FastTrack: Leveraging Heterogeneous FPGA Wires to Design Low-cost High-performance Soft NoCs

Nachiket Kapre + Tushar Krishna
nachiket@uwaterloo.ca, tushar@ece.gatech.edu

Claim

FPGA overlay NoCs designed to exploit interconnect properties of the FPGA fabric can surpass existing state-of-the-art NoCs by:

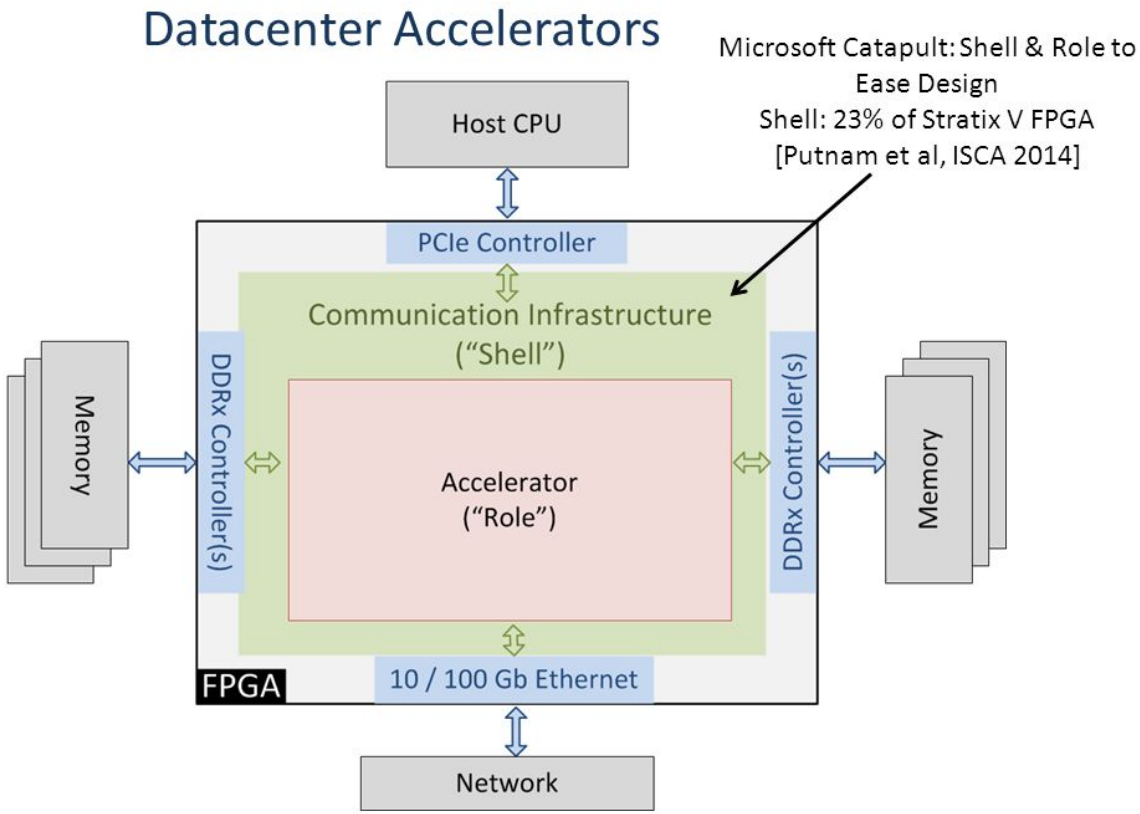
- 2.5–2.8× throughput ↑
- 2.2× energy ↓
- at 2.5× LUT cost ↑

Xilinx Virtex-7 485T FPGA, 8×8 system size, synthetic+real-world traffic.

Context

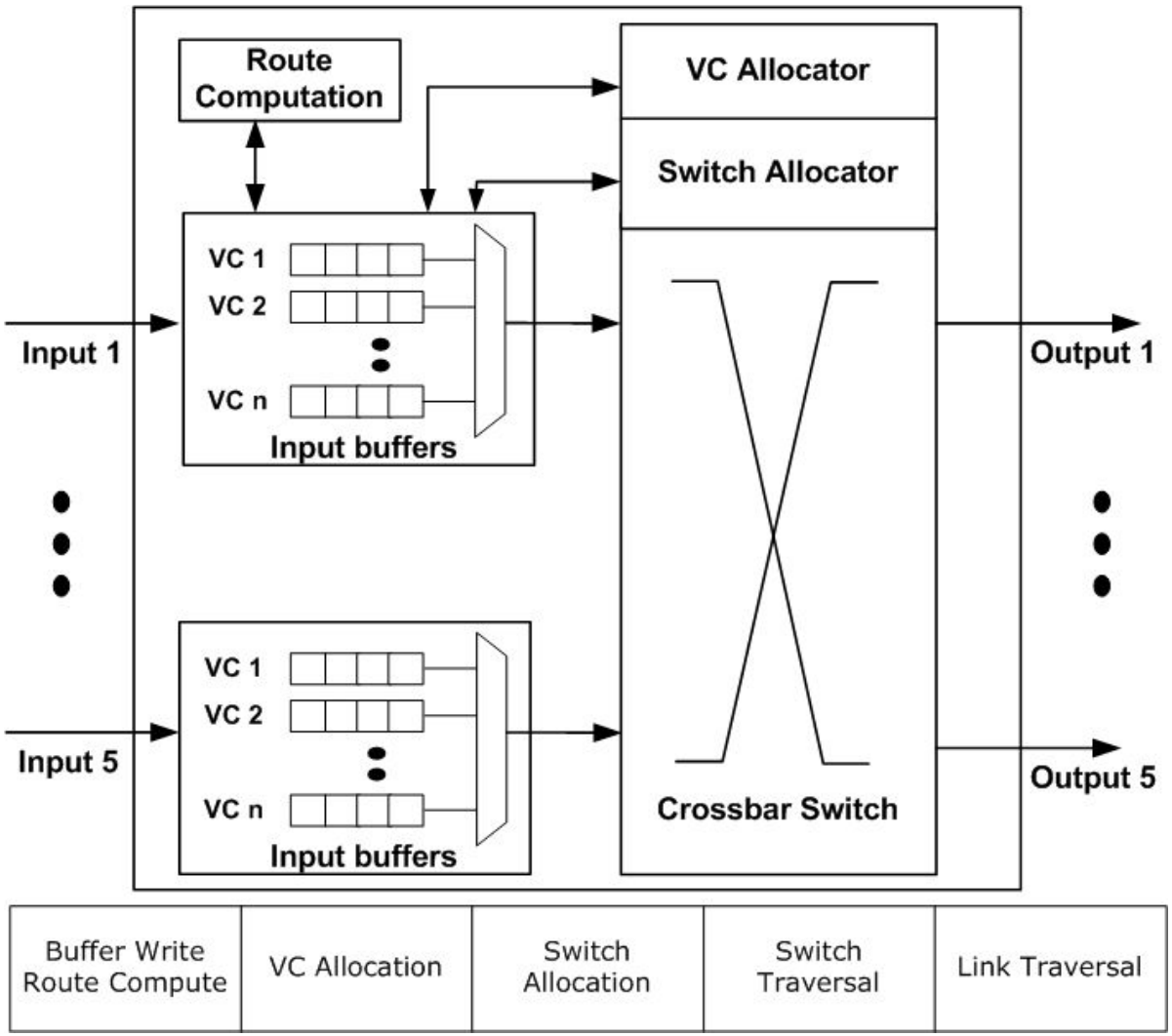
- FPGAs finding comfortable home in datacenters
  - Offloading compute intensive workloads to the FPGA
  - Energy-efficiency, fast coupling to networking
- **Common Infrastructure**: NoCs for apps + system IO

Context

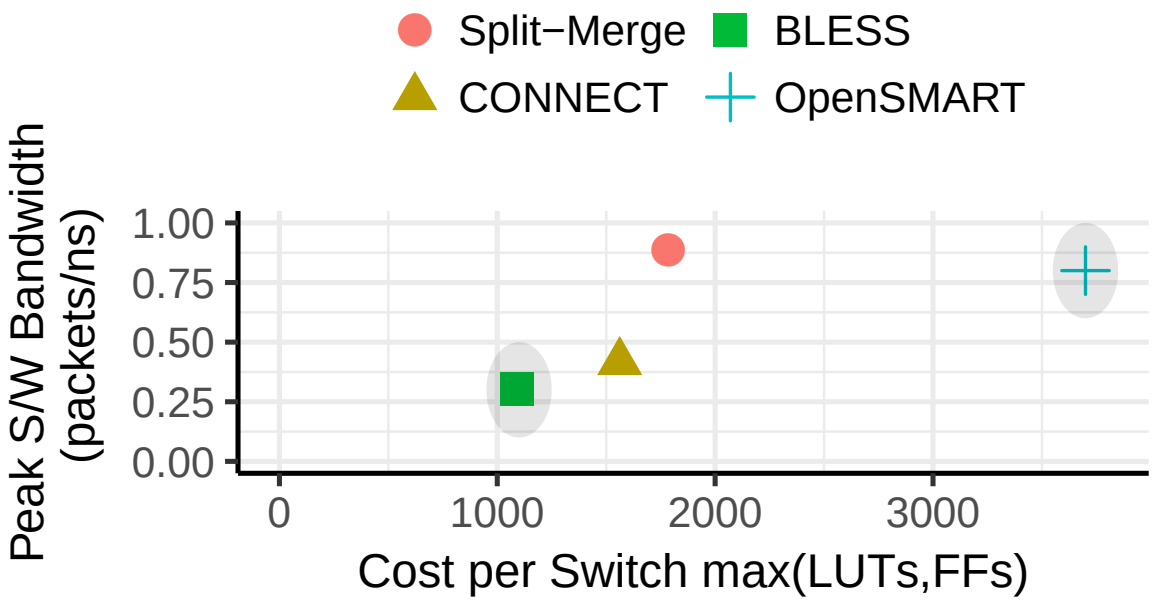


- FPGAs finding comfortable home in datacenters
  - Offloading compute intensive workloads to the FPGA
  - Energy-efficiency, fast coupling to networking
- **Common Infrastructure**: NoCs for apps + system IO

Landscape of contemporary FPGA NoC Routers

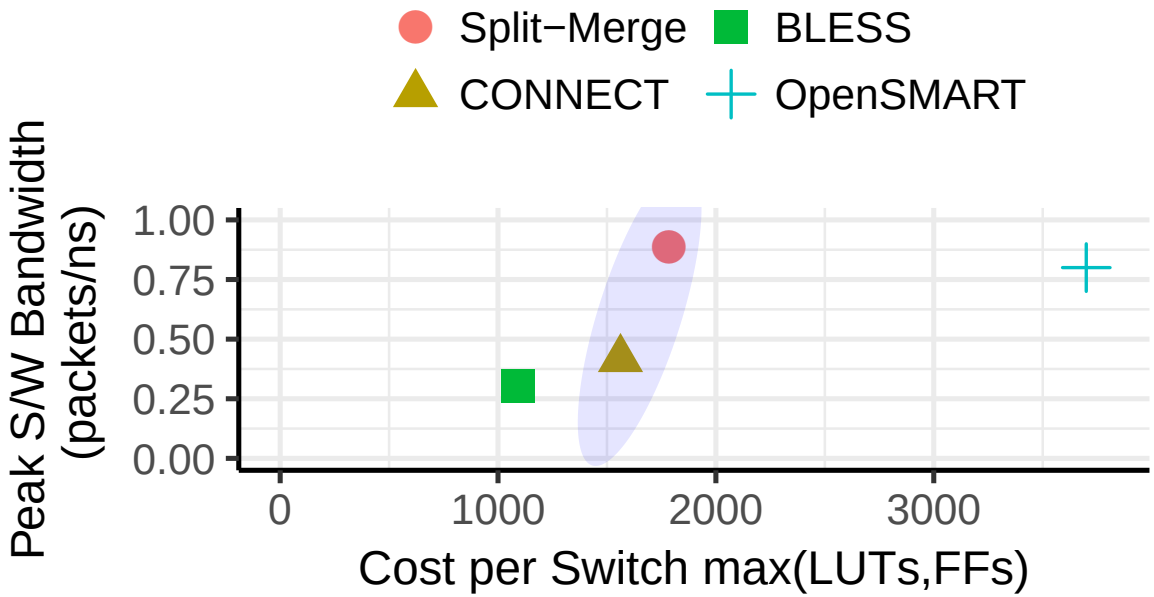


Landscape of contemporary FPGA NoC Routers



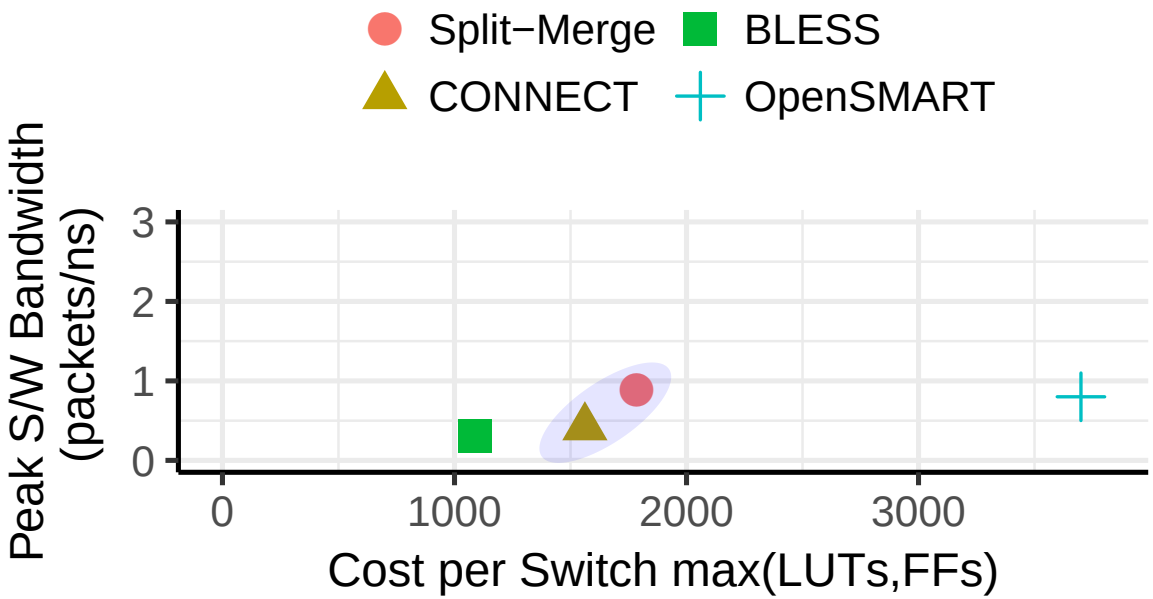
- ASIC clones transplanted onto FPGAs fare poorly! $\rightarrow$ expensive buffers, virtual channels, multi-ported switches
- Even contemporary FPGA routers are expensive and slow
- **FastTrack**: Deflection-routing + Bufferless + Torus

Landscape of contemporary FPGA NoC Routers



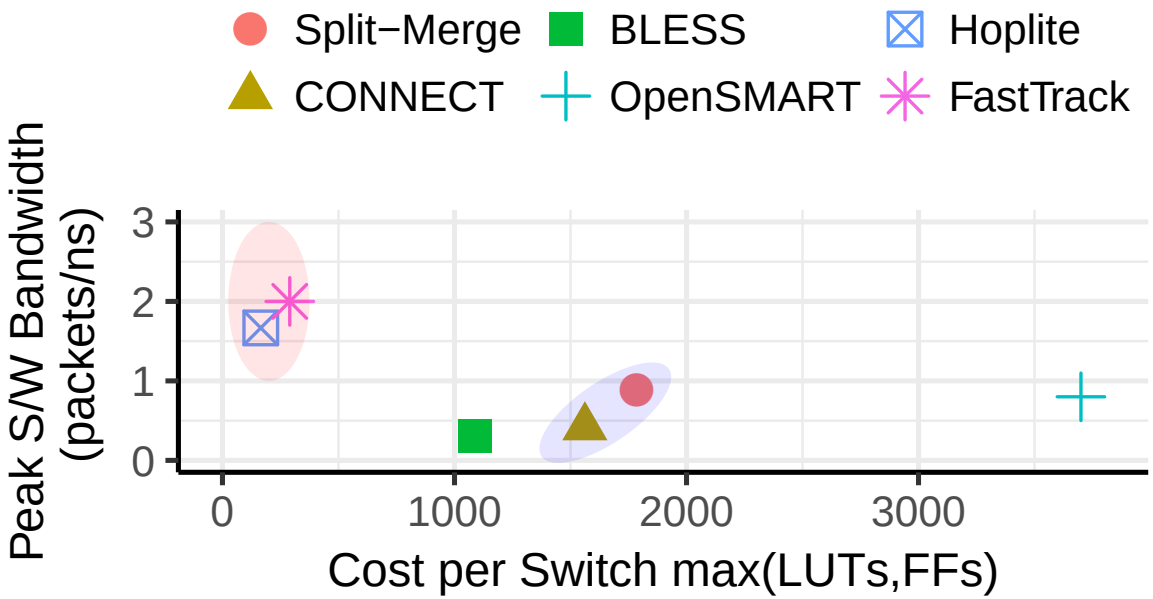
- ASIC clones transplanted onto FPGAs fare poorly! $\rightarrow$ expensive buffers, virtual channels, multi-ported switches
- Even contemporary FPGA routers are expensive and slow
- **FastTrack**: Deflection-routing + Bufferless + Torus

Landscape of contemporary FPGA NoC Routers



- ASIC clones transplanted onto FPGAs fare poorly! → expensive buffers, virtual channels, multi-ported switches
- Even contemporary FPGA routers are expensive and slow
- **FastTrack**: Deflection-routing + Bufferless + Torus

Landscape of contemporary FPGA NoC Routers

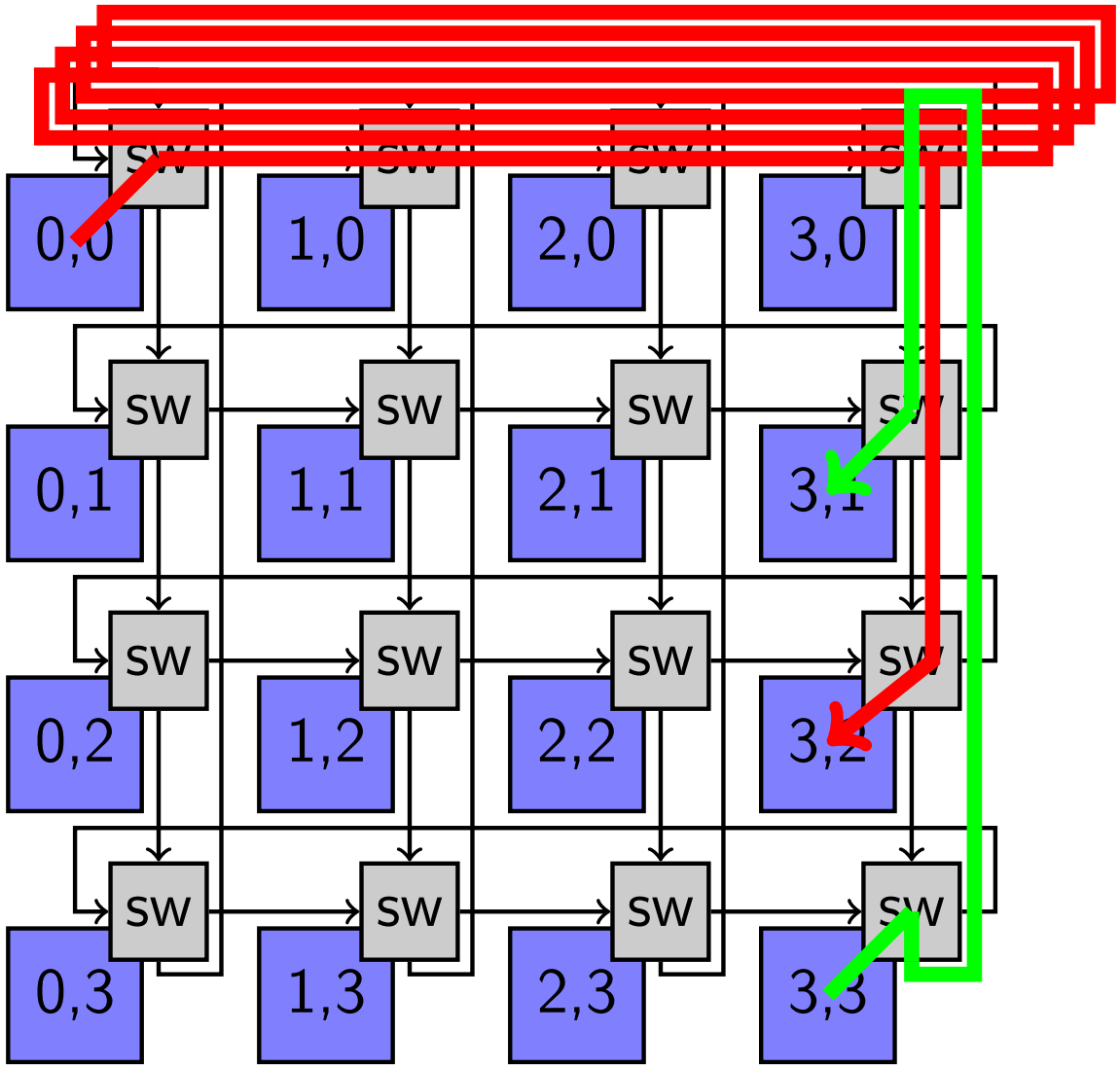


- ASIC clones transplanted onto FPGAs fare poorly! → expensive buffers, virtual channels, multi-ported switches
- Even contemporary FPGA routers are expensive and slow
- **FastTrack**: Deflection-routing + Bufferless + Torus

Qualitative Comparison of FPGA NoC Routers

Router	**Cost**		
	Xbar+Arb	**Buffers**	**VCs**
OpenSMART	✗	✗	✗
BLESS	✗	✓	✓
CONNECT	✗	✗	✗
Split-Merge	✗	✗	✓
Hoplite	✓✓	✓	✓

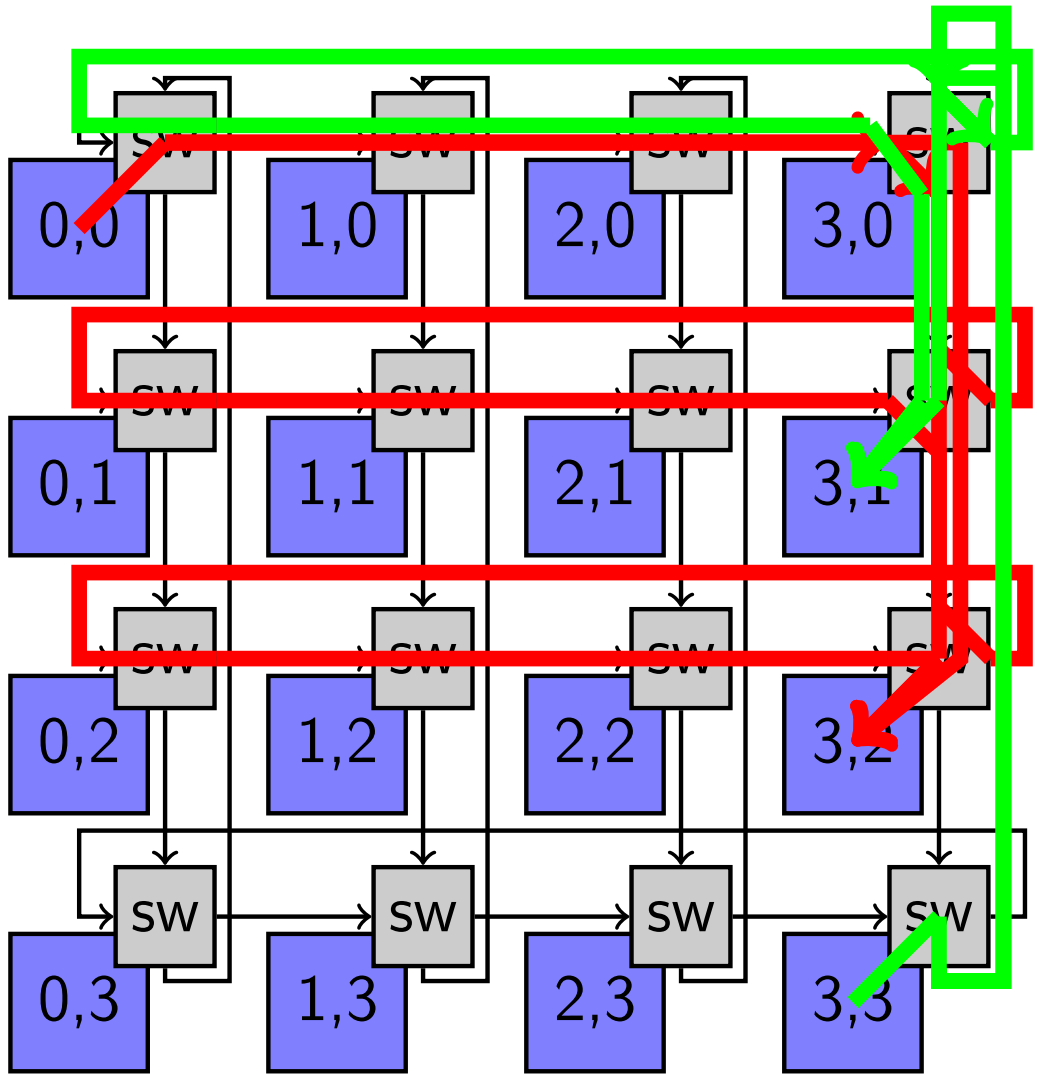
Quick Tutorial on Hoplite



Hoplite: A Deflection-Routed Directional Torus NoC for FPGAs, TRETS 2017

Hoplite: Building Austere Overlay NoCs for FPGAs, FPL 2015

Quick Tutorial on HopliteRT



HopliteRT: An Efficient FPGA NoC for Real-Time Applications, FPT 2017

Qualitative Comparison of FPGA NoC Routers

Router	**Cost**		
	Xbar+Arb	**Buffers**	**VCs**
OpenSMART	✗	✗	✗
BLESS	✗	✓	✓
CONNECT	✗	✗	✗
Split-Merge	✗	✗	✓
Hoplite	✓✓	✓	✓

Qualitative Comparison of FPGA NoC Routers

Router	**Cost**			**Perf**	
	Xbar+Arb	**Buffers**	**VCs**	**Tput**	**Latency**
OpenSMART	✗	✗	✗	✓	✓
BLESS	✗	✓	✓	✓	✗
CONNECT	✗	✗	✗	✓	✓
Split-Merge	✗	✗	✓	✓	✓
Hoplite	✓✓	✓	✓	✗	✗

Challenge

- Deflection routing $\rightarrow$ inefficient use of wiring resources
  - Deflected packets stay in network for longer $\rightarrow$ latency$\uparrow$
  - Steal bandwidth from other traffic $\rightarrow$ throughput $\downarrow$
- **Can we allow improve NoC performance under deflection routing?**
- **Are there unique opportunities provided by the FPGA fabric?**
  - Hoplite cheap in LUT cost...
  - FastTrack $\rightarrow$ inspect FPGA interconnect

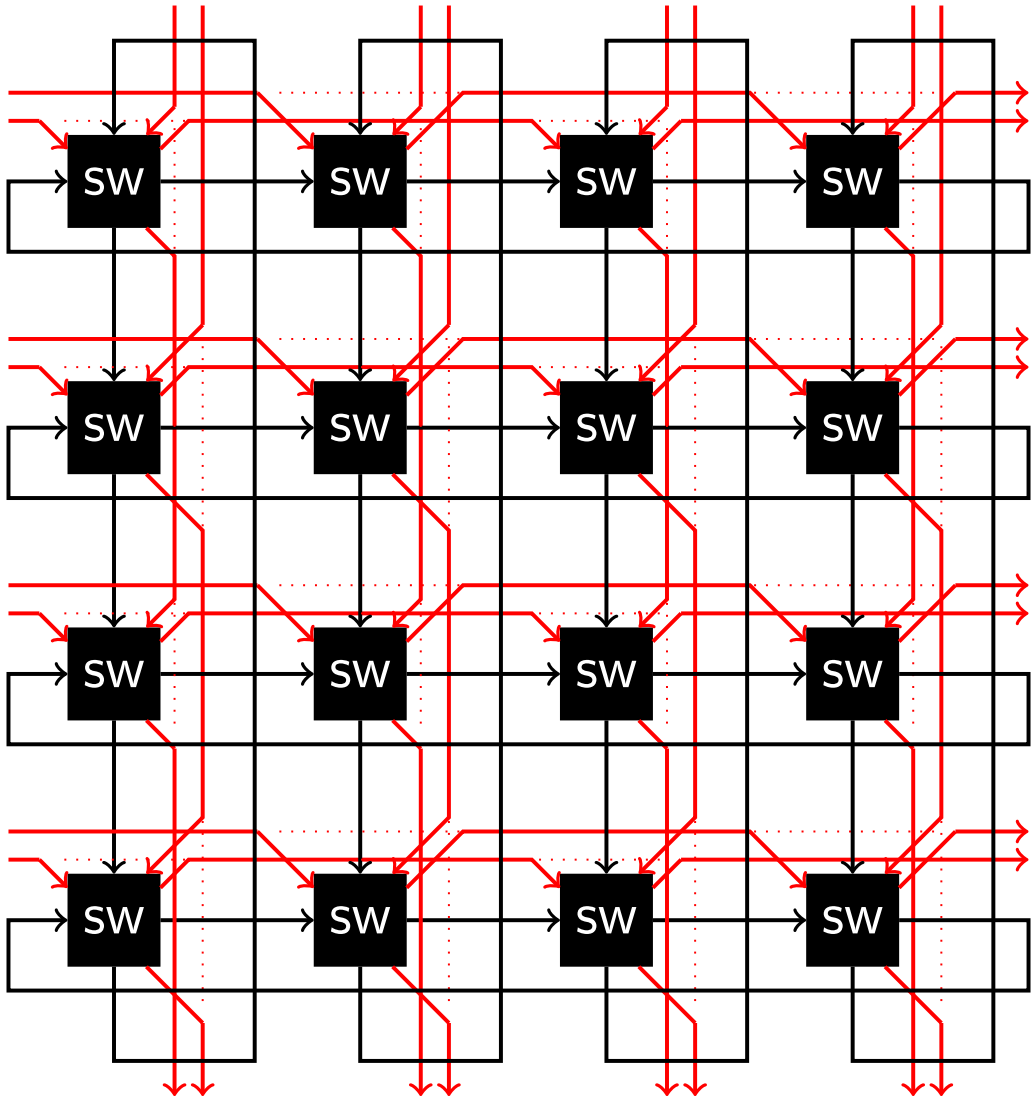
Outline

Outline

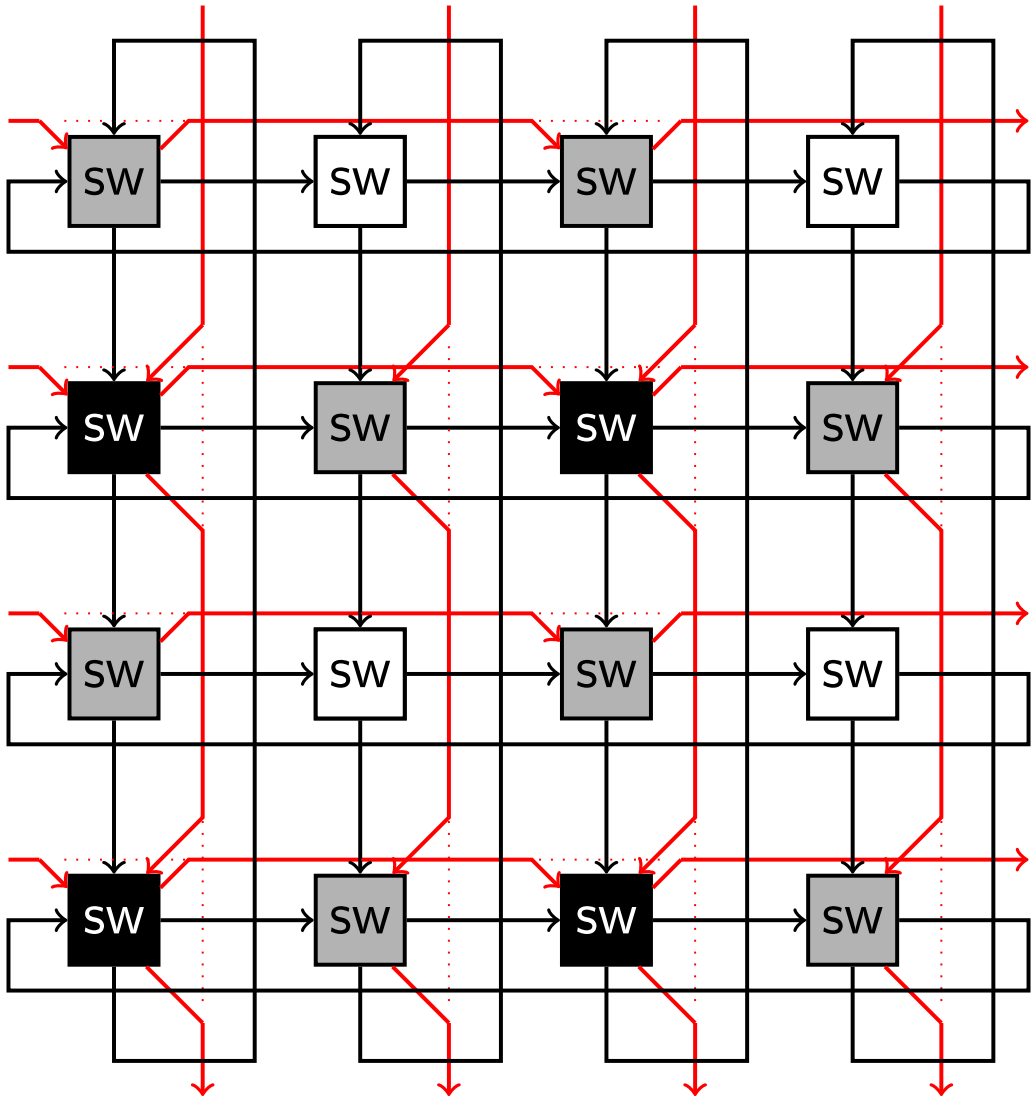
FPGA Wire Speeds

distances not to scale

FastTrack NoC Organization



Depopulated Topology Generation

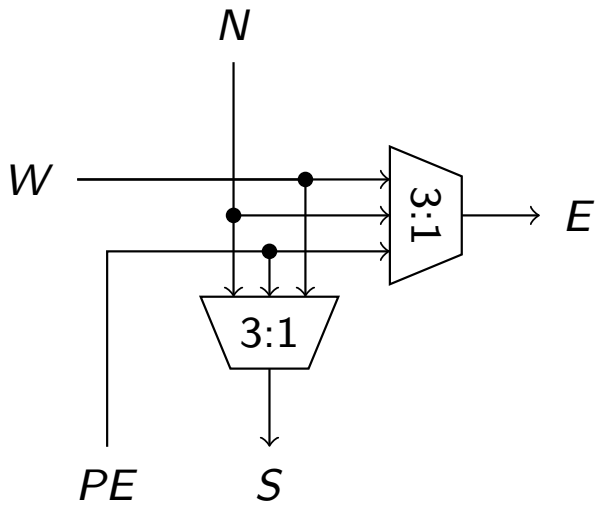


Parametric Topology generation

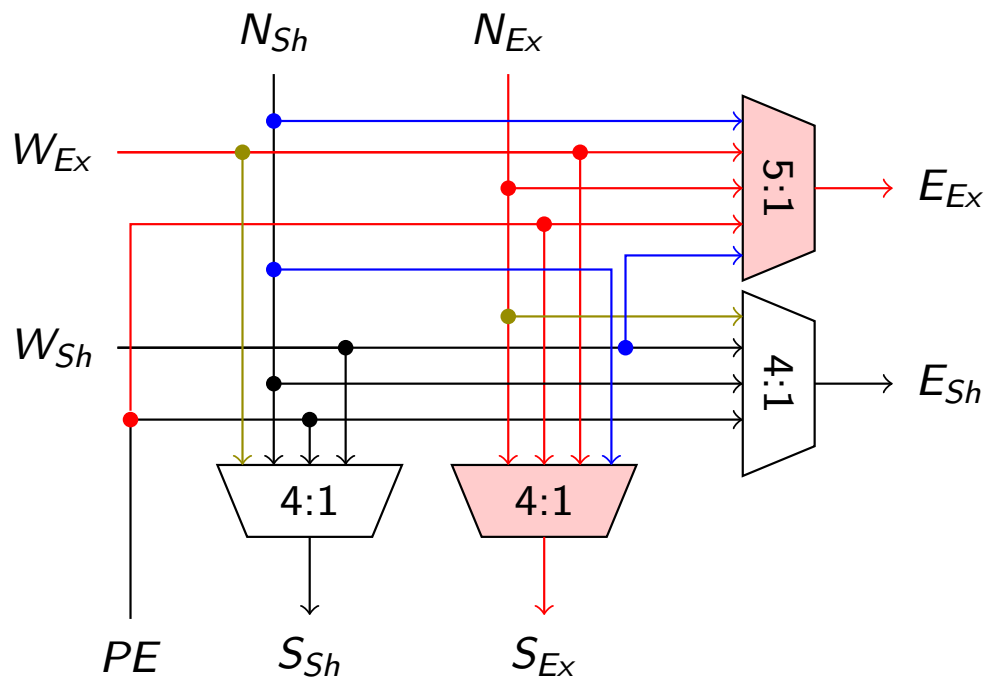
- FPGA NoC parameterized by three terms:
  - **N** System size
  - **D** Distance of express link
  - **R** Depopulation parameter $\rightarrow$ controls how many routers are FastTrack vs. vanilla Hoplite
- Fully populated 4$\times$4 NoC $\rightarrow$ FT(16,2,1)
- Half population 4$\times$4 NoC $\rightarrow$ FT(16,2,2)

Outline

FastTrack Switch Organization

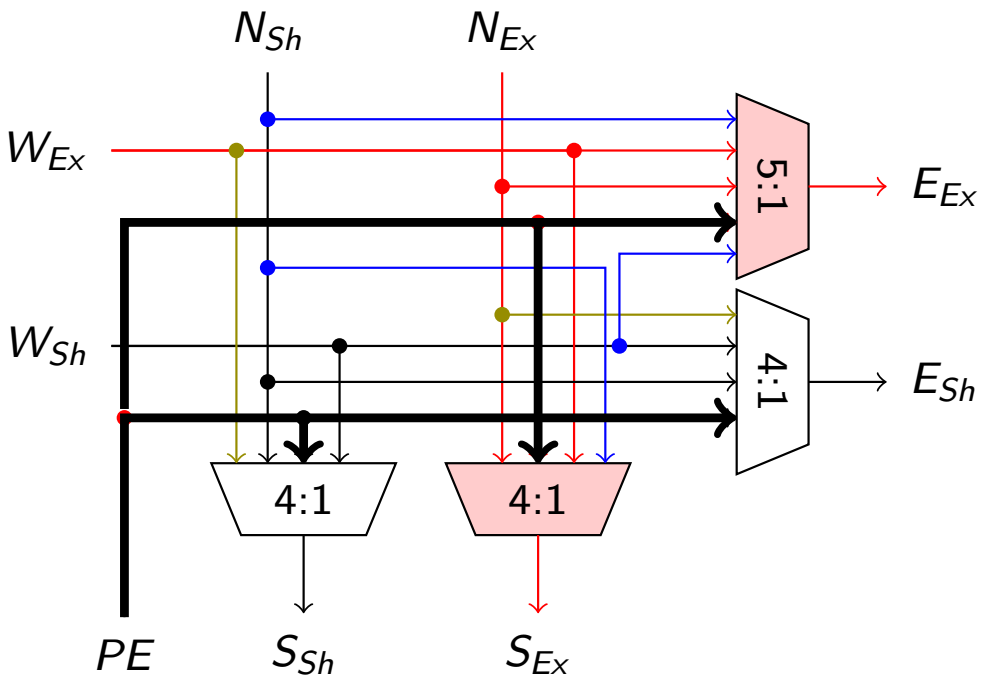


(a) Base HopliteRT



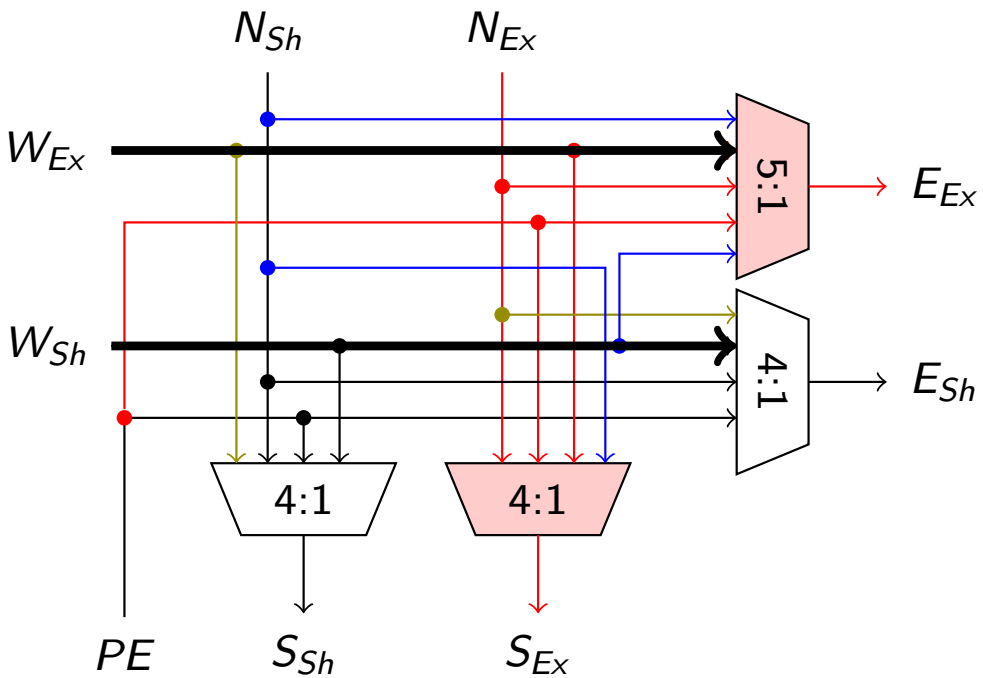
(b) FastTrack

Switch Operation



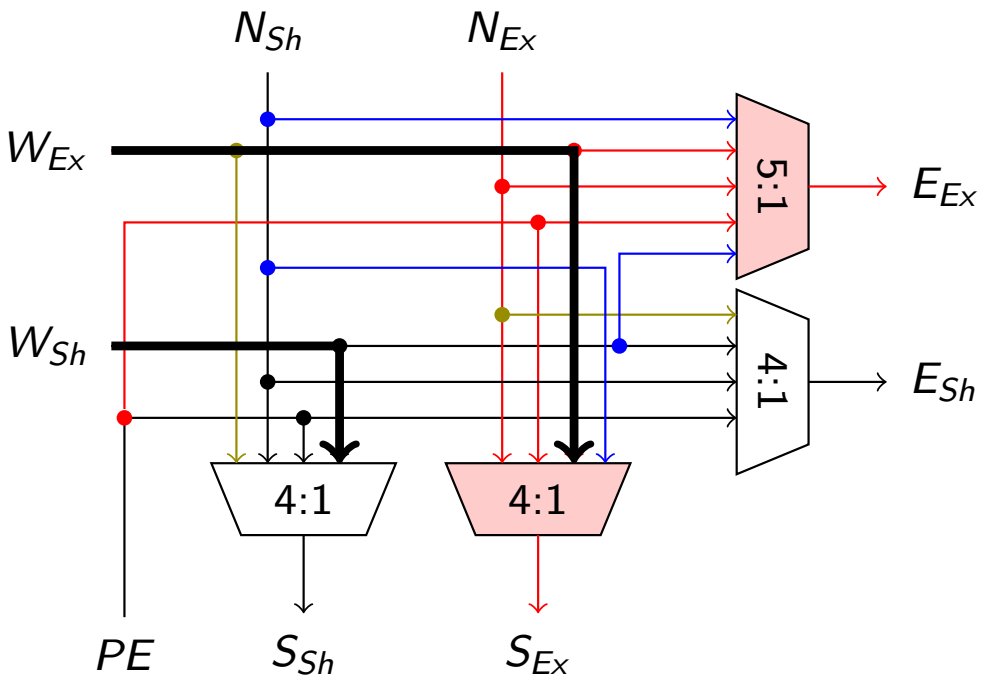
- Packets can start in either **short** or **express** links
- DOR routing function: travel in X first, then Y
- Packets can *upgrade* to fast links if they can
- Packets can *downgrade* to slow links only on turn!
- Livelock avoidance: $W \rightarrow S > N \rightarrow S$
- Express links=higher priority, deflected packets acquire higher priority $\rightarrow$ progress

Switch Operation



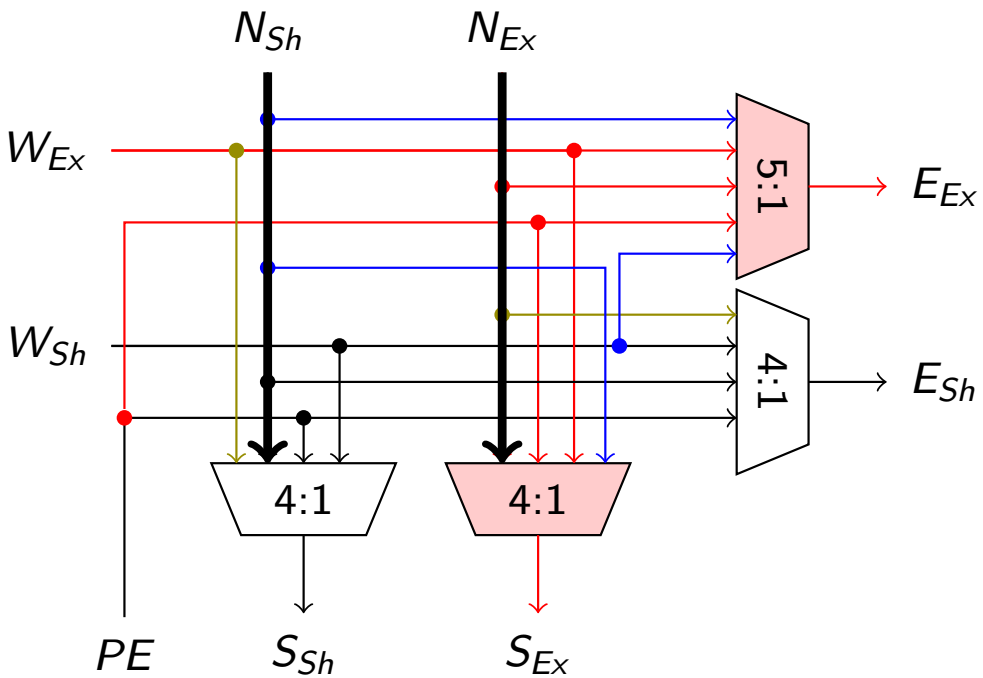
- Packets can start in either **short** or **express** links
- DOR routing function: travel in X first, then Y
- Packets can *upgrade* to fast links if they can
- Packets can *downgrade* to slow links only on turn!
- Livelock avoidance: $W \rightarrow S > N \rightarrow S$
- Express links=higher priority, deflected packets acquire higher priority $\rightarrow$ progress

Switch Operation



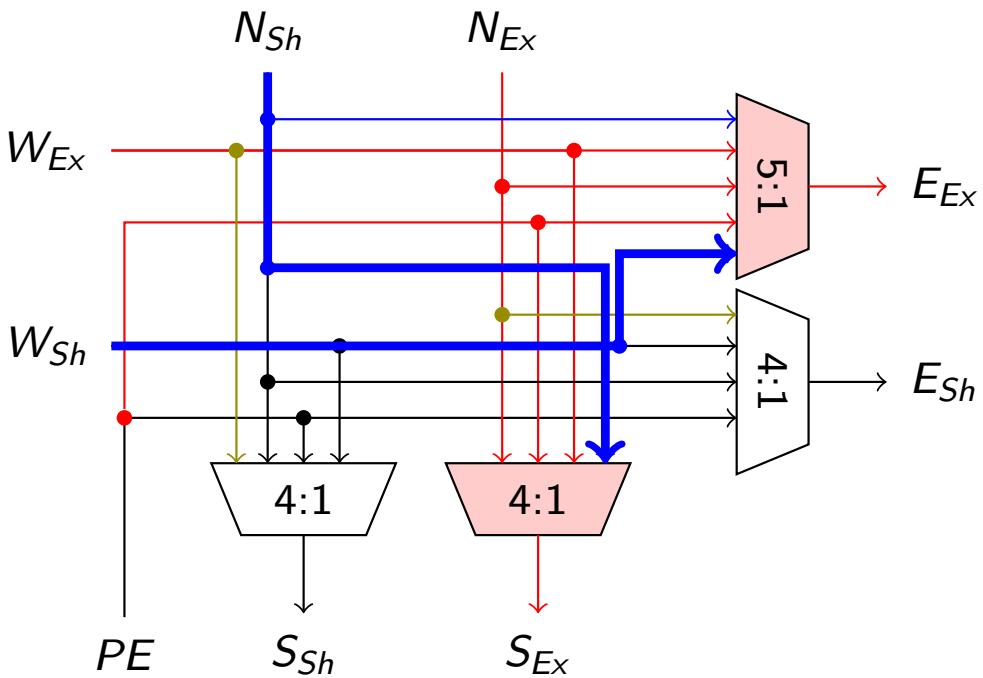
- Packets can start in either **short** or **express** links
- DOR routing function: travel in X first, then Y
- Packets can *upgrade* to fast links if they can
- Packets can *downgrade* to slow links only on turn!
- Livelock avoidance: $W \rightarrow S > N \rightarrow S$
- Express links=higher priority, deflected packets acquire higher priority $\rightarrow$ progress

Switch Operation



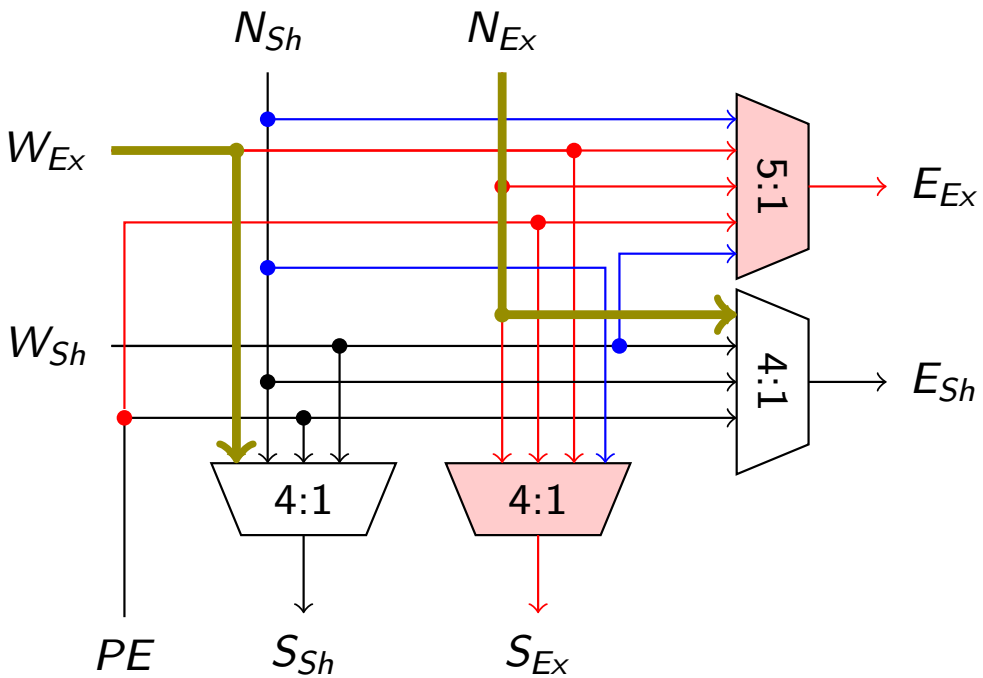
- Packets can start in either **short** or **express** links
- DOR routing function: travel in X first, then Y
- Packets can *upgrade* to fast links if they can
- Packets can *downgrade* to slow links only on turn!
- Livelock avoidance: $W \rightarrow S > N \rightarrow S$
- Express links=higher priority, deflected packets acquire higher priority $\rightarrow$ progress

Switch Operation



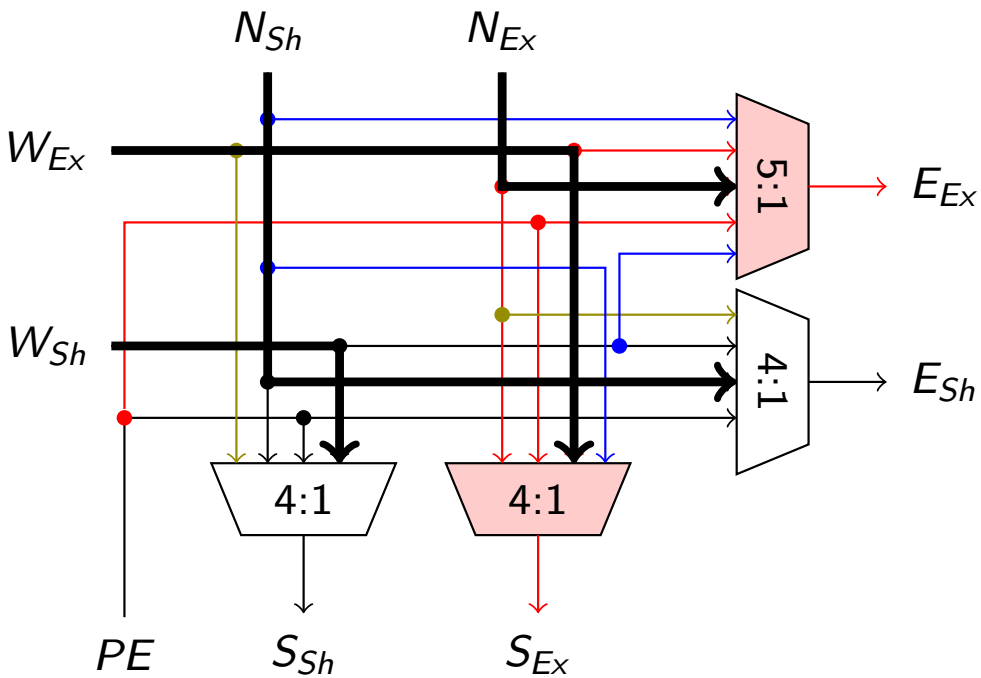
- Packets can start in either **short** or **express** links
- DOR routing function: travel in X first, then Y
- Packets can *upgrade* to fast links if they can
- Packets can *downgrade* to slow links only on turn!
- Livelock avoidance: $W \rightarrow S > N \rightarrow S$
- Express links=higher priority, deflected packets acquire higher priority $\rightarrow$ progress

Switch Operation



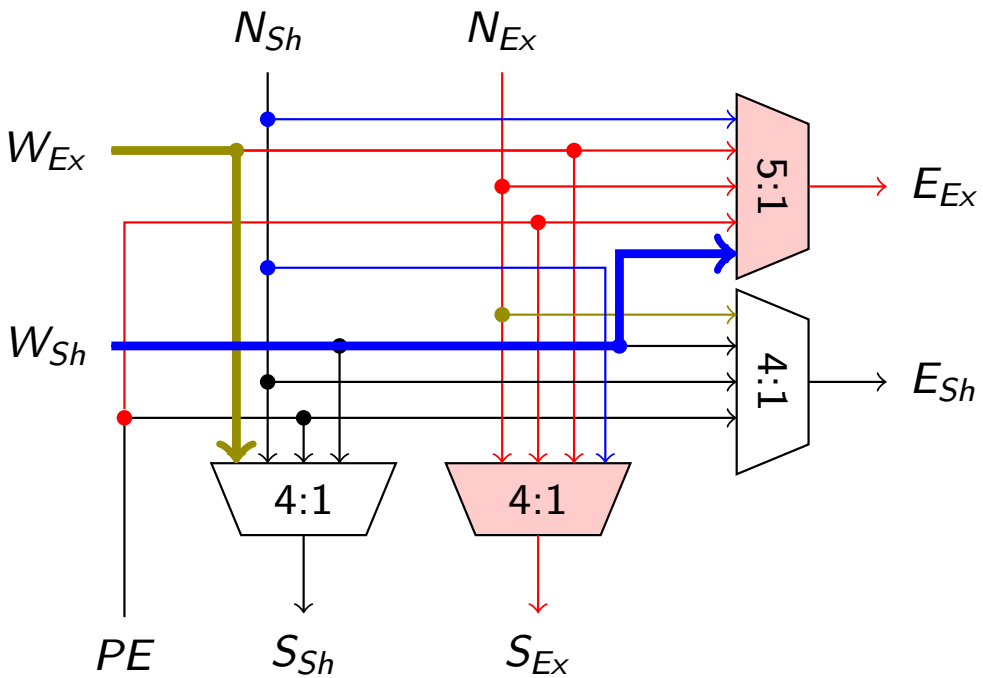
- Packets can start in either **short** or **express** links
- DOR routing function: travel in X first, then Y
- Packets can *upgrade* to fast links if they can
- Packets can *downgrade* to slow links only on turn!
- Livelock avoidance: $W \to S > N \to S$
- Express links=higher priority, deflected packets acquire higher priority $\to$ progress

Switch Operation



- Packets can start in either **short** or **express** links
- DOR routing function: travel in X first, then Y
- Packets can *upgrade* to fast links if they can
- Packets can *downgrade* to slow links only on turn!
- Livelock avoidance: $W \rightarrow S > N \rightarrow S$
- Express links=higher priority, deflected packets acquire higher priority $\rightarrow$ progress

Switch Operation



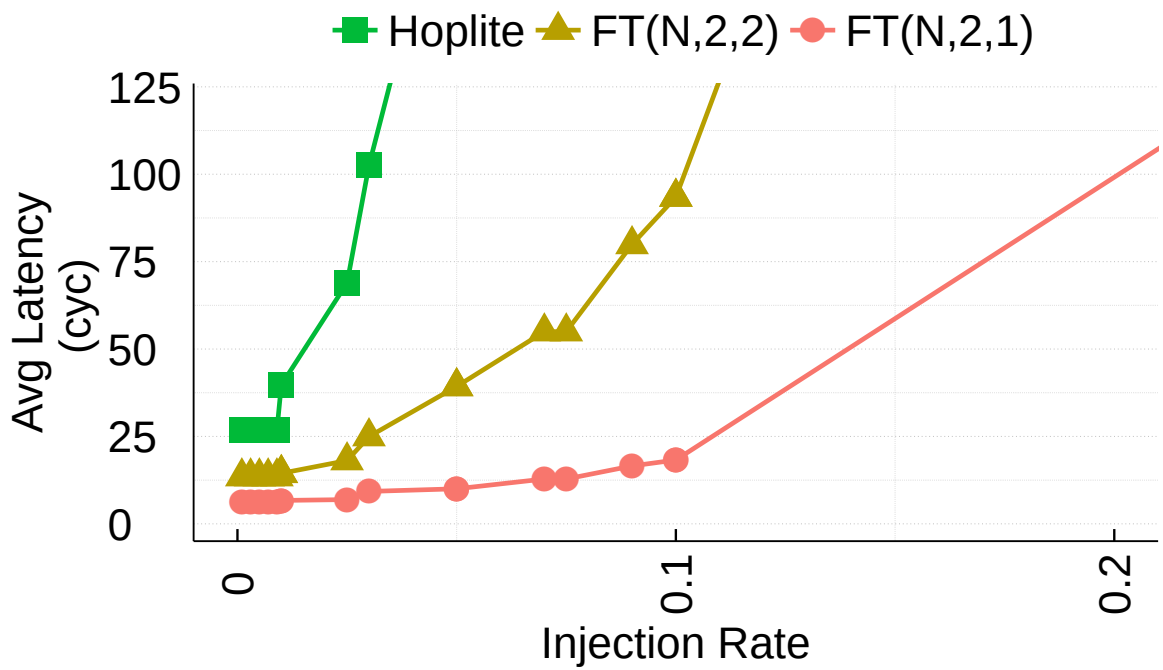
- Packets can start in either **short** or **express** links
- DOR routing function: travel in X first, then Y
- Packets can *upgrade* to fast links if they can
- Packets can *downgrade* to slow links only on turn!
- Livelock avoidance: $W \to S > N \to S$
- Express links=higher priority, deflected packets acquire higher priority $\to$ progress

Outline

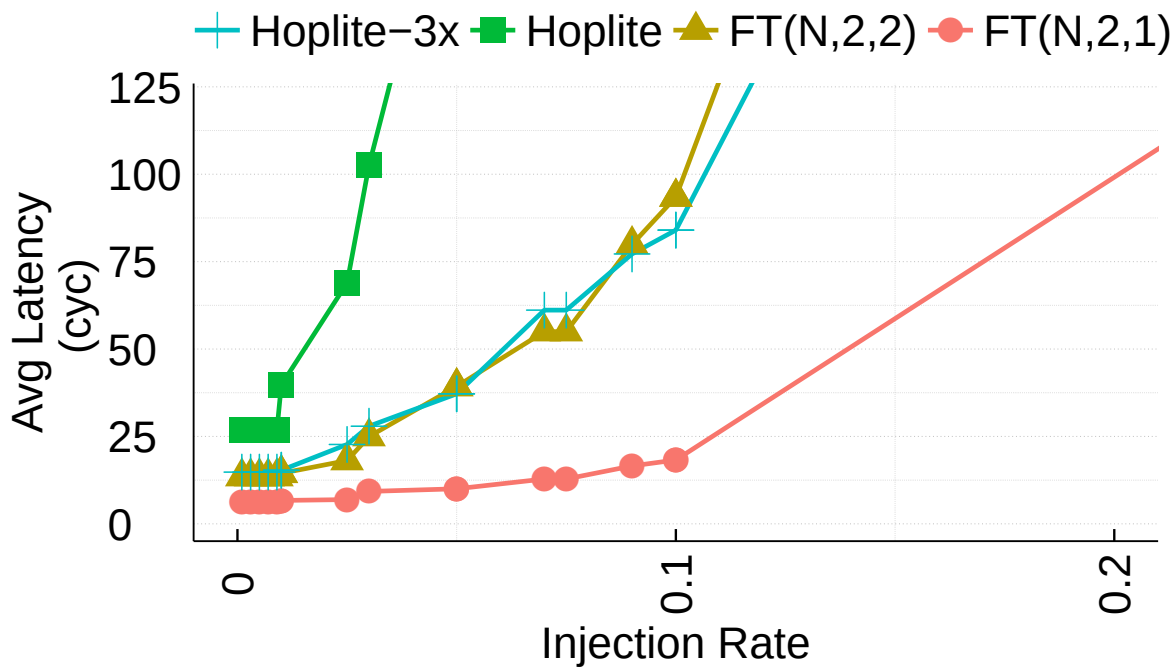
Experimental Setup

- RTL implementation of Routers $\rightarrow$ parameterized
  - D, R parameters control cost
- Cycle-accurate simulations $\rightarrow$ Verilator
- FPGA synthesis + out-of-context place-and-route + XDC floorplanning constraints $\rightarrow$ Vivado
- Benchmarking:
  - Synthetic traffic patterns at various injection rates
  - Traces from real workloads SpMV, Graph Analytics, Multi-processing
- Measure sustained throughput, average latency, power model

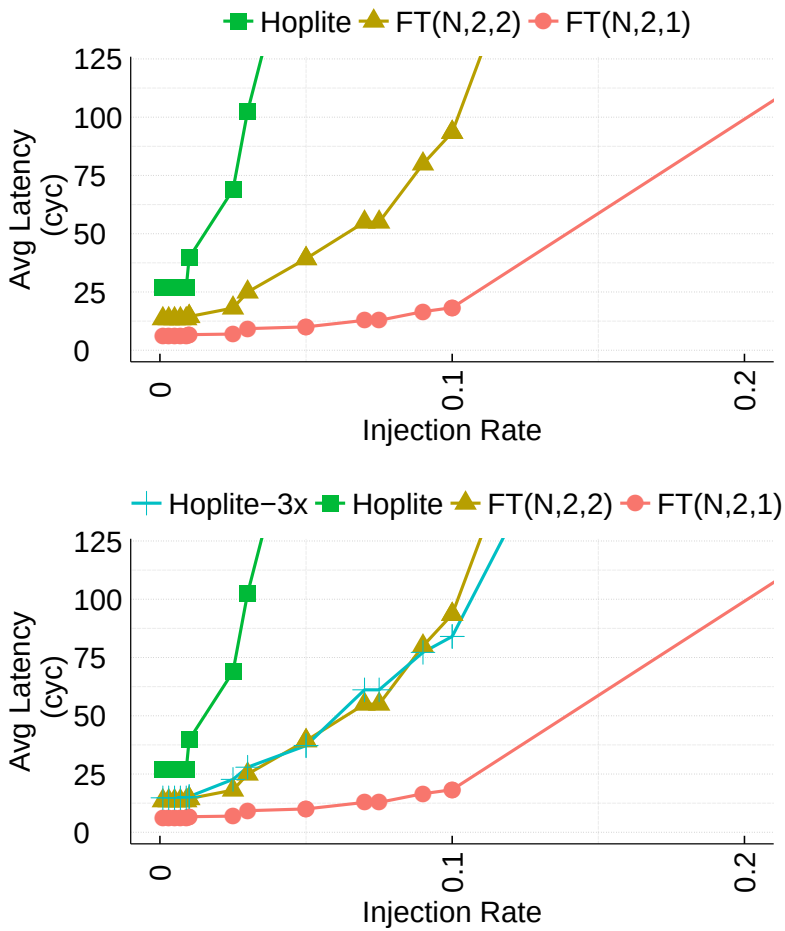
Avg. Latency RANDOM traffic 8×8 NoC



Avg. Latency RANDOM traffic 8×8 NoC

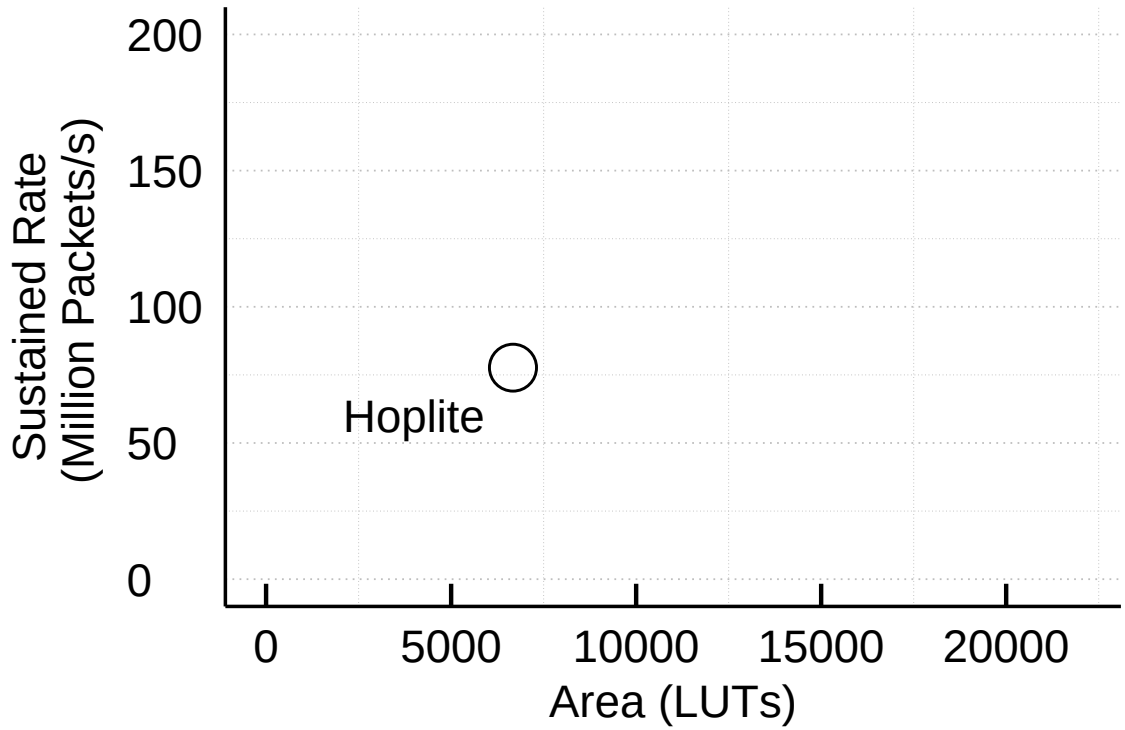


Avg. Latency RANDOM traffic

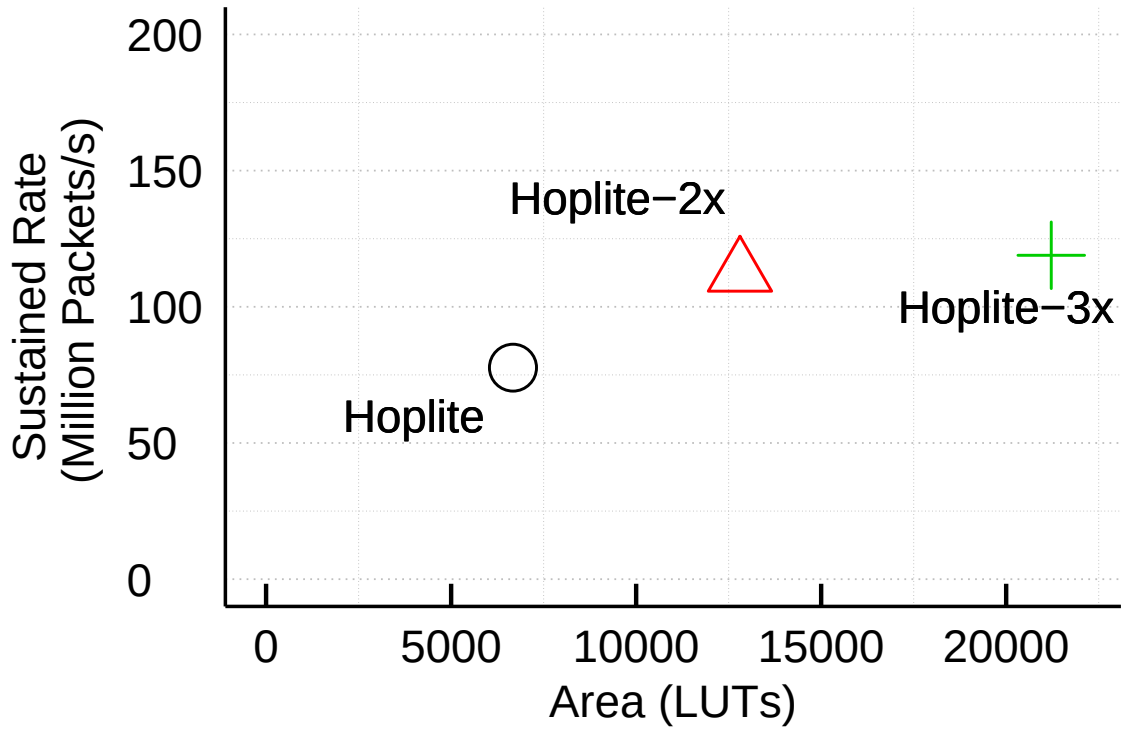


- FastTrack saturates at 4–5× higher injection rate than Hoplite
- vs Replicated Hoplite, still better but by smaller margin
- Replicated Hoplite has a new kind of livelock possibility (delivery)

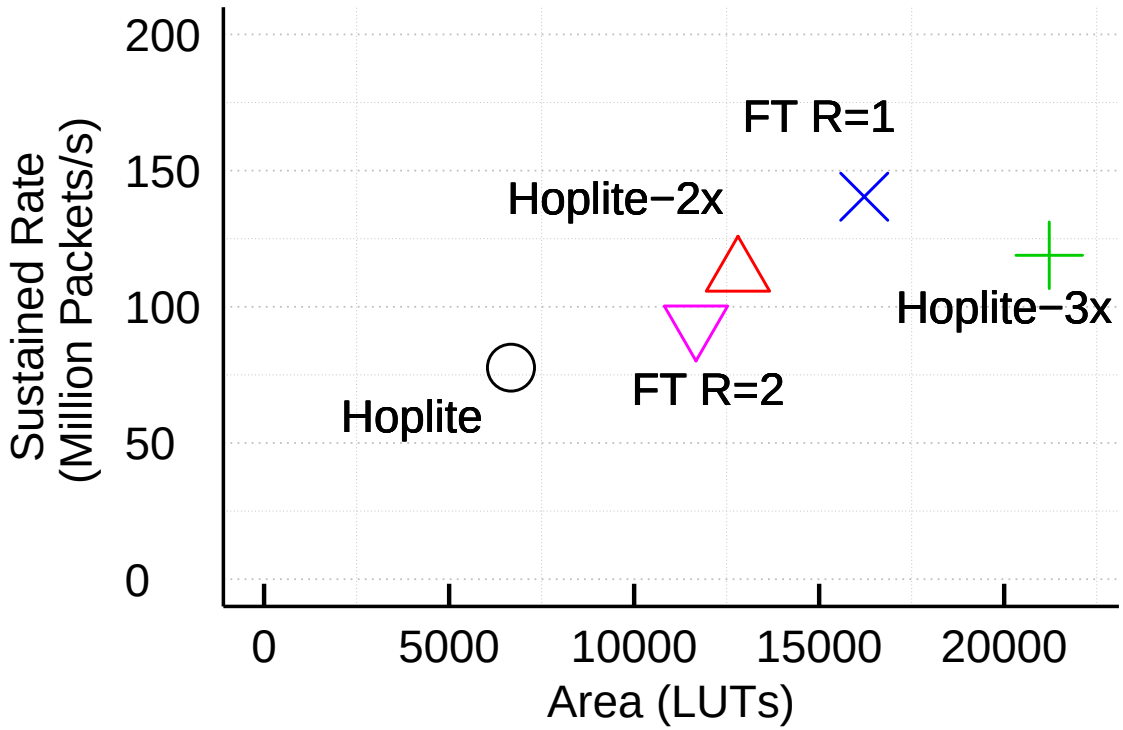
Results – LUT vs Throughput 8×8 NoC



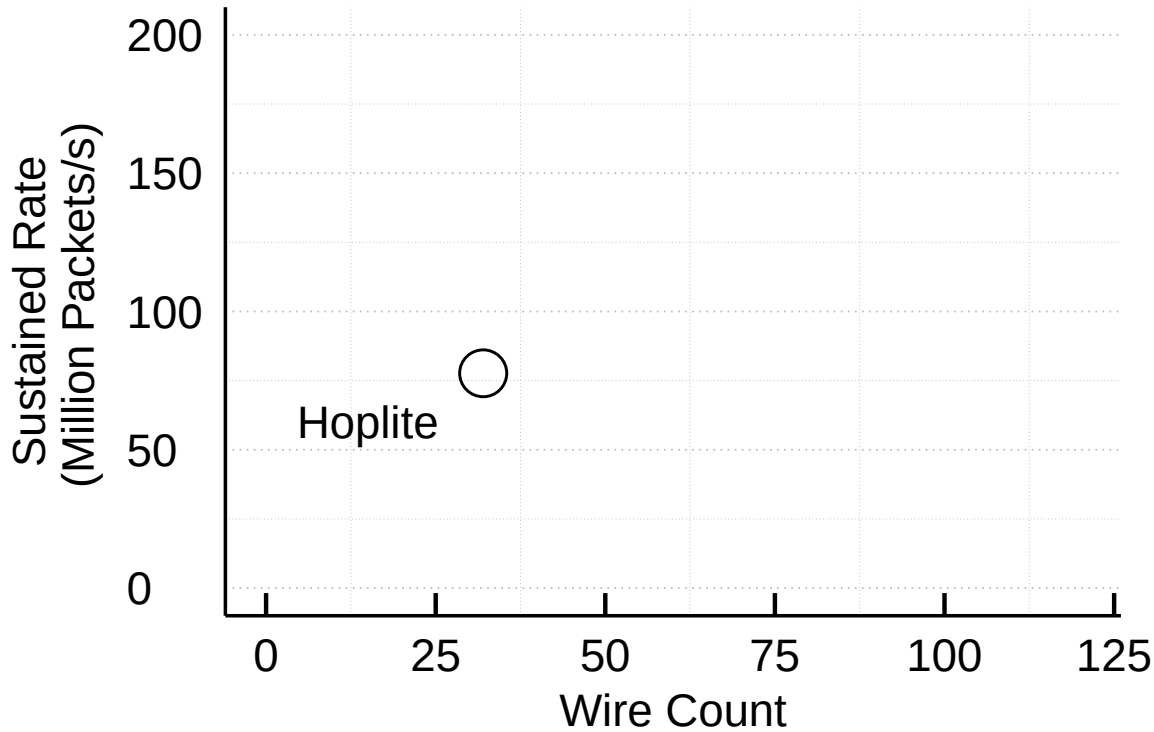
Results – LUT vs Throughput 8×8 NoC



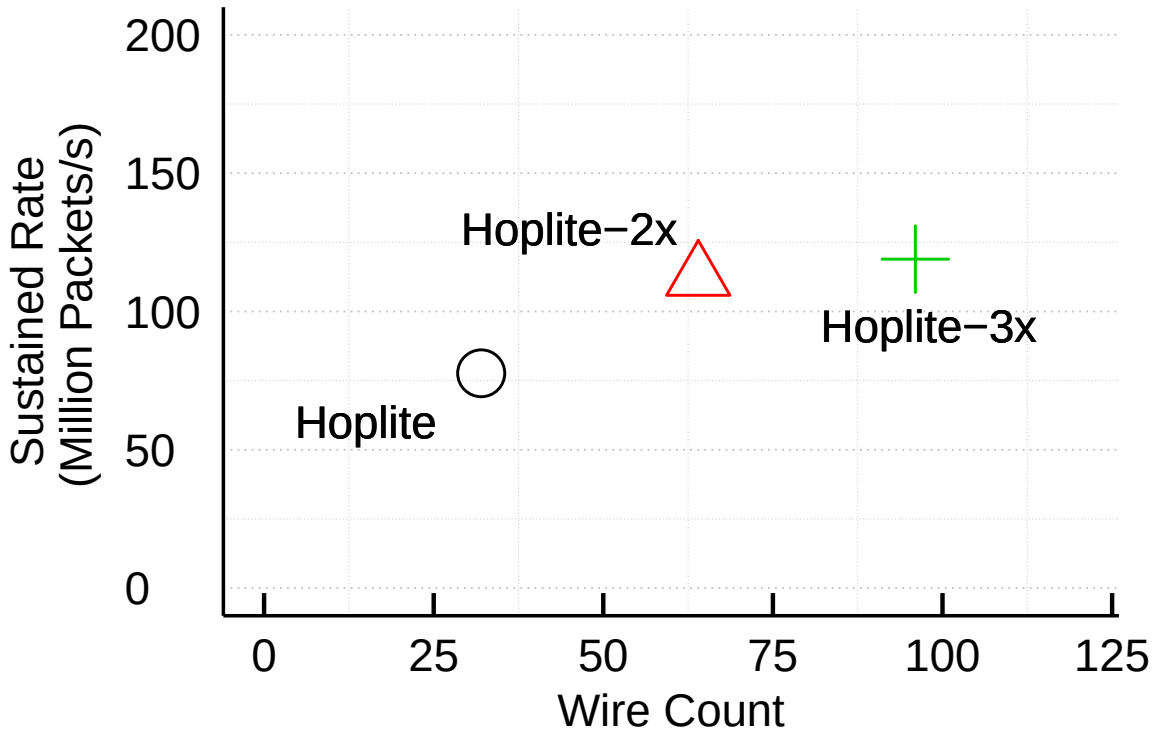
Results – LUT vs Throughput 8×8 NoC



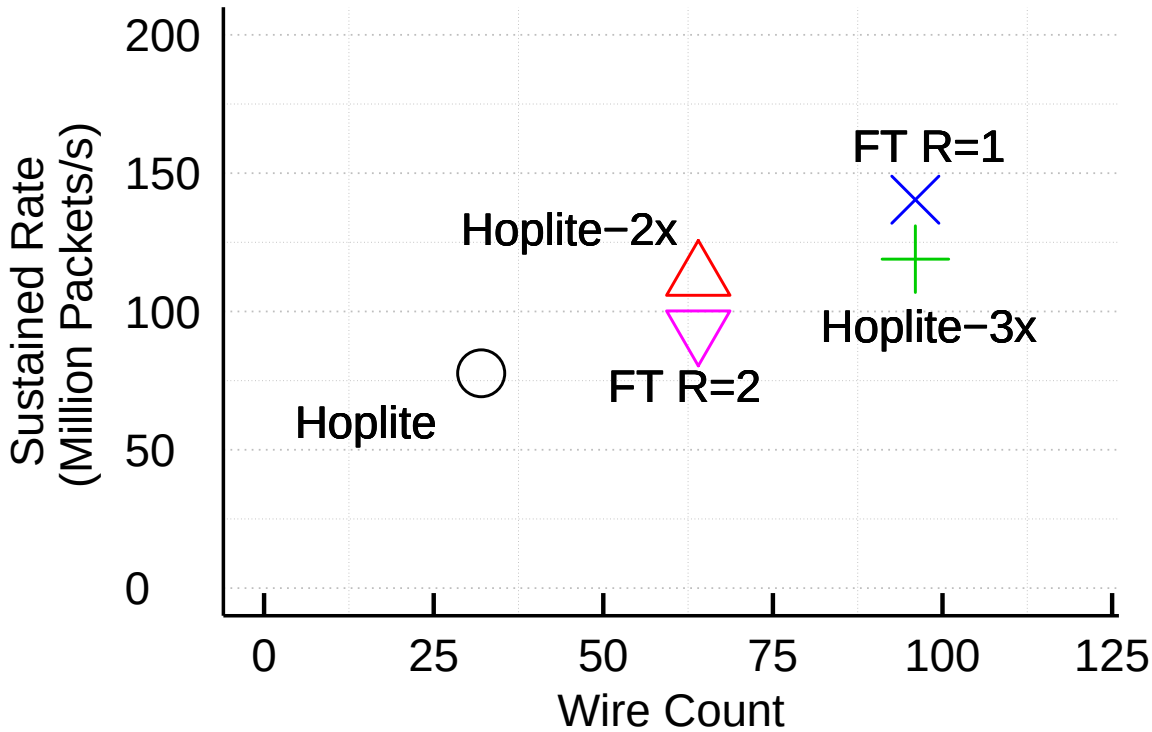
Results – Wiring vs. Throughput 8×8 NoC



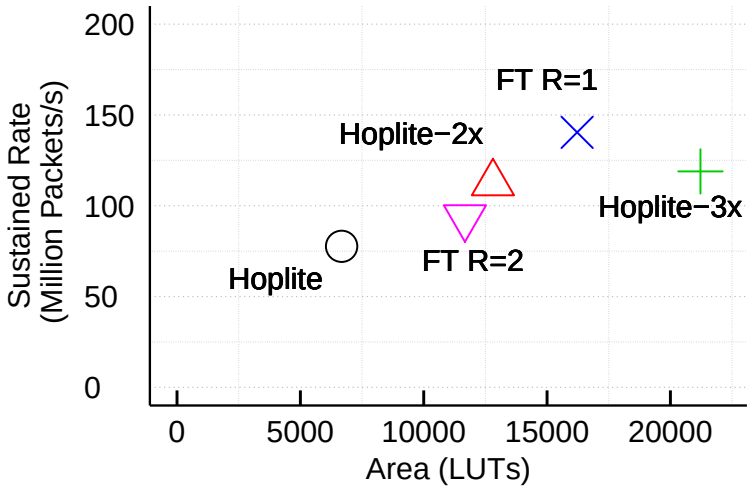
Results – Wiring vs. Throughput 8×8 NoC

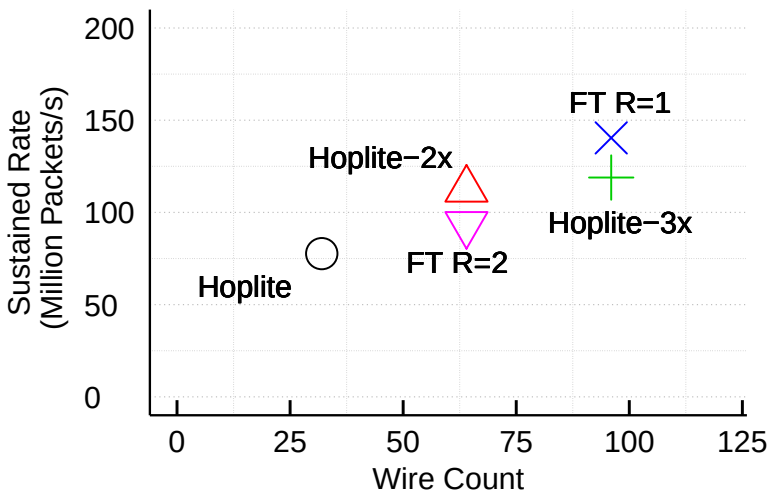


Results – Wiring vs. Throughput 8×8 NoC



Results – Cost vs. Throughput 8×8 NoC





- FastTrack makes better use of FPGA resources (LUTs, and wires)
- Packets are allowed to leave the NoC faster, freeing up resources
- Must pick proper combination of FT design parameters

Qualitative Comparison of FPGA NoC Routers

Router	**Cost**		
	Xbar+Arb	**Buffers**	**VCs**
OpenSMART	✗	✗	✗
BLESS	✗	✓	✓
CONNECT	✗	✗	✗
Split-Merge	✗	✗	✓
Hoplite	✓✓	✓	✓

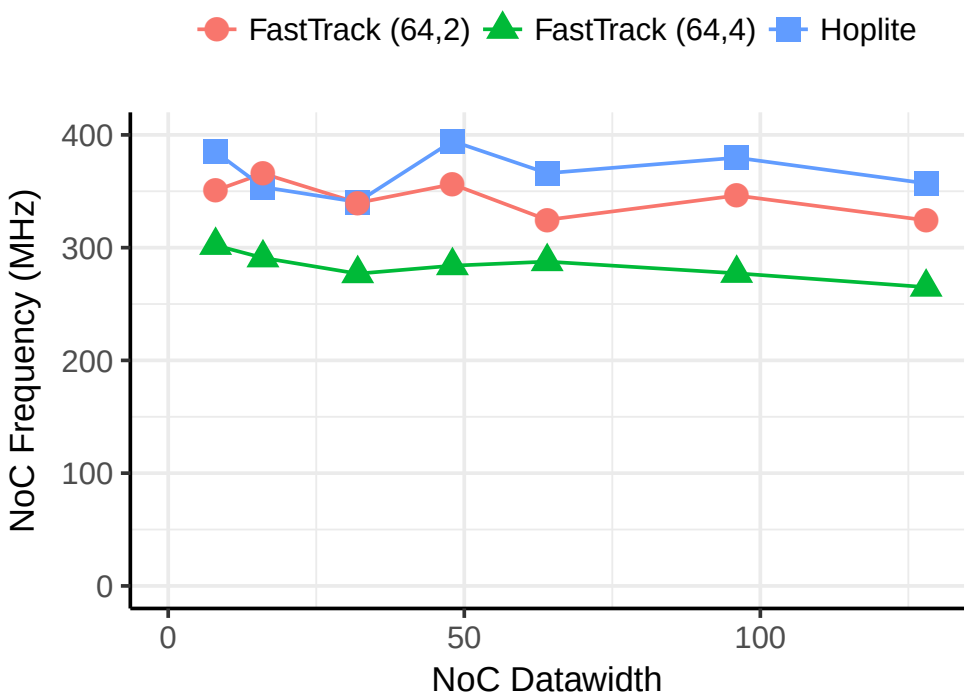
Qualitative Comparison of FPGA NoC Routers

Router	**Cost**			**Perf**	
	Xbar+Arb	**Buffers**	**VCs**	**Tput**	**Latency**
OpenSMART	✗	✗	✗	✓	✓
BLESS	✗	✓	✓	✓	✗
CONNECT	✗	✗	✗	✓	✓
Split-Merge	✗	✗	✓	✓	✓
Hoplite	✓✓	✓	✓	✗	✗

Qualitative Comparison of FPGA NoC Routers

Router	**Cost**			**Perf**	
	Xbar+Arb	**Buffers**	**VCs**	**Tput**	**Latency**
OpenSMART	✗	✗	✗	✓	✓
BLESS	✗	✓	✓	✓	✗
CONNECT	✗	✗	✗	✓	✓
Split-Merge	✗	✗	✓	✓	✓
Hoplite	✓✓	✓	✓	✗	✗
FastTrack	✓	✓	✓	✓	✓

FPGA Mapping Frequency 8×8 NoC



- Calibration studies showed express links can travel quickly on chip
- Fmax for 2-hop FastTrack keeps up with original Hoplite
- 4-hop express link distance too large, some noticeable slowdown

Conclusions

- FastTrack outperforms state-of-the-art Hoplite FPGA NoC by
  - 2.5× for synthetic traffic, 2.8× for real-world traces
  - 2.2× on energy efficiency
  - 2.5× more LUTs required
- FastTrack better at larger system sizes
- Ideal hop distance is 2–4 (4–256 PEs)
- Fmax gap between FastTrack and Hoplite is small