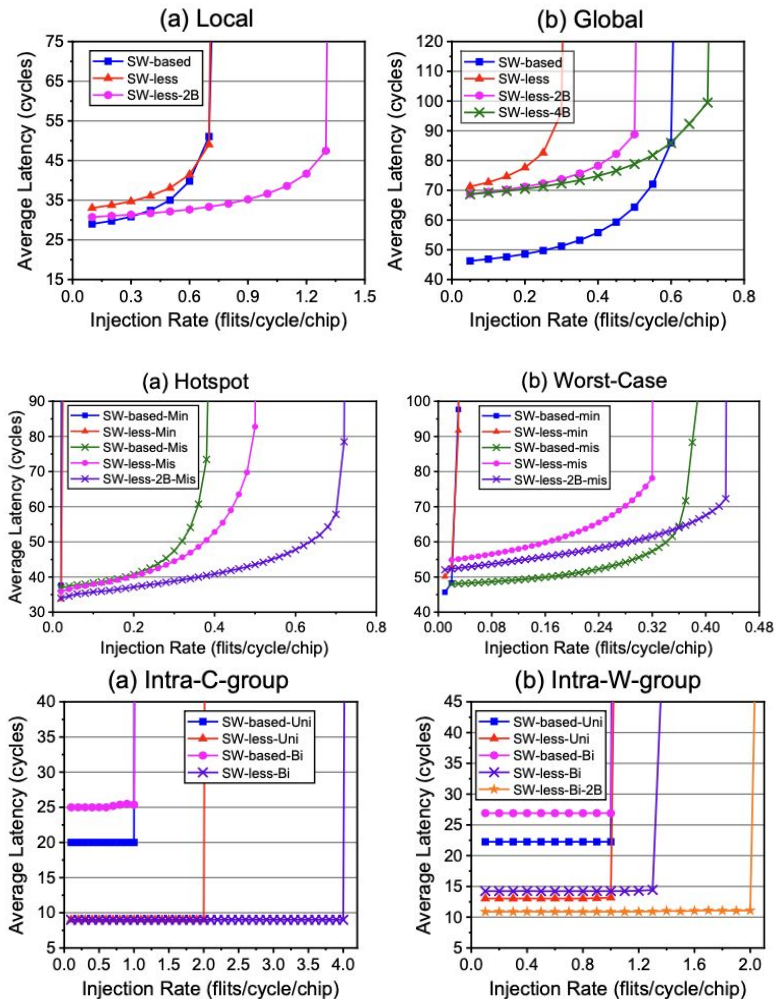


# Evaluation results

18560 chips

Routing

Ring Allreduce



# Discussion

- ❑ A scalable wafer-based interconnection architecture
- ❑ Removes costly high-radix switches while improving local bandwidth
- ❑ Proposes deadlock-free minimal/non-minimal routing
- ❑ Shows a wafer-scale layout

How would switch-less Dragonfly handle failure?

Will boundary chiplet experience be overloaded due to inter-wafer communication?

Do you think the evaluation (injection bandwidth and energy) is thorough?

What are the potential limits of 2D-mesh bisection bandwidth at large scales?

Why is reducing the number of virtual channels so important?