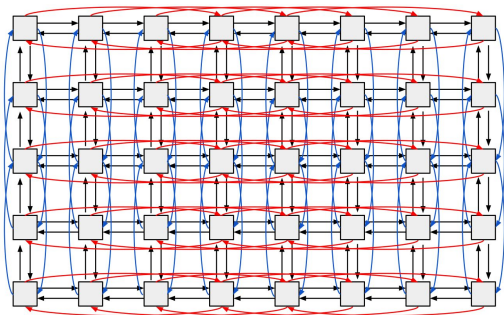


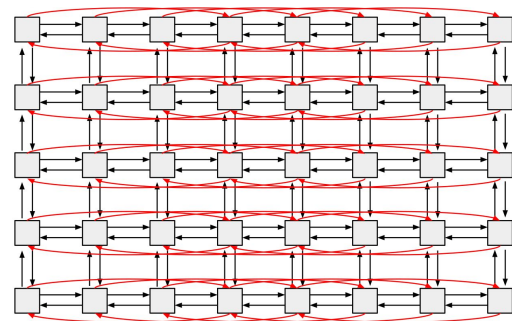
Ruche Networks: Wire-Maximal No-Fuss NoCs

Dai Cheol “Tommy” Jung, Scott Davidson,
Chun Zhao, Dustin Richmond, Michael Bedford Taylor



Bespoke Silicon Group
University of Washington

NOCS 2020



Outline: Ruche Networks

1. Motivation: Physical Design Awareness
2. Ruche Networks
3. Example System: HammerBlade NoC
4. Ruche Networks Are Physically Implementable

Why are NoCs in real chips often just simple meshes?

Many novel NoC topologies with non-local links have been proposed in the literature.

However, in practice, most NoCs taped out in silicon are simple mesh networks.

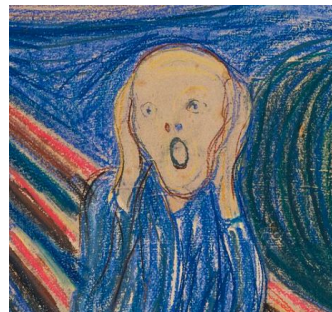
- **MIT Raw** [ISCA, Taylor et al, 2004]
- **TRIPS processor** [ICCD, Gratz et al, 2006]
- **Intel Teraflops** [ISSCC, S.Vangal et al, 2007]
- **TILE64** [ISSCC, S.Bell et al, 2008]
- **Execution Migration Machine** [ICCD, Shim et al, 2013]
- **Epiphany-V** [CoRR, A. Olofsson, 2016]
- **KiloCore** [IEEE JSSC, B.Bohnenstiehl, 2017]
- **OpenPiton** [HPCA, McKeown et al, 2018]
- **Celerity** [IEEE Micro, Davidson et al, 2018]

What are the key properties of 2D mesh network that makes it so popular?

Physical Design Awareness is a first-order Requirement for NoCs

- **Placement of gates and routing wires** are the major steps in physical design (taping out) of a chip. Most chip designers employ **Automatic Place-and-Route** (APR) to perform this task, but the underlying problem is **NP-complete**.
- For 6 mm² design, APR easily takes 36 hours on the fastest Xeon machine with terabytes of RAM, if it succeeds.
 - When it does not succeed, the tool run indefinitely and sometimes crashes. =)
- Running a 600 mm² (GPU-sized) design could easily take over a week.
- In the race to tapeout, taking 6 days to determine if a bug was fixed is unacceptably slow. Even if you are perfect at fixing bugs, you can only fix 60 bugs per year.
 - *You may grow old and die before you tape out the chip.*

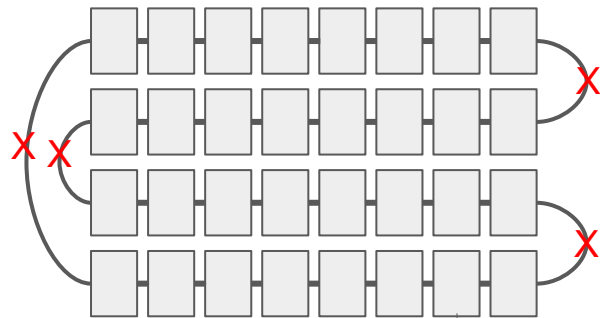
Takeaway: we need to be very conservative about how our NoCs are wired to ensure that APR runs without issues. This is why 2D mesh networks are so common.



How can we structure a design to make Physical Design Easier?

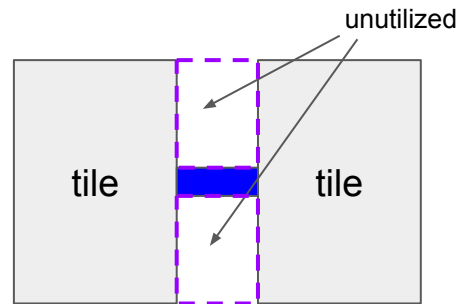
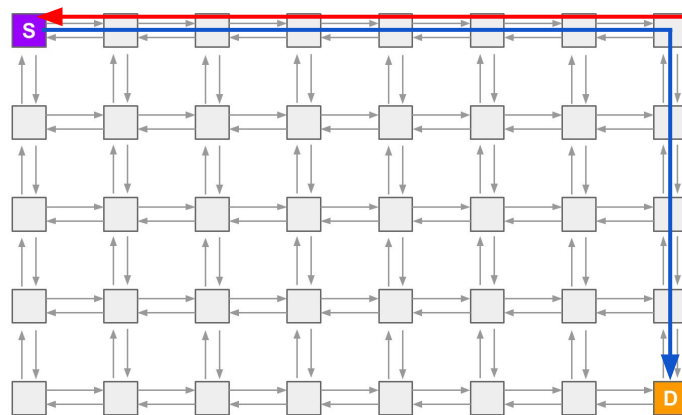
- **Partition the design into many smaller, replicated sub-blocks.**
 - Smaller blocks take a shorter amount of time to perform APR.
 - No asymmetry. Exploit symmetry and reuse.
- **The “Top-level” that instantiates the blocks and wires them together should be simple**
 - Wiring connections for replicated instances should have very similar wire lengths and similar timing deadlines.
 - Short, nearest-neighbor wiring between replicated blocks are ideal.
 - Majority of design complexity should be pushed inside sub-blocks as much as possible.
 - Iterate on complexity using faster APR.

2D mesh topology NoCs are popular because they fulfill these properties by providing simple, regular, local wiring between replicated sub-blocks.



If 2D Mesh Networks are so great for physical design, why are we motivated towards other topologies?

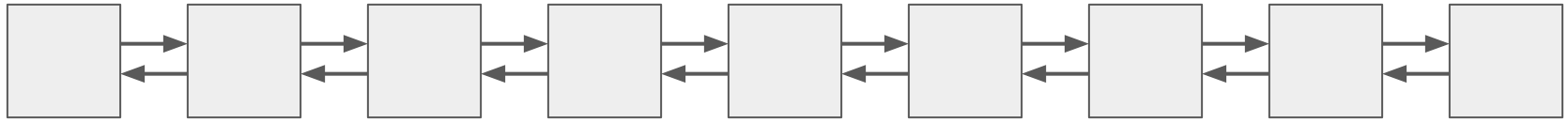
- Packets are limited to **one hop per cycle**, even though **the raw speed of wires** can be 3-5x faster.
- As the number of nodes increases to 512 and beyond:
 - **network diameter** becomes substantial.
 - **bisection bandwidth** becomes a bottleneck. $O(\sqrt{n})$
- Because the router area grows with link width, routing resources are underutilized, both between tiles and inside a tile.



What are Ruche Networks trying to solve?

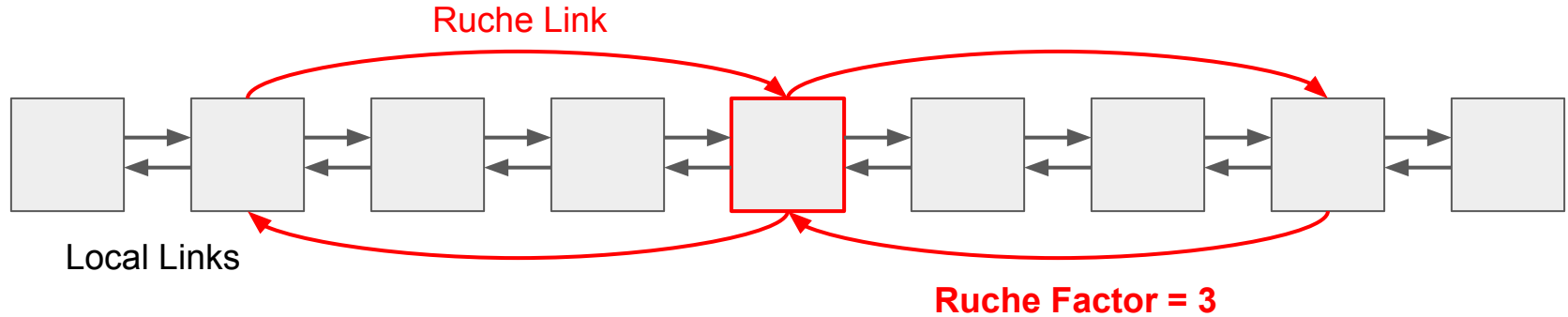
1. A wire-maximal NoC that considers Physical Design as a first class constraint
2. Ability to convert excess wiring into network performance
3. Efficient utilization of excess wires without too much hardware overhead
4. Propagate packets at closer to the full speed of wire
5. Low hardware complexity (critical path + routability)

Traditional Mesh Network (in 1D)



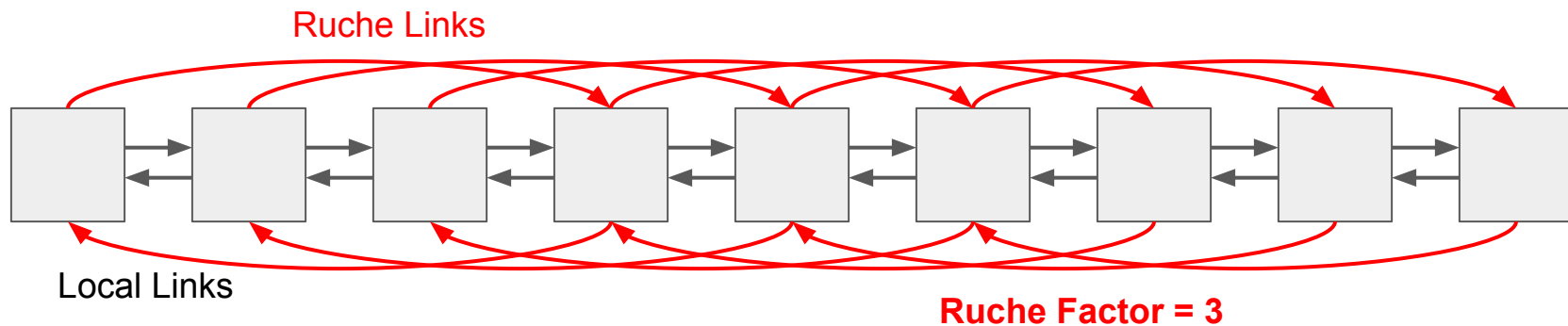
Local Links

Augmenting a Node with Ruche Links



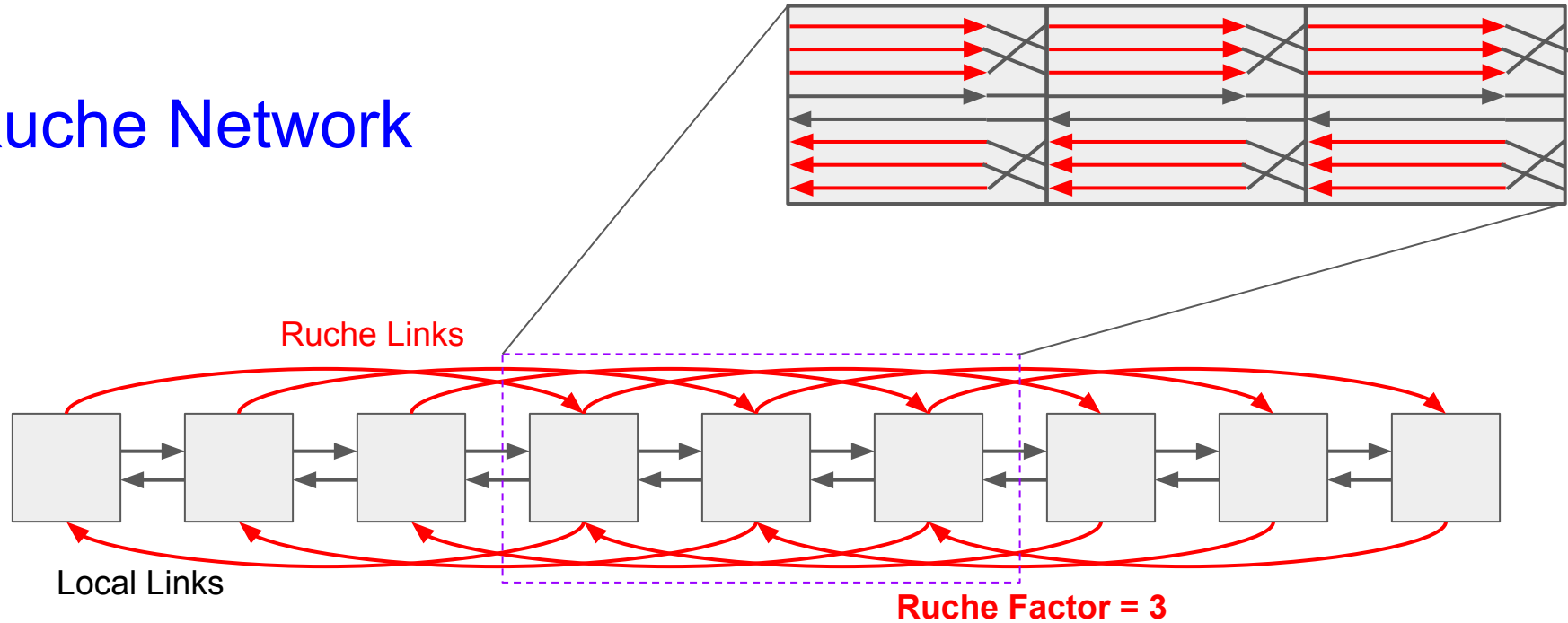
Ruche Factor = # of hops that ruche links extend

Ruche Network



bisection bandwidth: $1x \rightarrow 4x$
network diameter: $8 \rightarrow 4$

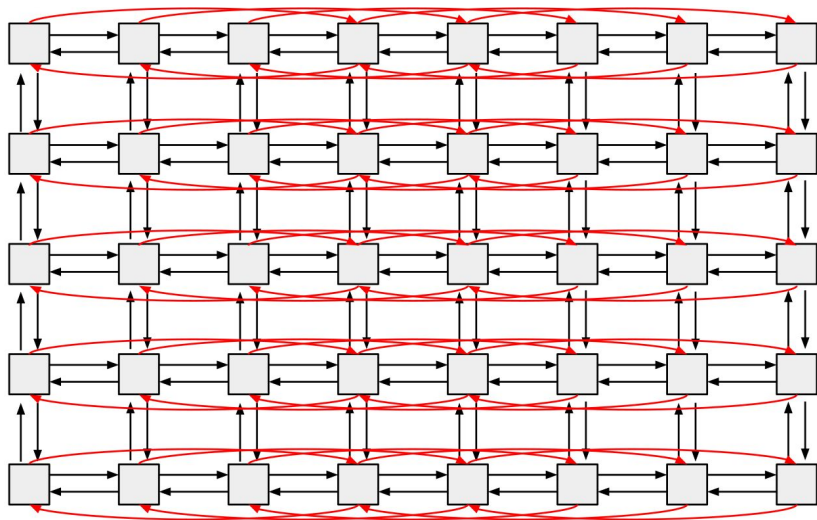
Ruche Network



Each tile in the Ruche network has 100% identical logic and wiring, so the tiles can be replicated and placed directly abutting each other. This addresses the physical design challenge.

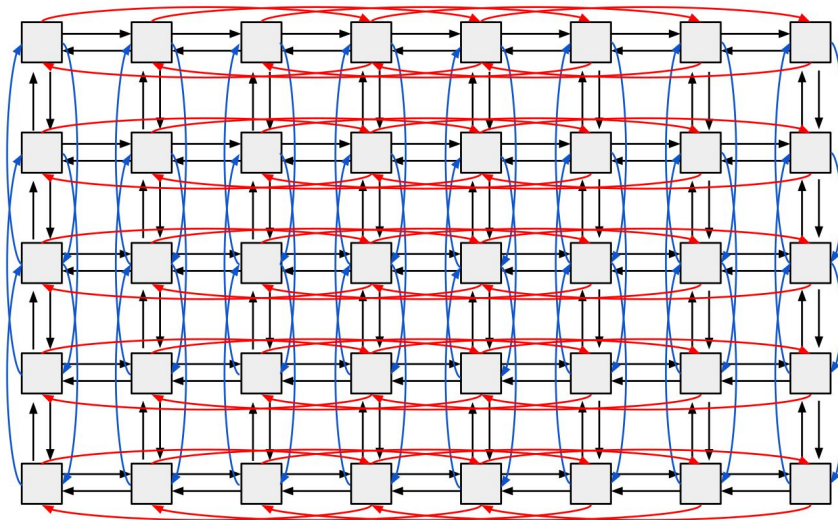
Half Ruche vs Full Ruche

half ruche



Ruche Factor X = 3

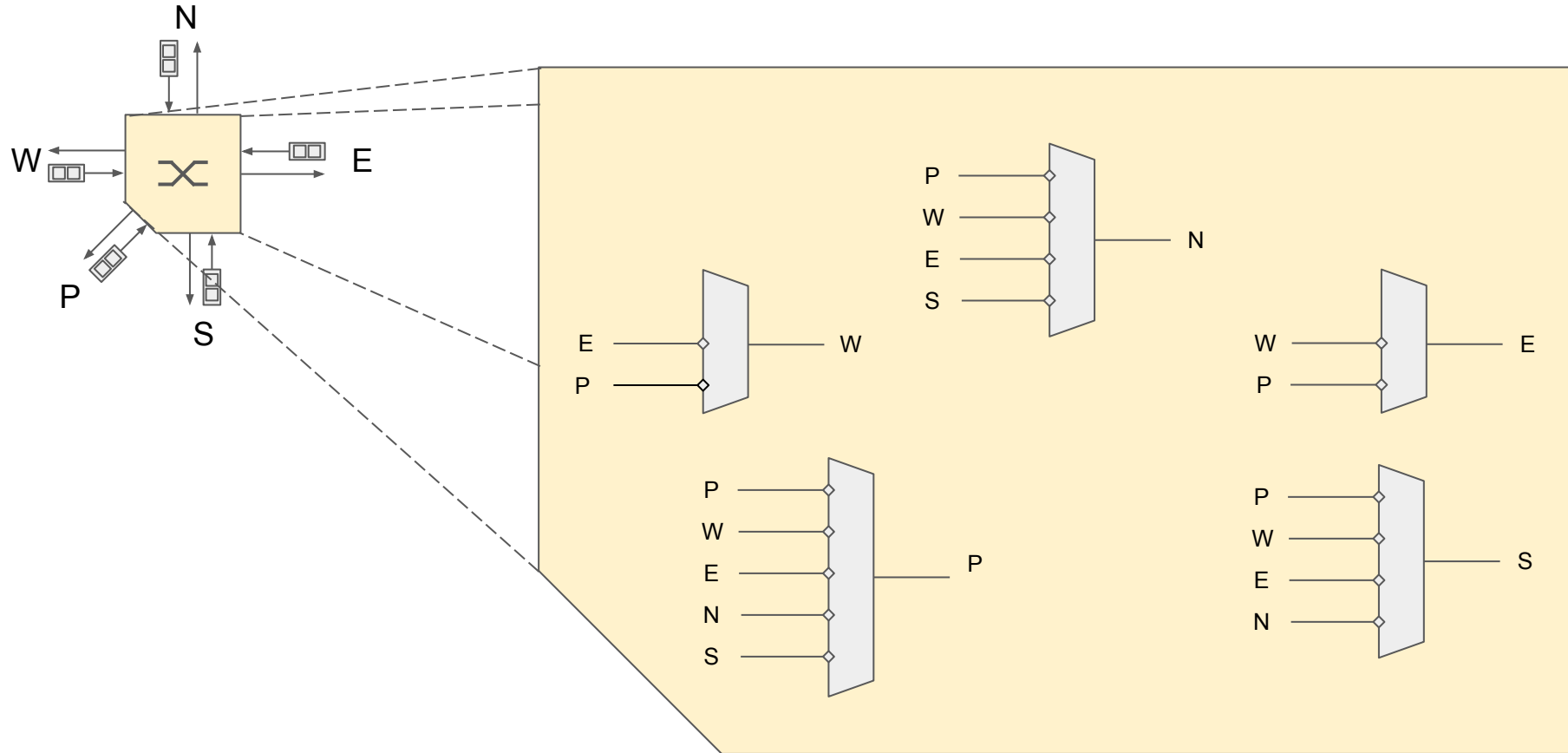
full ruche



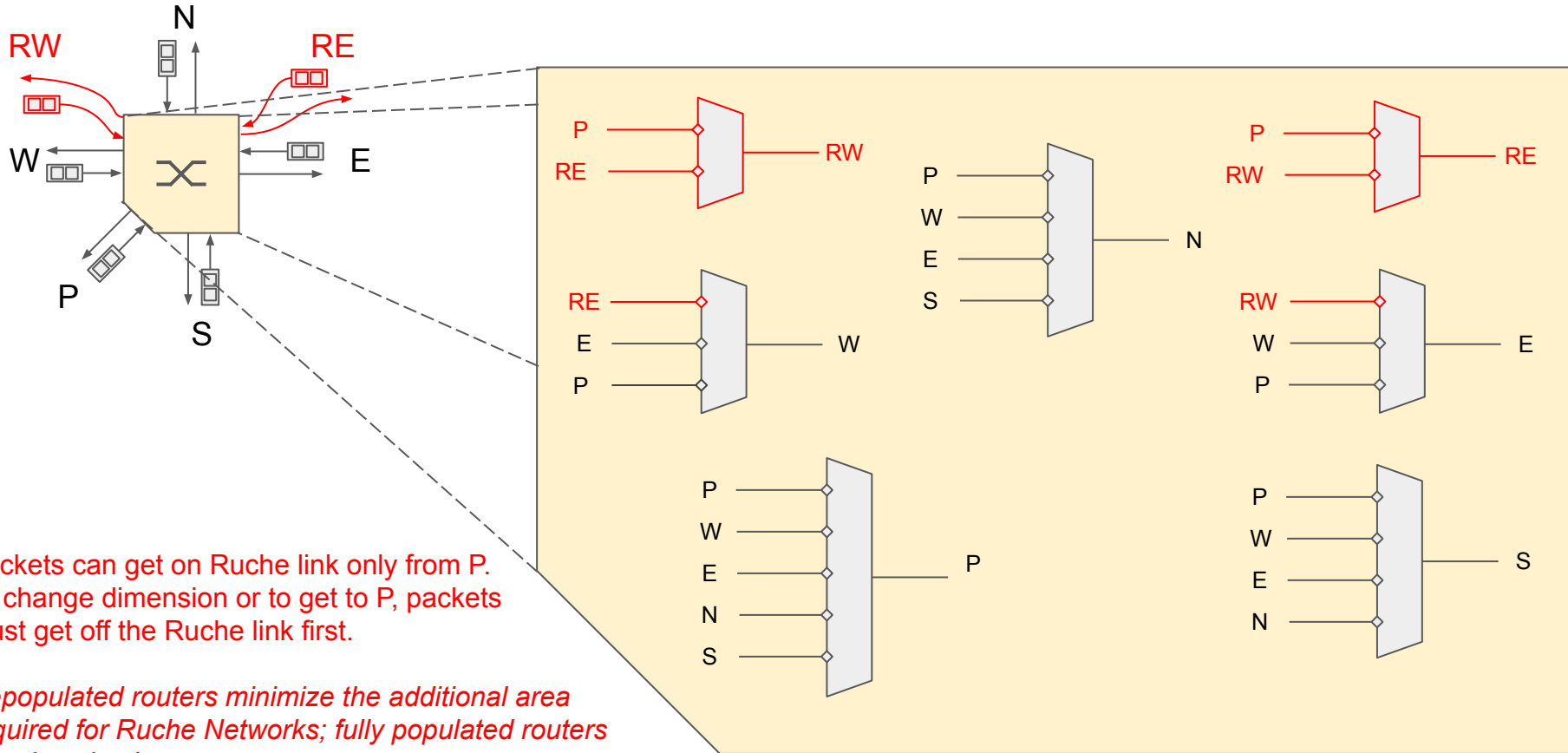
Ruche Factor X = 3

Ruche Factor Y = 2

Dimensioned-Ordered 2D Mesh Router (X then Y)



Half Ruche Depopulated Router (X then Y)



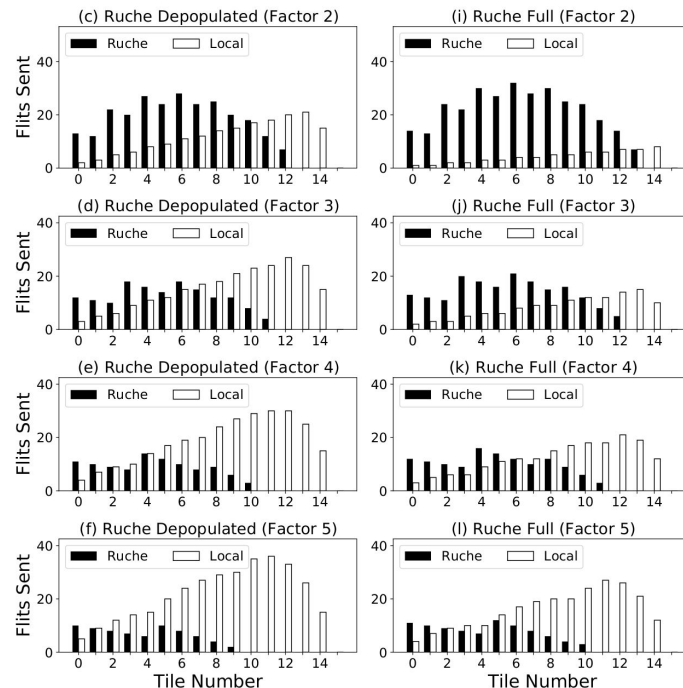
Load Balancing between Ruche and Local

We want to balance the load on ruche and local links to maximize the bandwidth.

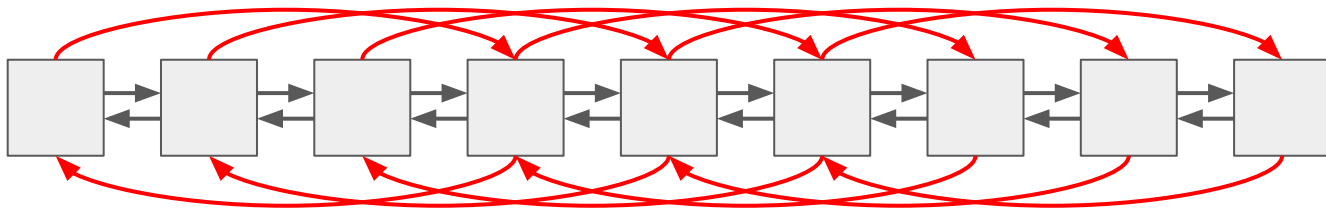
Key Takeaways

- As ruche factor increases, the pressure on local link increases.
- Depopulated / Populated changes the balance between the local and ruche links.
- Interesting questions for follow-on research!

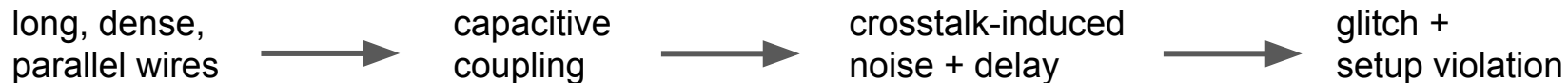
All-to-all unidirectional communication
(16 tiles in 1D network)
(all packets going to East)



Optimizing Signal Integrity



Large Rube Factors may have many closely spaced wires; makes sense to analyse signal integrity



First, let's understand the **physical properties of wires**, and how the signals behave under **different routing configurations**.

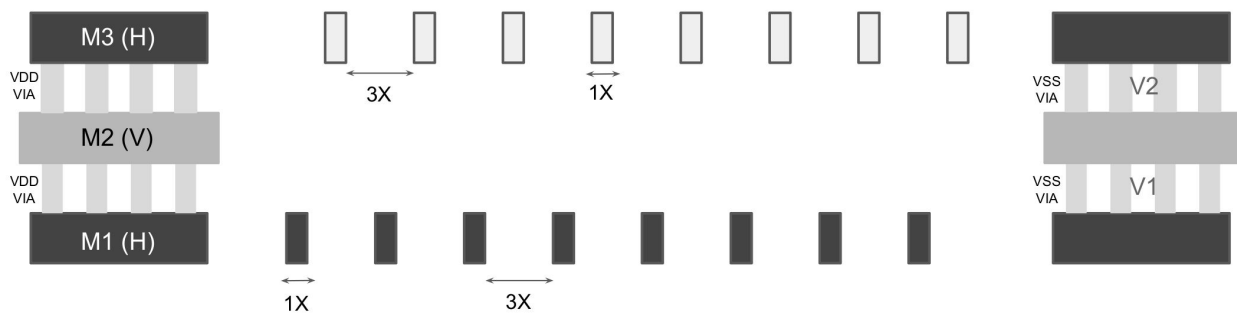
Finding Optimal Wire Spacing

Consider different **spacing** between wires to control **coupling capacitance**.

- single-space = too noisy, too slow
- double-space = ideal for both noise and density
- triple-space = meaningless, because of DRC metal density rule

Try alternating wires on different metal layers.

- Wires are usually thicker than they are wide.
- There are two via layers and one vertical metal layer between two horizontal metal layers.

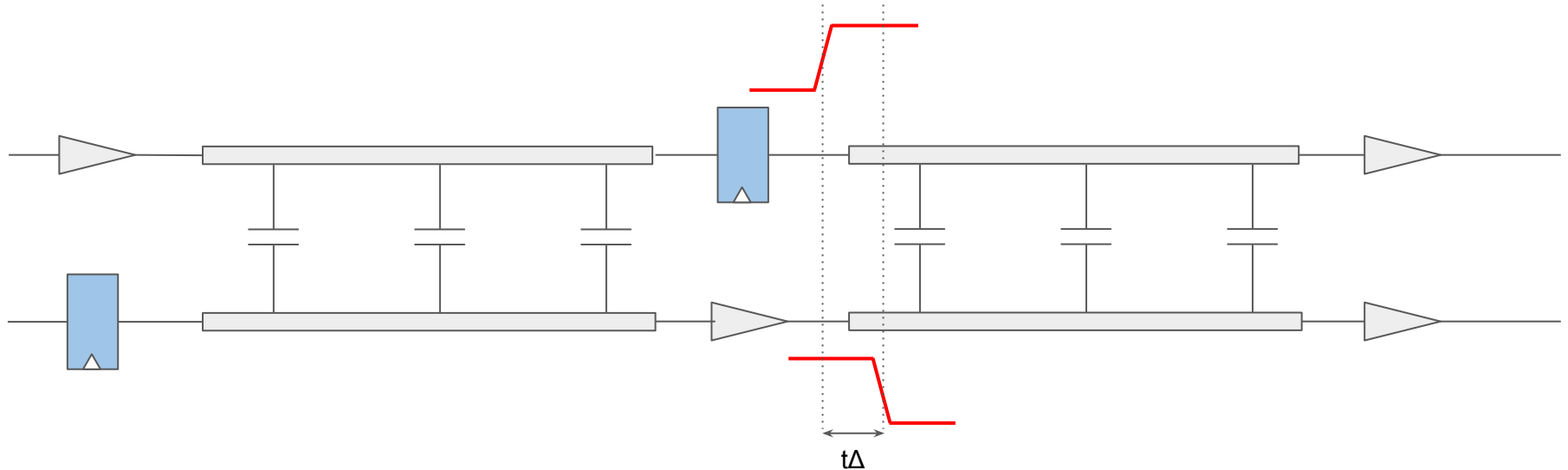


cross-sectional view

Minimizing Miller Effect

Prevent adjacent wires switching at the same time.

Interleave ruche links in different stages so that **signal arrival time** is further apart.



Optimizing Inverter Stages

Try different inverter stages and drive strength

- In terms of wire length,
unbuffered wire delay = **quadratic**
buffered wire delay = **linear**

Inverters with greater drive strength

Less net delay (+)
More resilient to crosstalk (+)

Greater cell area (-)
More gate delay (-)
Dissipate more energy (-)
Noisier on neighbors (-)

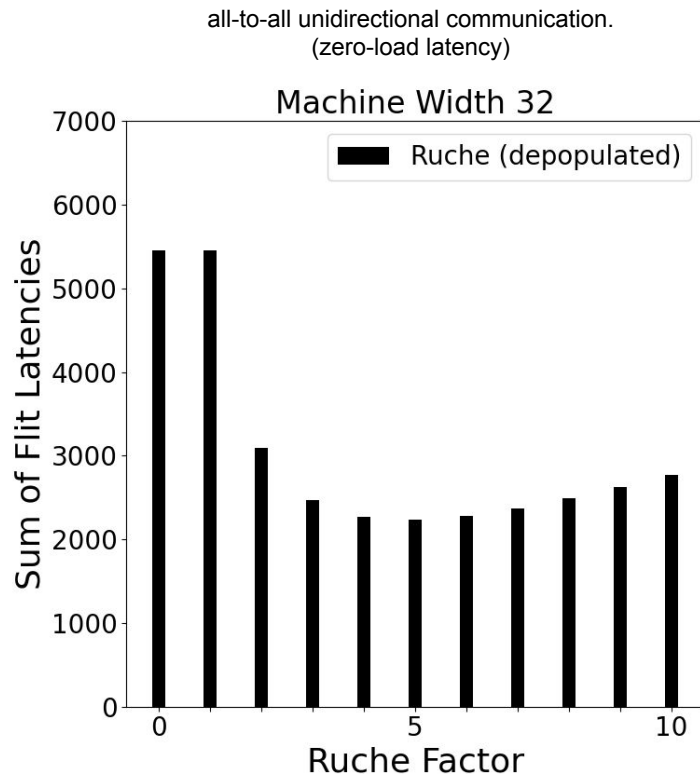
Finding Optimal Ruche Factor

Adjust **Ruche Factor**, based on

- Available routing tracks
- Timing constraints

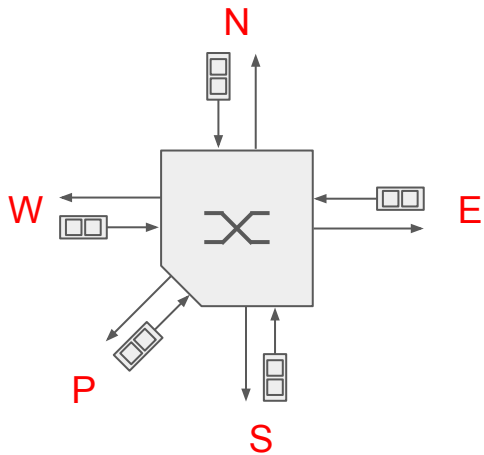
Key Takeaway

- Ruche Factor = 1 does not reduce diameter.
- Most benefits come from 1→2
- Diminishing return as Ruche Factor increases.
- As the ruche factor grow, more endpoints where the packets have to travel via local links.

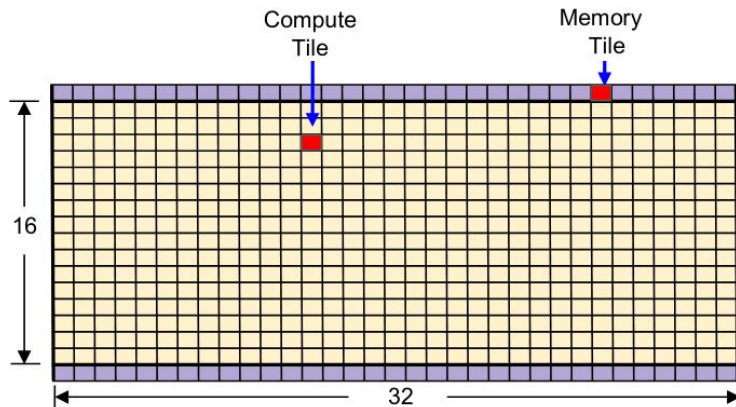


HammerBlade Manycore

- In HammerBlade, there are two mesh networks.
 - Request network is **X-then-Y**
 - Response network is **Y-then-X**.
- HammerBlade has a single-flit, single-cycle per hop packet.
- Arbitration logic is simple round-robin.



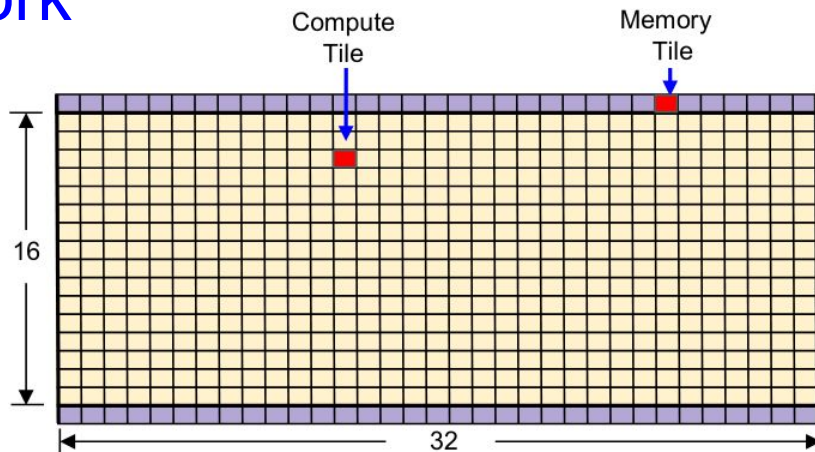
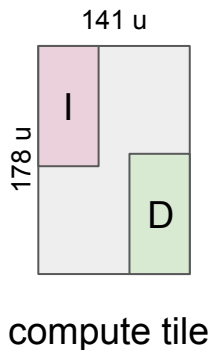
- A Pod is a macro unit that consists of 32x16 compute tiles.
- Top and bottom rows are memory tiles (cache)
- 2:1 aspect ratio for higher memory bandwidth



HammerBlade Pod

HammerBlade + Ruche Network

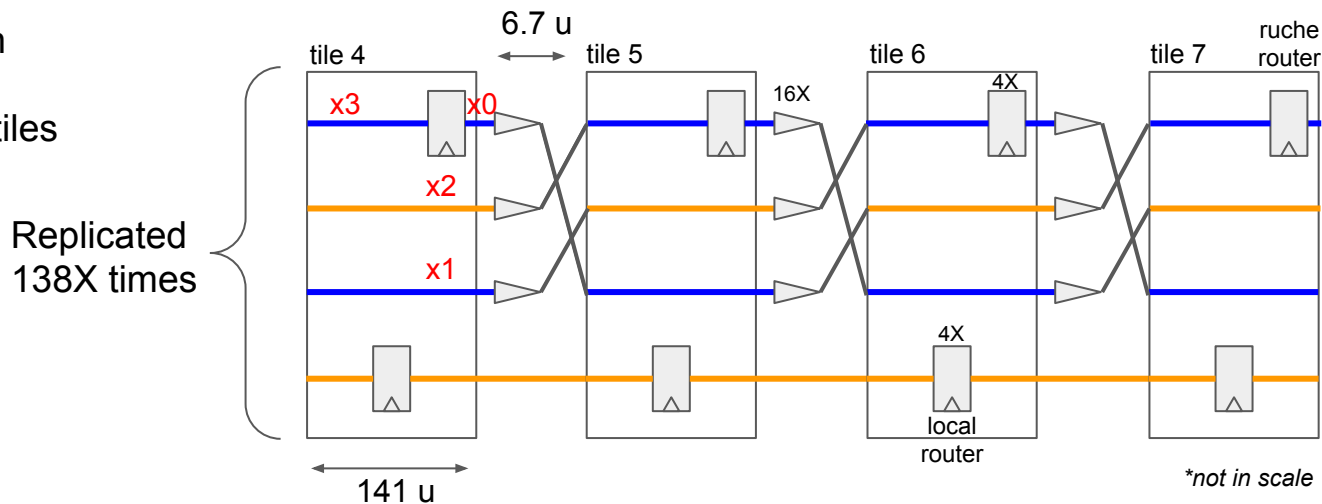
For a given tile dimension,
we can fit a half ruche network in x-direction, with
ruche factor of 3.



	Vanilla Mesh	Half-Ruche (X = 3)
Bisection Bandwidth (request per cycle)	16	64
Network Diameter	46	26
Track Utilization (%)	24	96

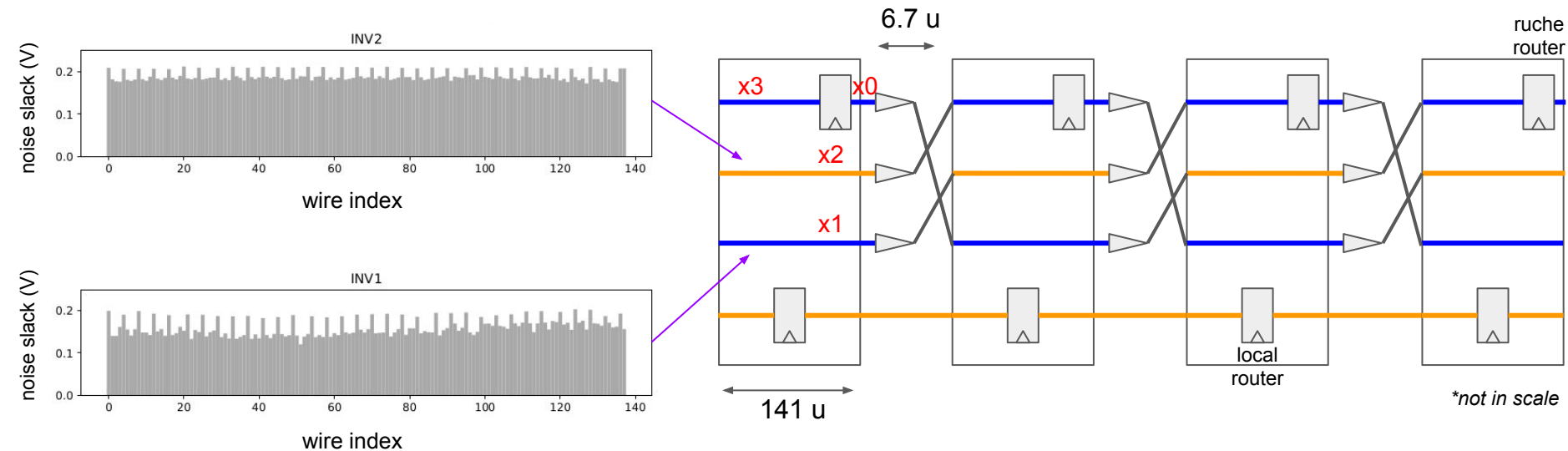
Proof of Concept Model

- Standard-Cell design
- GlobalFoundries 12nm
- Half Ruche in X-direction
- Ruche Factor $X = 3$
- Wire swizzling between tiles



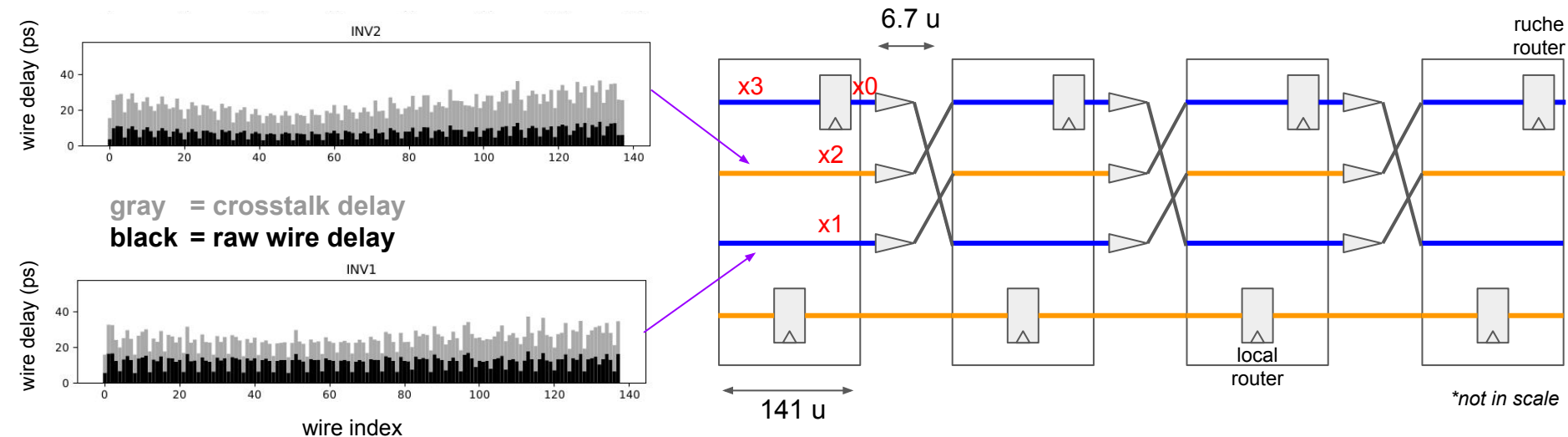
$x3$ and $x1$ are interleaved on the same metal layer to mitigate the Miller effect.

Signal Integrity - Noise Slack



Noise slack does not go below 50% of the noise margin.

Signal Integrity - Wire Delay



In worst-case corner, this meets our **2 GHz** timing constraints with some extra slack.

Summary

1. Physical design is the first-order constraint for NoCs.
2. Ruche Networks is a new kind of *physical design aware* NoC.
3. Ruche networks can better utilize free wiring resources with minimal area increases, and offer >3X greater bandwidth than standard meshes, and >3X lower diameter.
4. We analyze Ruche Networks and show that they can achieve excellent signal integrity with extreme wiring densities.

Acknowledgement

This material is based on research sponsored by Air Force Research Laboratory (AFRL) and Defense Advanced Research Projects Agency (DARPA) under agreement numbers FA8650-18-2-7856 and FA8650-18-2-7852. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of Air Force Research Laboratory (AFRL) and Defense Advanced Research Projects Agency (DARPA) or the U.S. Government.

This work was partially supported by NSF SaTC Award 1563767, NSF SaTC Award 1565446, and by the DARPA/SRC JUMP ADA Center.