A Case for Bufferless Routing in On-Chip Networks

Paper by

Thomas Moscibroda
Microsoft Research

Onur Mutlu
Carnegie Mellon University

ISCA 2009
Austin, Texas

Presented by Bastian Amrhein

Executive Summary

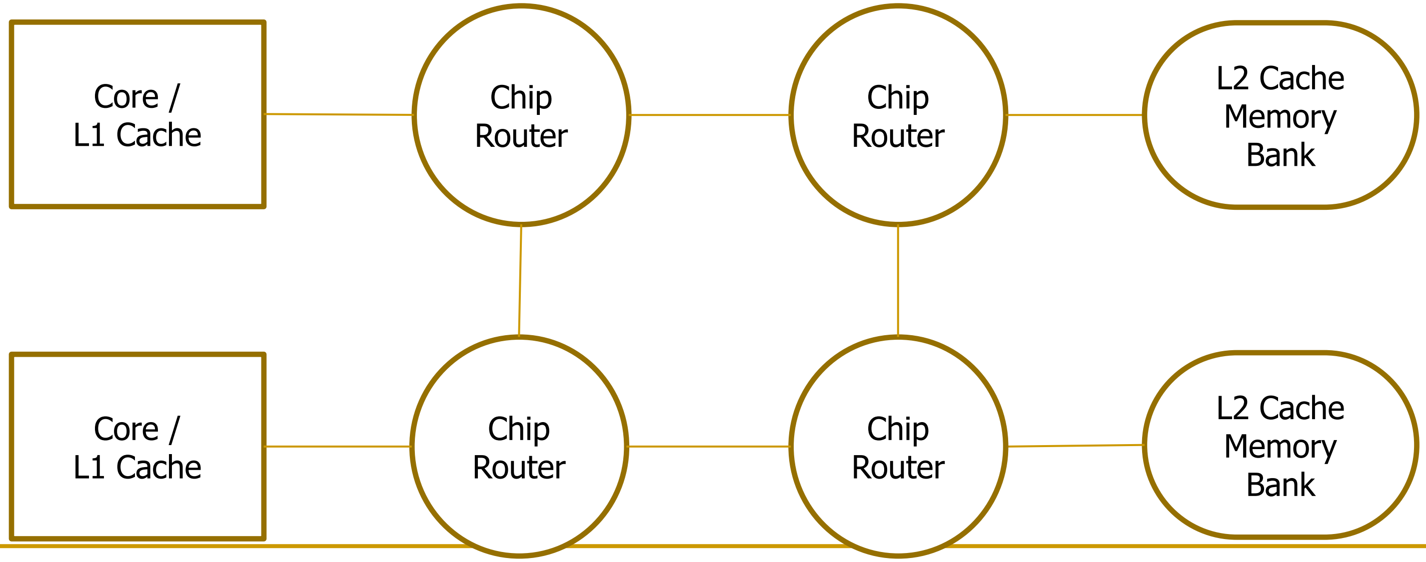
- Problem: Buffers in the routers of a on chip network occupy much physical area and consume a lot of energy. They ensure however time efficient routing.
- Proposal: New routing algorithms that use packet deflection can make buffers obsolete and thereby reduce the cost and power consumption of a on chip network
- Results: There are viable alternatives to buffered routing
  - Area savings of ~ 60%
  - Average energy consumption decrease by 39.4%
  - Average performance decrease by 0.5%
  - Worst-case performance decrease by 3.2%

Section Overview

- What are on-chip networks?
- What the authors propose
- What would be the benefits
- Sample experimental evaluation
- Summary of results & key takeaways
- Strengths/weaknesses of the paper
- Improvements
- Questions & Discussion

What are On-Chip Networks

- Connect cores, caches, memory banks etc....
- Similar to a Computer Network
  - Chip router ~ Network router
  - Core/L1 Cache ~ Host
  - L2 Cache ~ Server
- Every router has a buffer to store incoming data
- 2x2 mesh with 2 cores and 2 L2 caches:



Design Goals of a On-Chip Network

- High throughput
- Low latency
- Fairness
- Low complexity
- Small size
- Low cost
- Low energy consumption
  - Number of cores increases
  - Less heat
- Same metrics as in Computer Networks

The Problem with Buffers

- High power consumption (~40%)
- Large size (~75%)
- Increase latency
- Highly complex

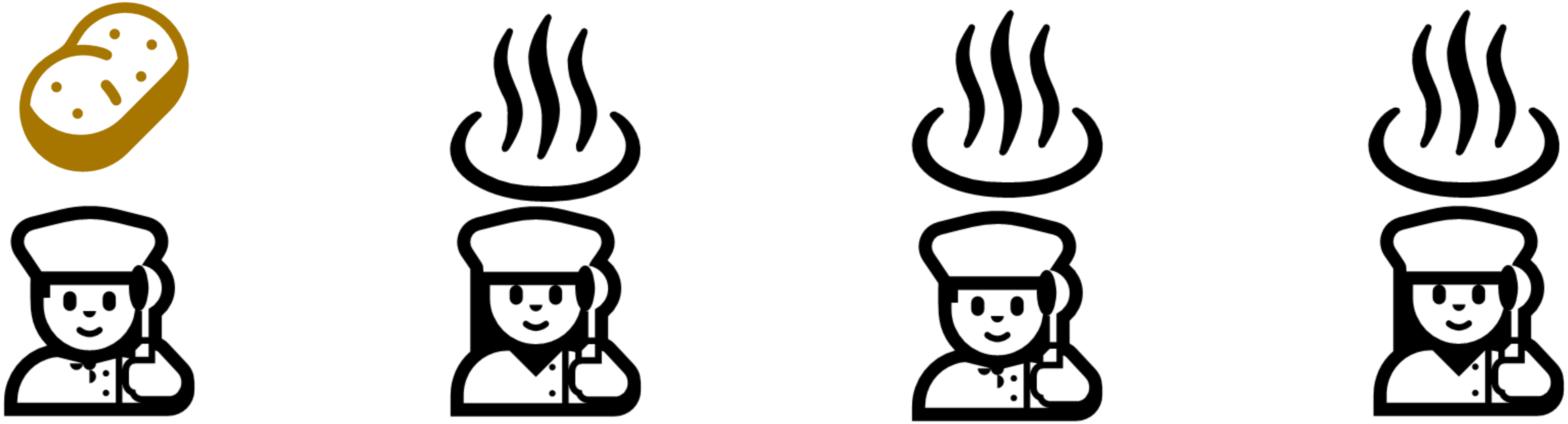
- Can we get rid of buffers?

Section Overview

- What are on-chip networks?
- What the authors propose
- What would be the benefits
- Sample experimental evaluation
- Summary of results & key takeaways
- Strengths/weaknesses of the paper
- Improvements
- Questions & Discussion

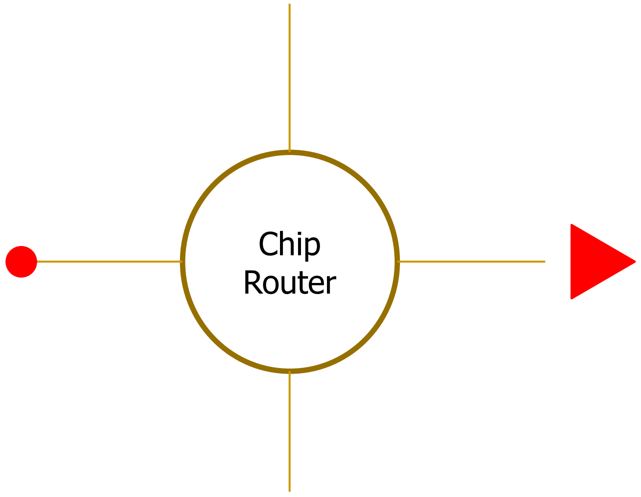
Key Approach

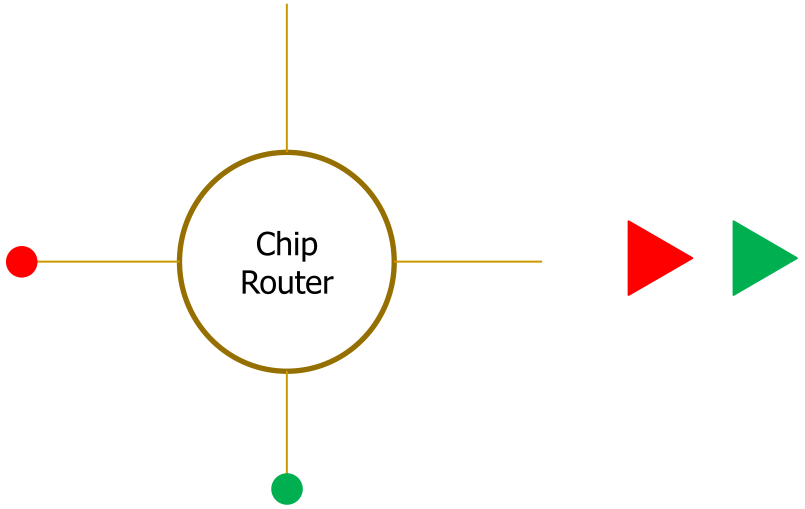
- Instead of buffer routing, use hot potato routing
- Always route a packet
- The links in the system act as the new buffer
- If only one potato/data packet is present:
  - Same latency as with buffers
  - Same power consumption



How does this actually work?

- Flit Bufferless Routing (Flit Bless)
- Split data packet into flits (flow control units)
- Reroute flit to best link
- If no good link is available, flit is deflected
- Multiple flits are routed based on their age -> no deadlock or livelock

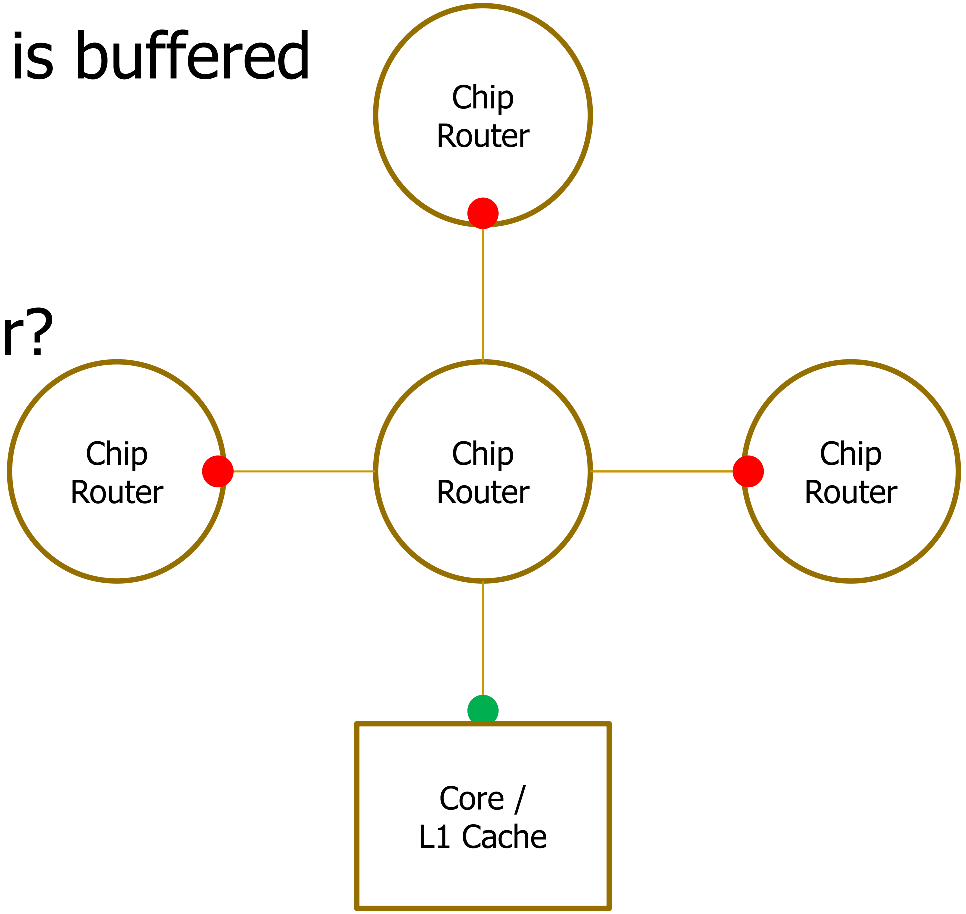




Bufferless Routing is not Bufferless

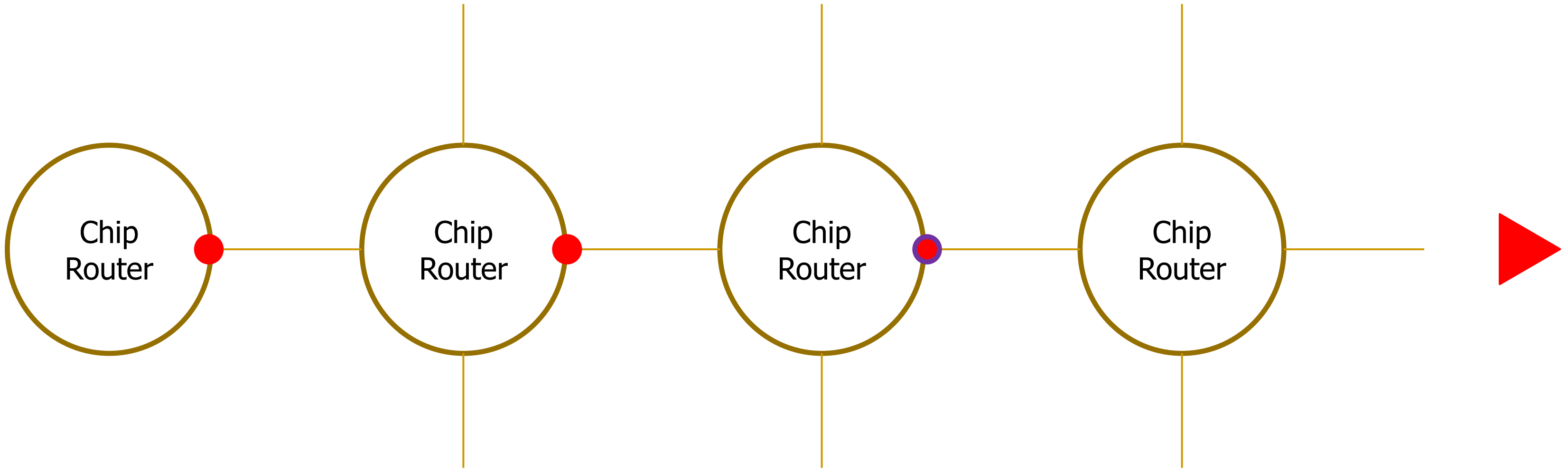
- Buffers are needed to prevent data loss
- Injection routers have more input ports, than output ports
- The youngest flit is buffered
- Self-throttling

- Can we get better?



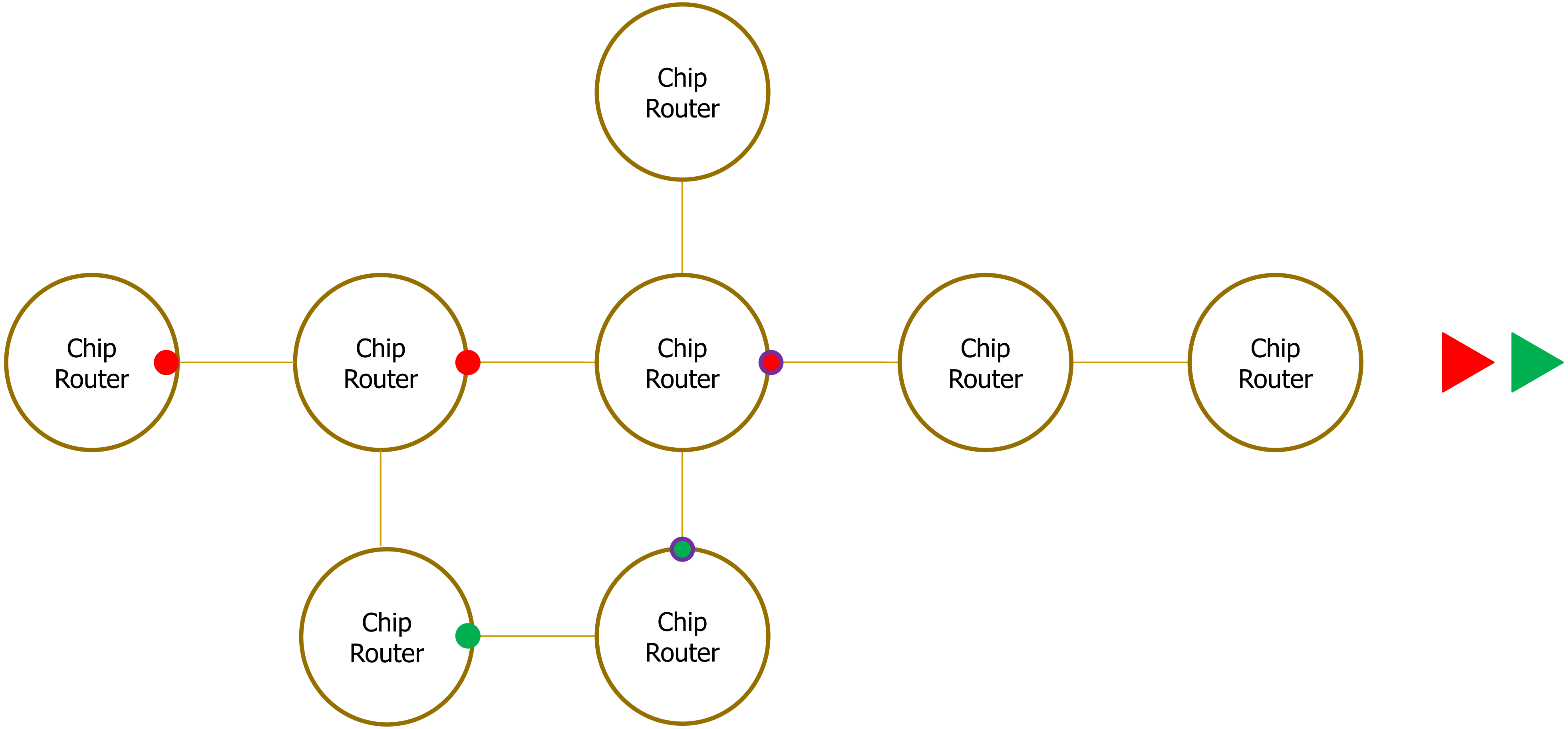
Worm Bless

- A worm is a column of flits following the Head flit
- All flits could arrive together as one worm
- Head flit stores all necessary data to forward worm to next router
- Decrease computation needed for each flit



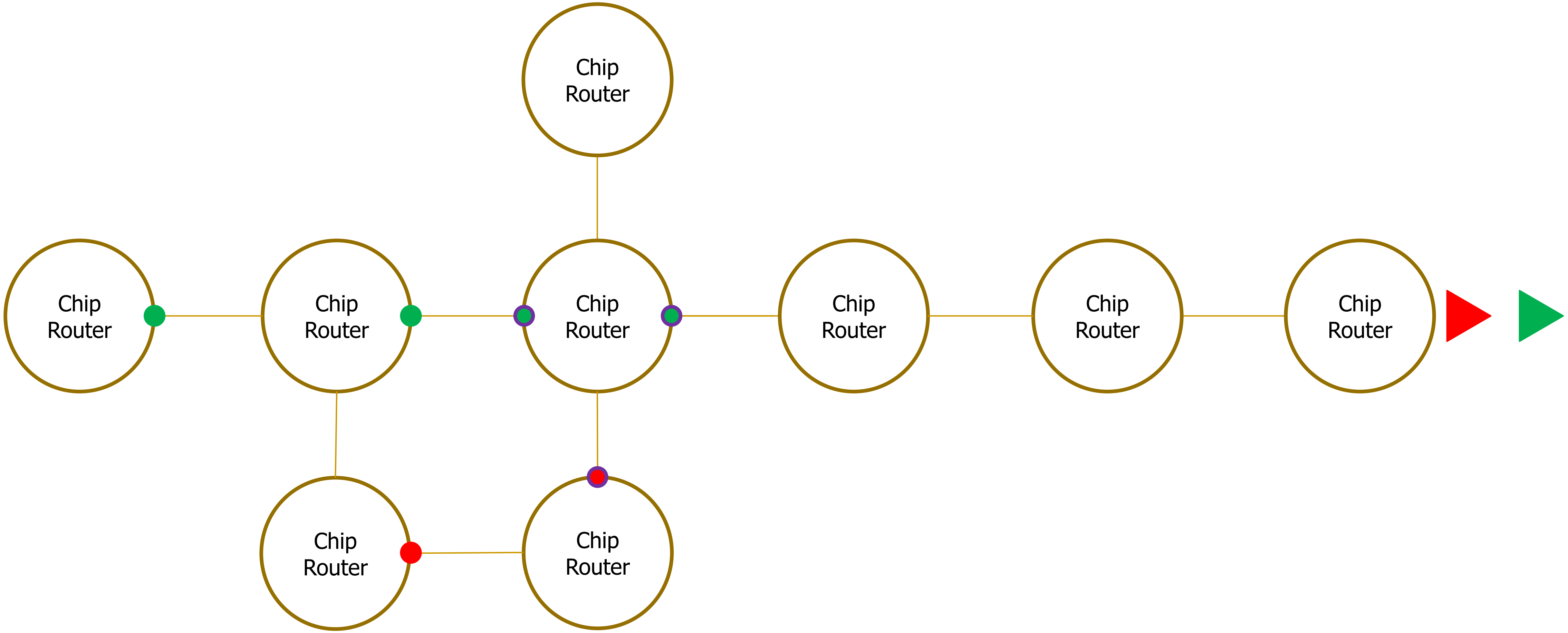
Worm deflection

- What if a younger worm arrives after an older worm



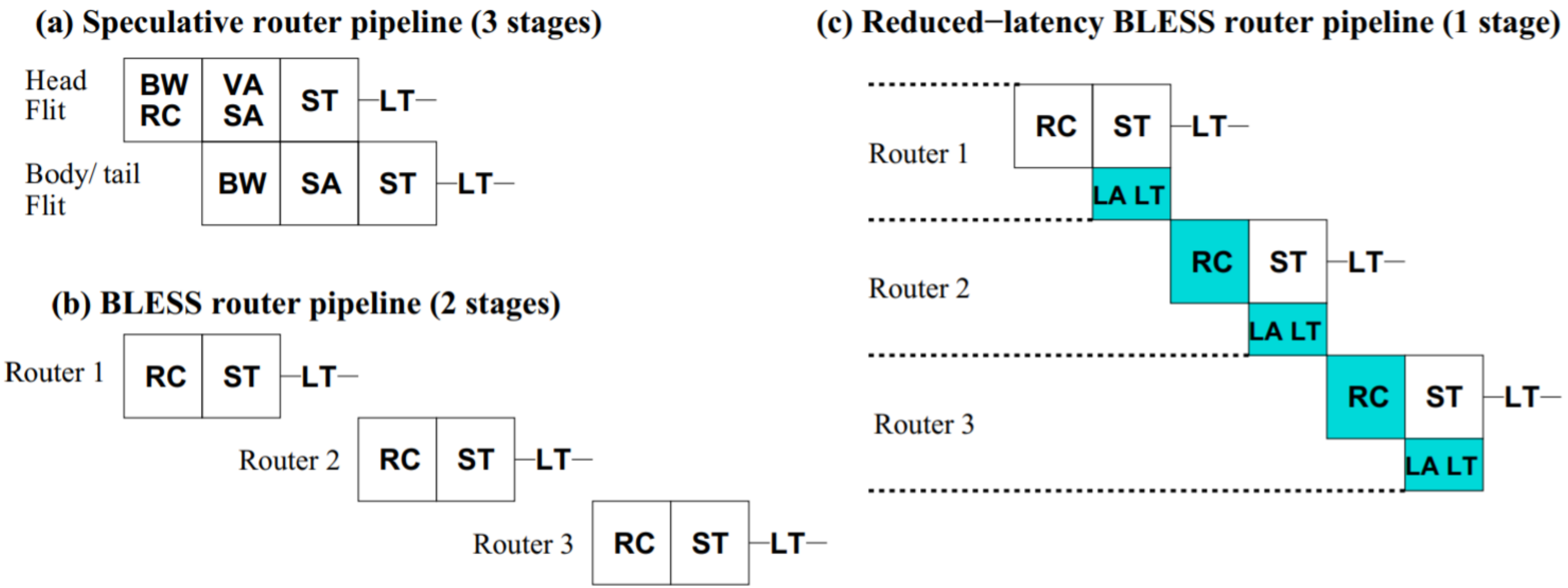
Worm truncation

- What if an older worm arrives after a younger worm
- No deadlocks or livelocks



Routing Mechanism

- Throughput remains the same
- Latency can be improved
- More circuitry with information present.



Section Overview

- What are on-chip networks?
- What the authors propose
- What would be the benefits?
- Sample experimental evaluation
- Summary of results & key takeaways
- Strengths/weaknesses of the paper
- Improvements
- Questions & Discussion

Advantages of bufferless routing

- No routing buffers
- Simpel control flow
- Low router latency
- No dead- or livelocks
- Adaptivity

Disadvantages of bufferless routing

- Increased overall latency, because of deflection
- Lower saturation throughput
- Reduced bandwith
- Increased buffers in the receiver
- Increased link width and power consumption

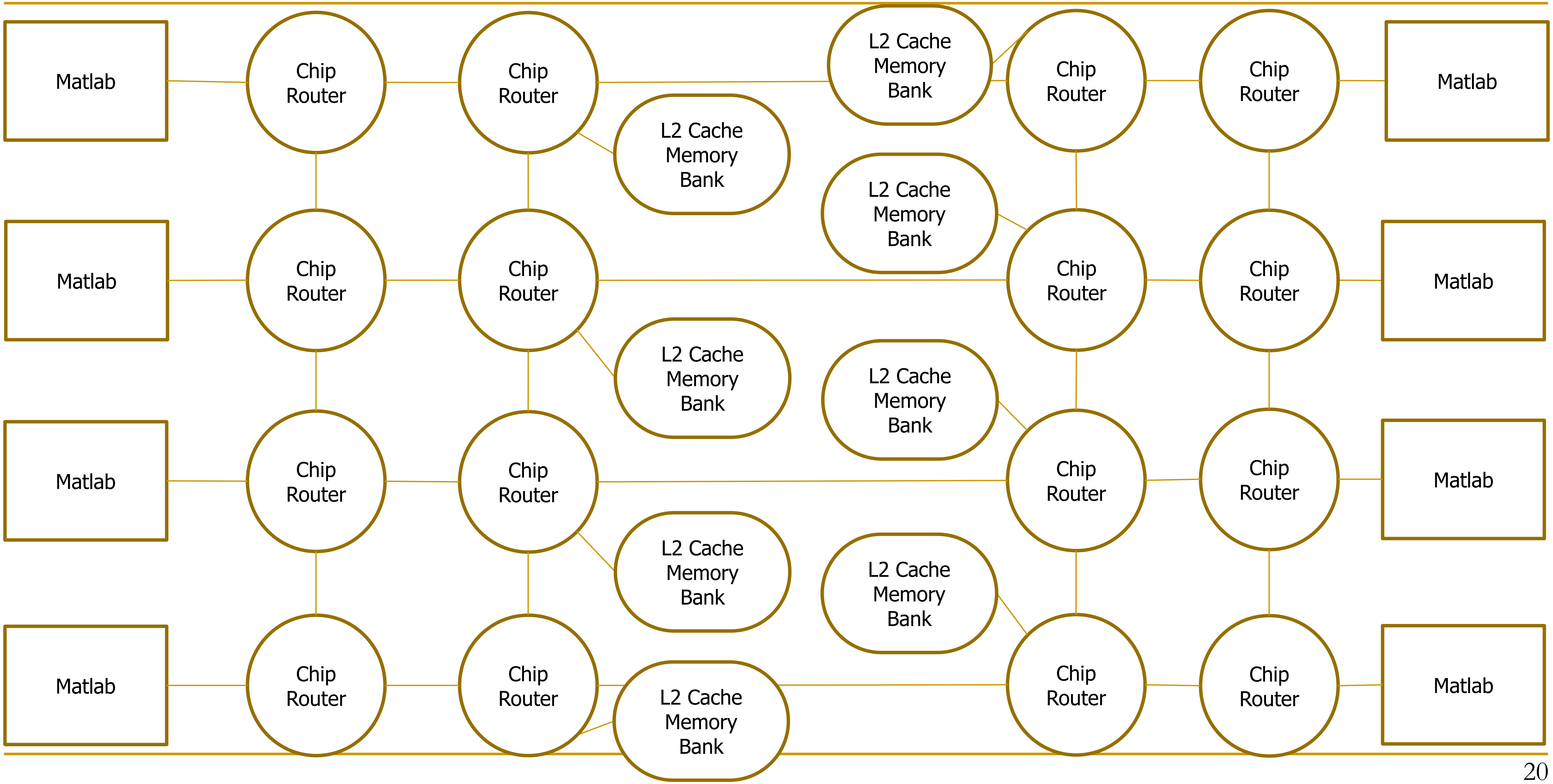
Section Overview

- What are on-chip networks?
- What the authors propose
- What would be the benefits
- Sample experimental evaluation
- Summary of results & key takeaways
- Strengths/weaknesses of the paper
- Improvements
- Questions & Discussion

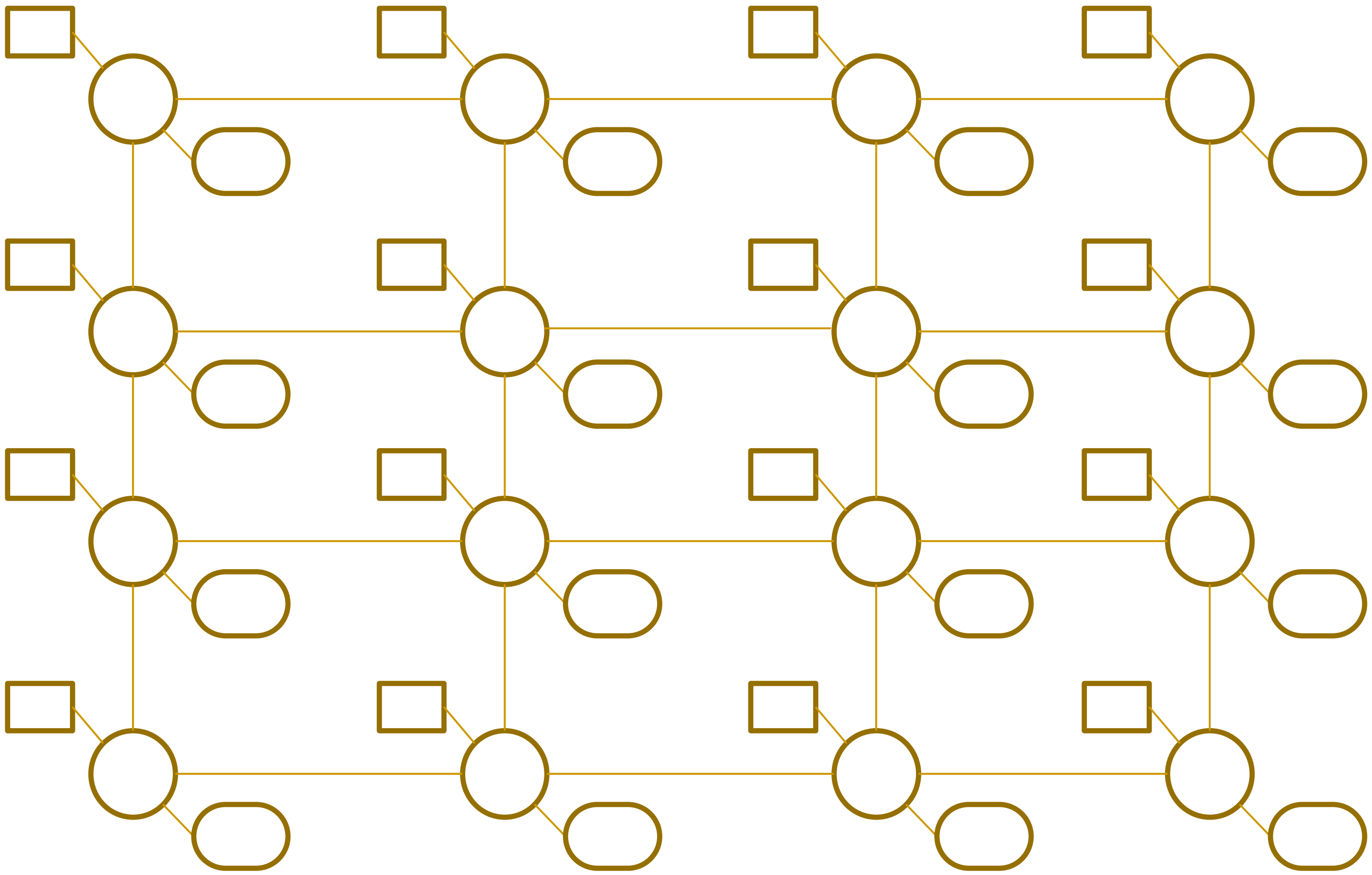
Experimental Evaluation

- Different chip-systems, all meshes
  - 4x4 with 8 Cores and 8 L2 Caches, sparse
  - 4x4 with 16 Cores and 16 L2 Caches, dense
  - 8x8 with 16 Cores and 64 L2 Caches, sparse
- Different workloads
  - Homogeneous, high propability of simultaneous accesses
  - Heterogeneous, more realistic
  - Applications & network intensity
    - Matlab, heavy
    - Milc, medium (=physical benchmark)
    - H264ref, low (=video encoder benchmark)
  - 2006 CPU Benchmark
  - Parallel applications?

Example Set Up: 4x4 8 Cores Homogeneous

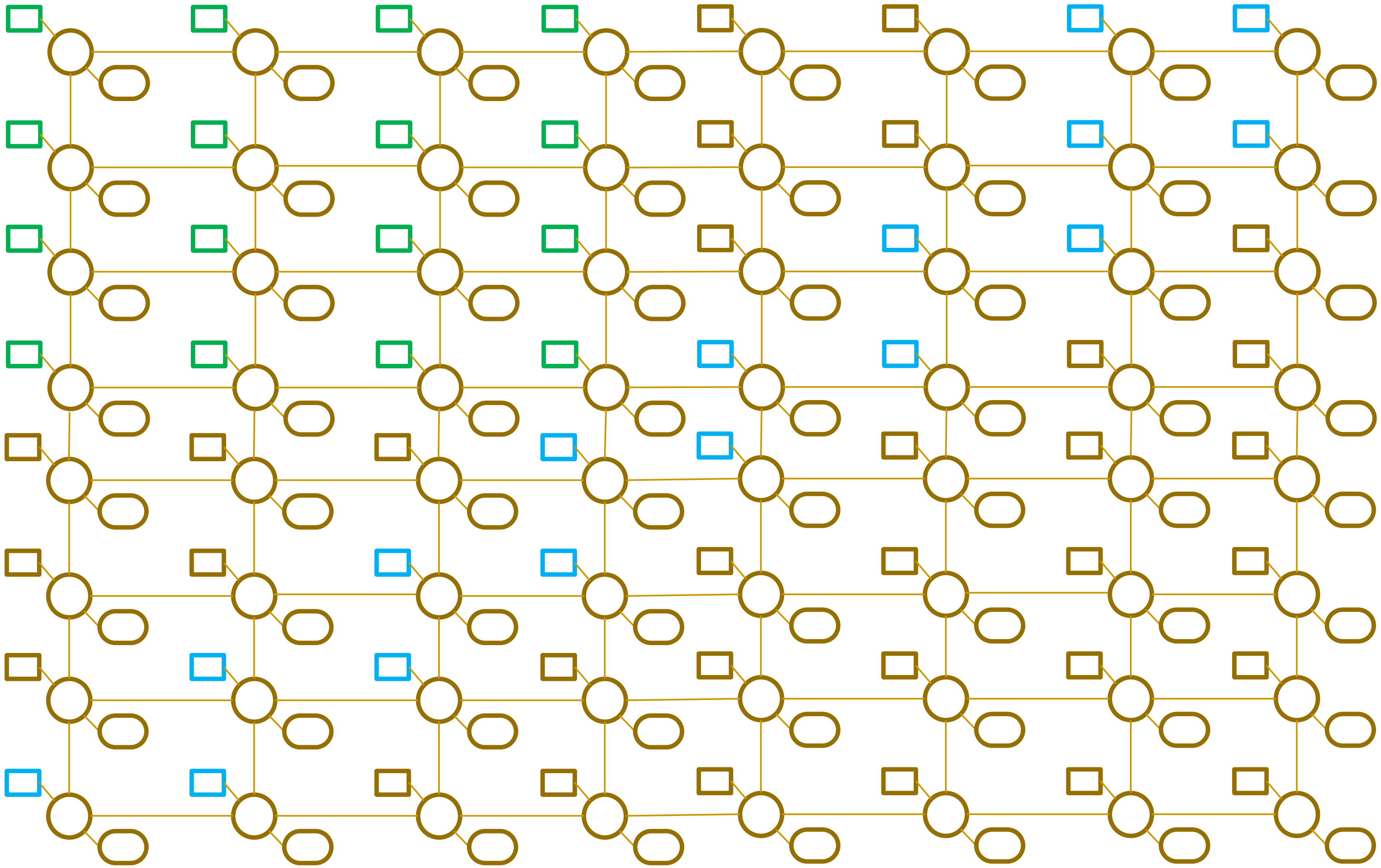


Example Set Up: 4x4 16 Cores



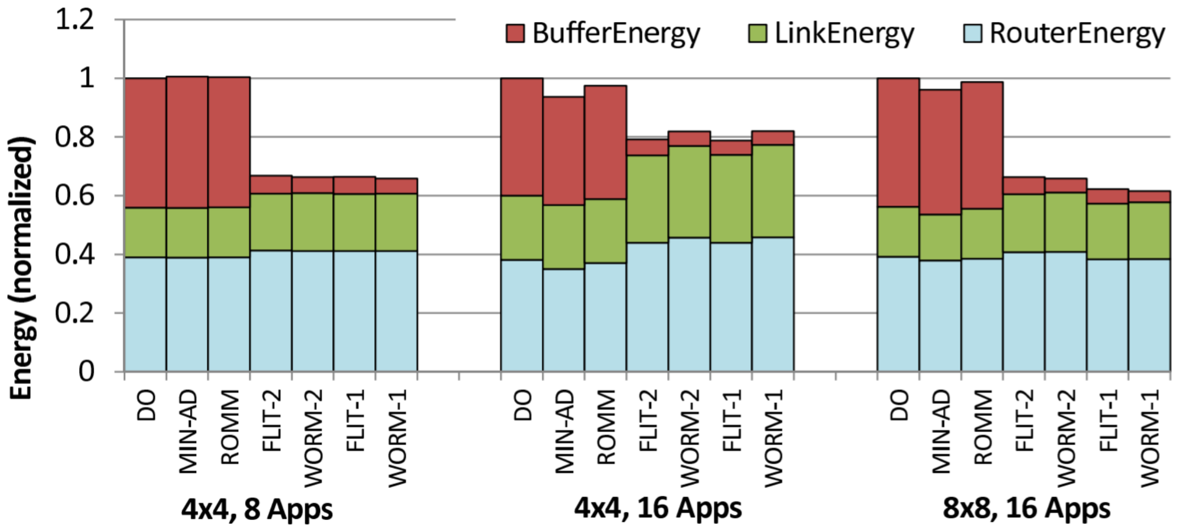
Example Set Up: 8x8 16 Cores

- 16 Cores and 64 L2 Caches



Energy consumption

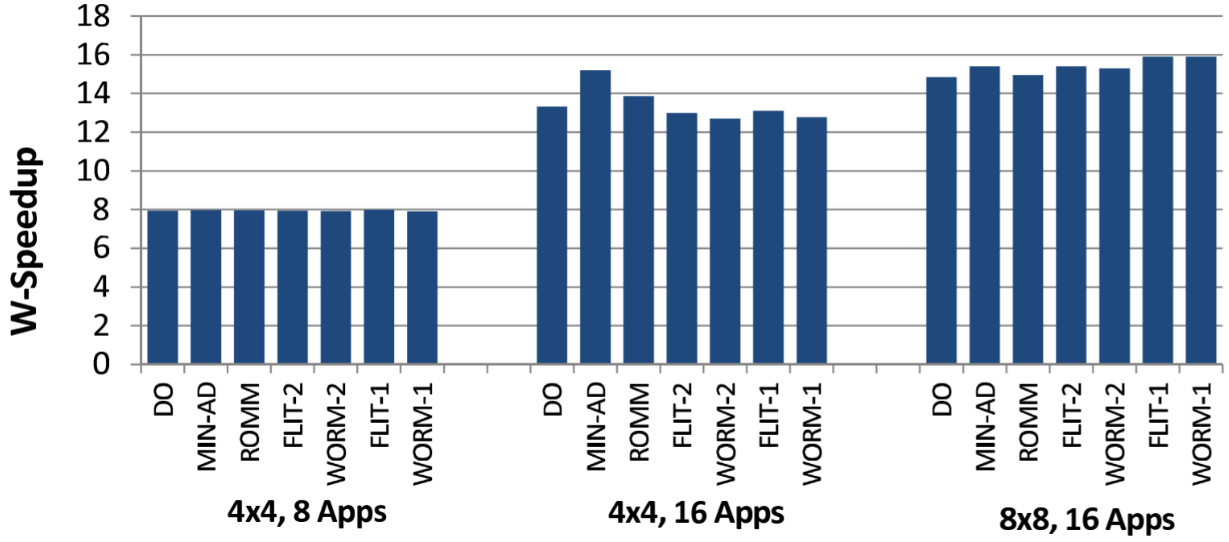
- Homogeneous
- Matlab
- Perfect L2 Caches
- DO, MIN-AD & ROMM with Buffers
- Increase in linkenergy & routerenergy
- Decrease in bufferenergy



Speedup

- Homogeneous
- Matlab
- Perfect L2 Caches
- DO, MIN-AD & ROMM with Buffers
- IPC alone with buffers = IPC alone without buffers
- Increase in sparse network & decrease in dense network

$$WeightedSpeedup = \sum_i IPC_i^{shared} / IPC_i^{alone}$$



Section Overview

- What are on-chip networks?
- What the authors propose
- What would be the benefits
- Sample experimental evaluation
- Summary of results & key takeaways
- Strengths/weaknesses of the paper
- Improvements
- Questions & Discussion

Key Results

- Area reduction by 60.4% because of routing buffers
- Area of links increased by 18.75%
- Lower saturation throughput
- Config 1 (sparse) & Config 2 (dense)

Network Config 1	Perfect L2		Realistic L2	
	Average	Worst-Case	Average	Worst-Case
Δ Network Energy	-39.4%	-28.1%	-46.4%	-41.0%
Δ System Performance	-0.5%	-3.2%	-0.15%	-0.55%

Network Config 2	Perfect L2		Realistic L2	
	Average	Worst-Case	Average	Worst-Case
Δ Network Energy	-32.8%	-14.0%	-42.5%	-33.7%
Δ System Performance	-3.57%	-17.1%	-0.65%	-1.53%

Takeaways

- Bufferless routing can be usefull
- A network is not like every other network

Further Research

- Congestion Controll
  - "On-chip networks from a networking perspective: congestion and scalability in many-core interconnects" by George Nychis, Chris Fallin, Thomas Moscibroda & Onur Mutlu, published in ACM SIGCOMM Computer Communication in 2012

- Router
  - "CHIPPER: A low-complexity bufferless deflection router" by Chris Fallin, Chris Craik & Onur Mutlu, published in: 2011 IEEE 17th International Symposium on High Performance Computer Architecture

Section Overview

- What are on-chip networks?
- What the authors propose
- What would be the benefits
- Sample experimental evaluation
- Summary of results & key takeaways
- Strengths/weaknesses of the paper
- Improvements
- Questions & Discussion

Strengths

- Conclusions are clear
- Different benchmarks
- Intuitive idea
- Algorithms are well explained step by step
- Good base for further research up to this day

Weakness

- Knowledge prerequisite
- Experimental evaluation is confusing
  - What?
  - Why?
  - How?
  - Results are clear however
- Parallel applications
  - Divide & Conquer: Sum over list
    - All cores sum up their parts of the list
    - Send result all to the same cache bank ⇒ Congestion
- No acknowledgement of further directions of expansion

Section Overview

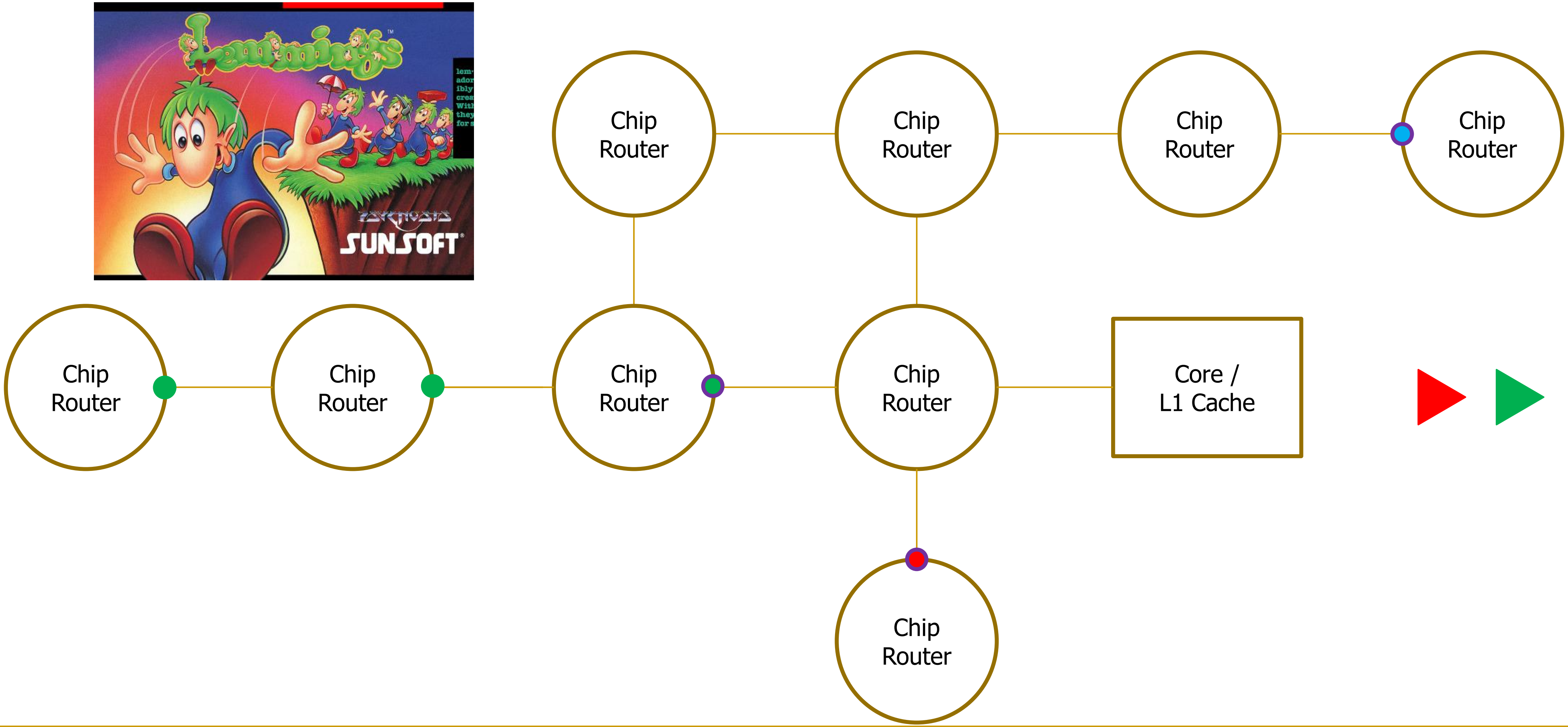
- What are on-chip networks?
- What the authors propose
- What would be the benefits
- Sample experimental evaluation
- Summary of results & key takeaways
- Strengths/weaknesses of the paper
- Improvements
- Questions & Discussion

Improvements

- Worm length?
  - All flits follow one Head flit
  - If deflected, all flits follow
- Network type?
  - Mesh
  - Torus
  - Something completey different
  - Higher Area, higher connectivity
  - Saturation throughput

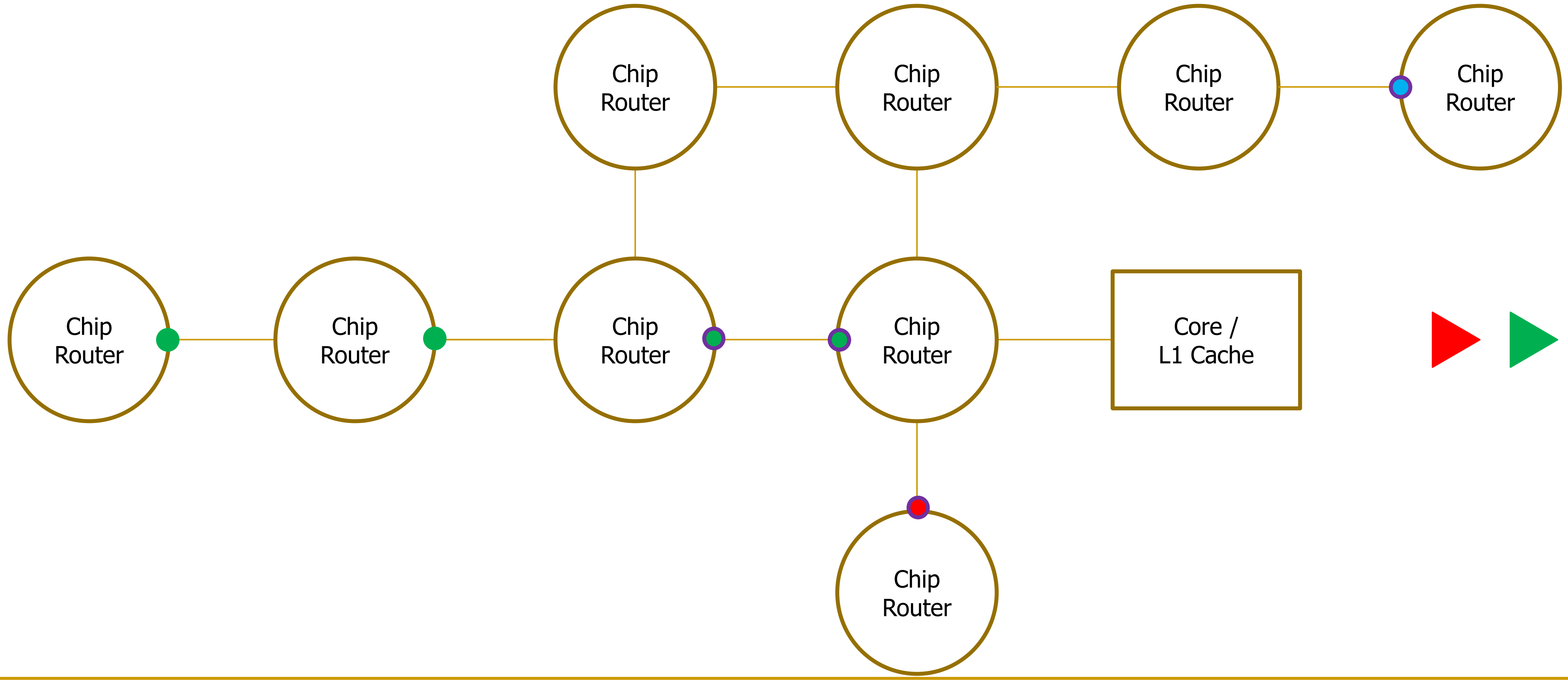
Worm Sluggishness

- What if a long young worm arrives at the same time as two shorter and older worms



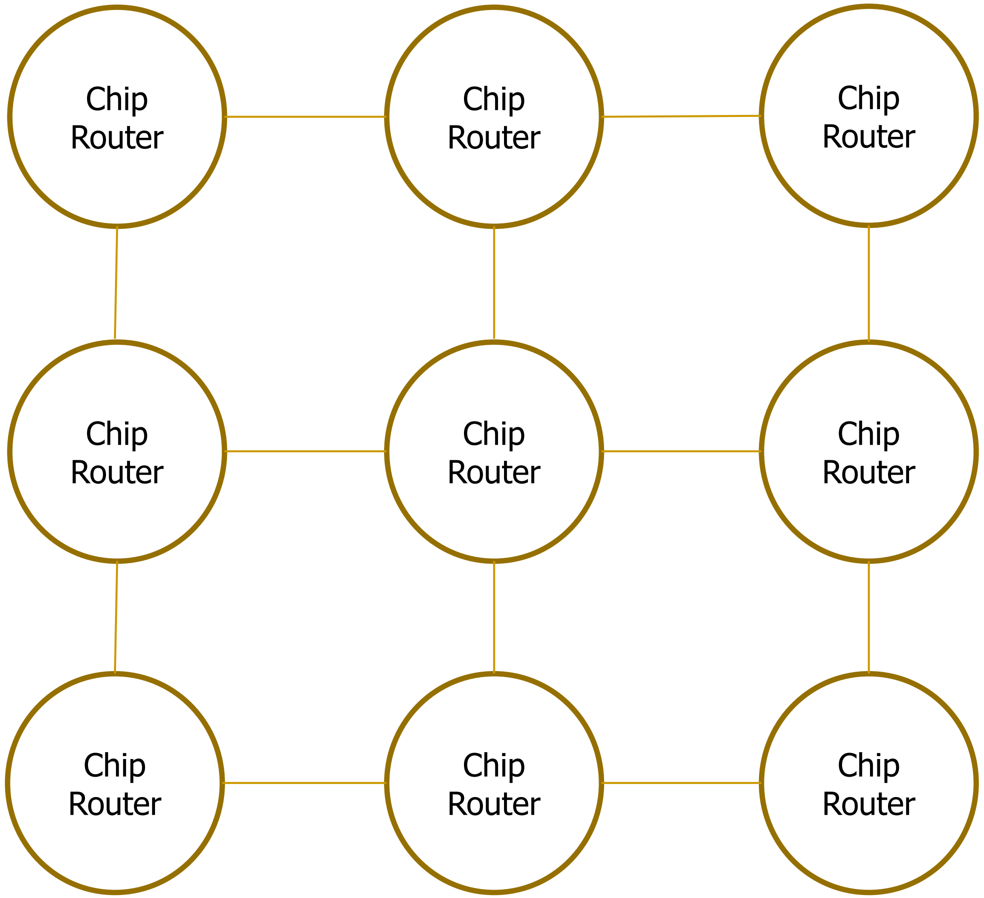
Worm Bless with more Truncation

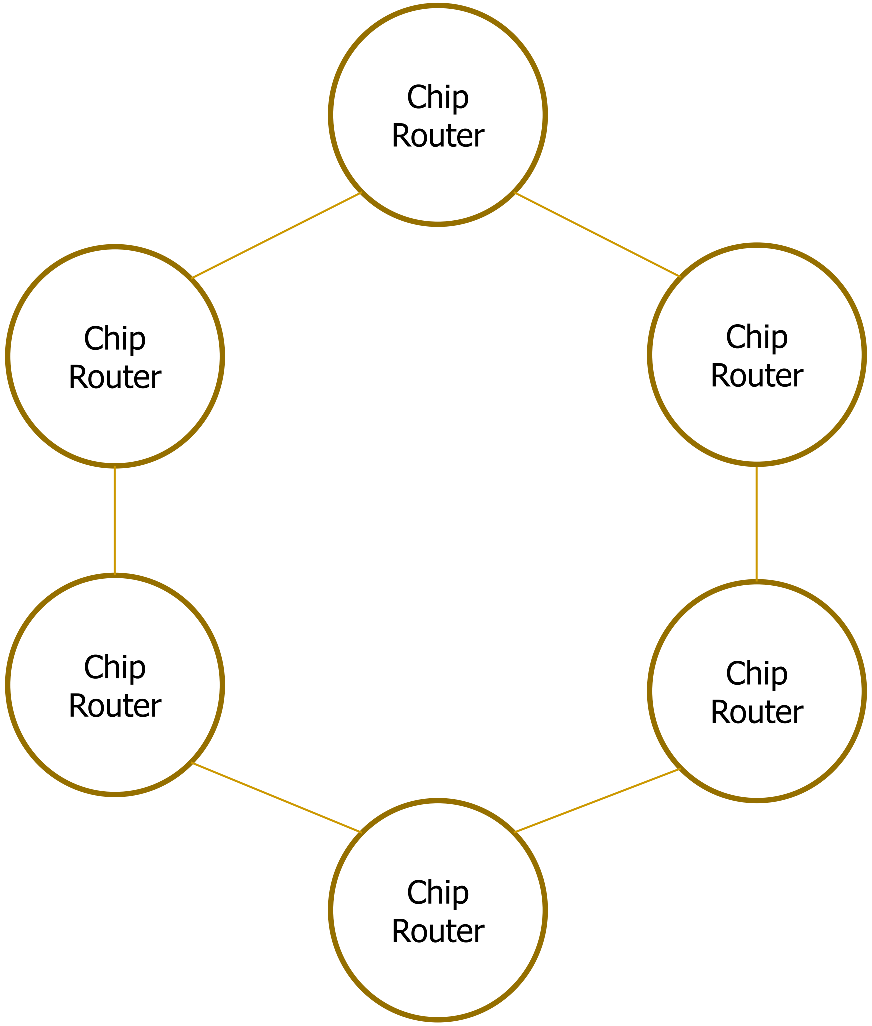
- What if a long young worm arrives at the same time as two shorter and older worms



Network Type

- Mesh
- Circle

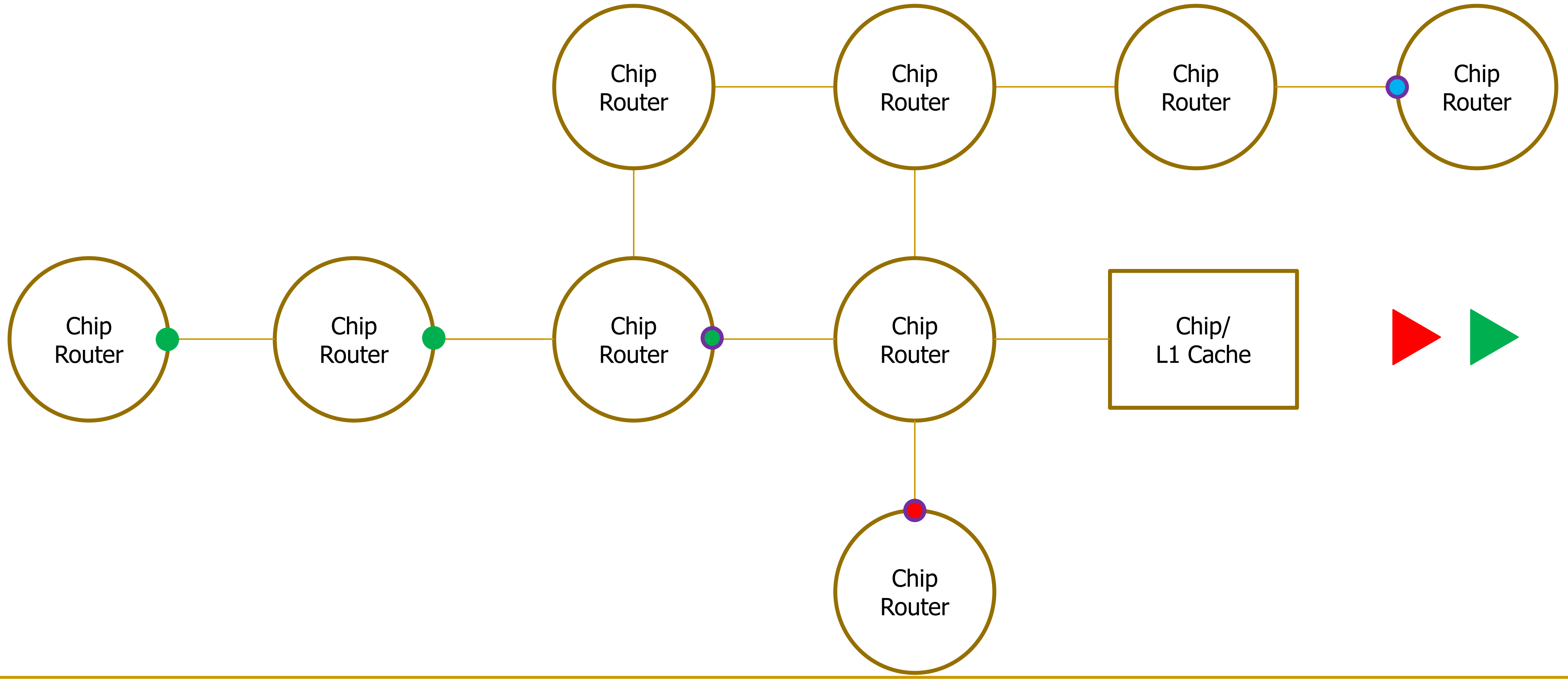




Questions & Discussion

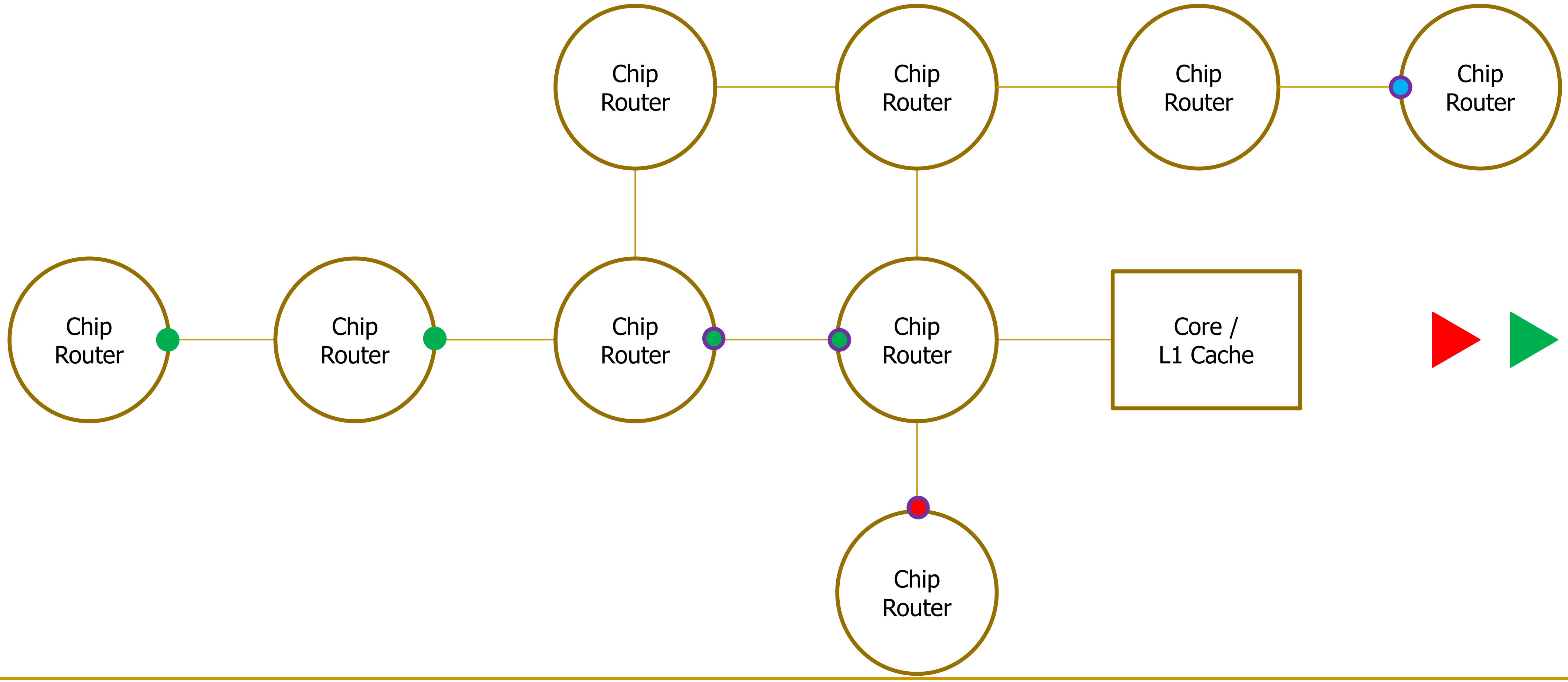
Discussion: Original Worm Algorithm

- What if a long young worm arrives at the same time as two shorter and older worms



Discussion: Worm Bless with more Truncation

- What if a long young worm arrives at the same time as two shorter and older worms

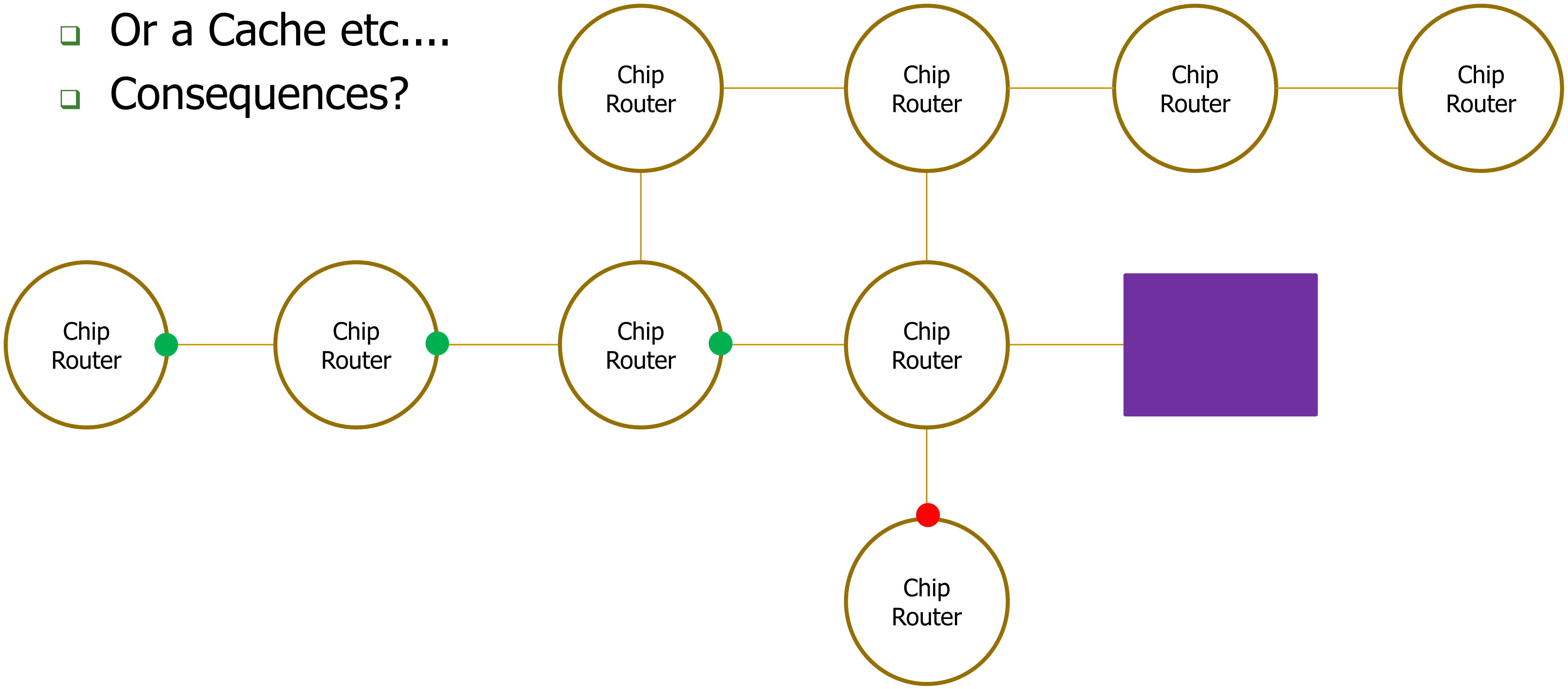


Discussion: Better Worm Algorithm

- Larger worm, less headflits, Sluggish
- Smaller worm, more headflits, less Sluggish
- Why not both?
- Split at the right moment
  - Possible, since Information in Head-flits is saved in the Node
  - Does not change anything if no split occurs

Discussion: To Split or not To Split

- What should this be?
  - Another Node
  - Or a Cache etc....
  - Consequences?

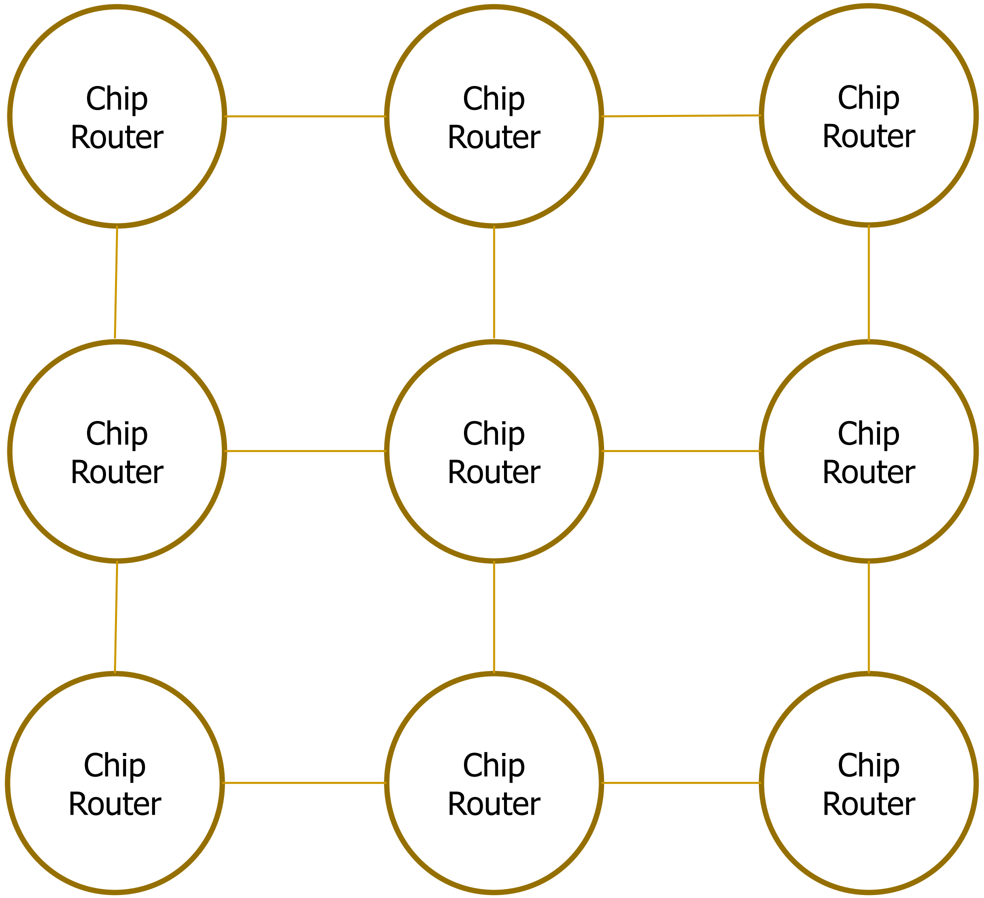


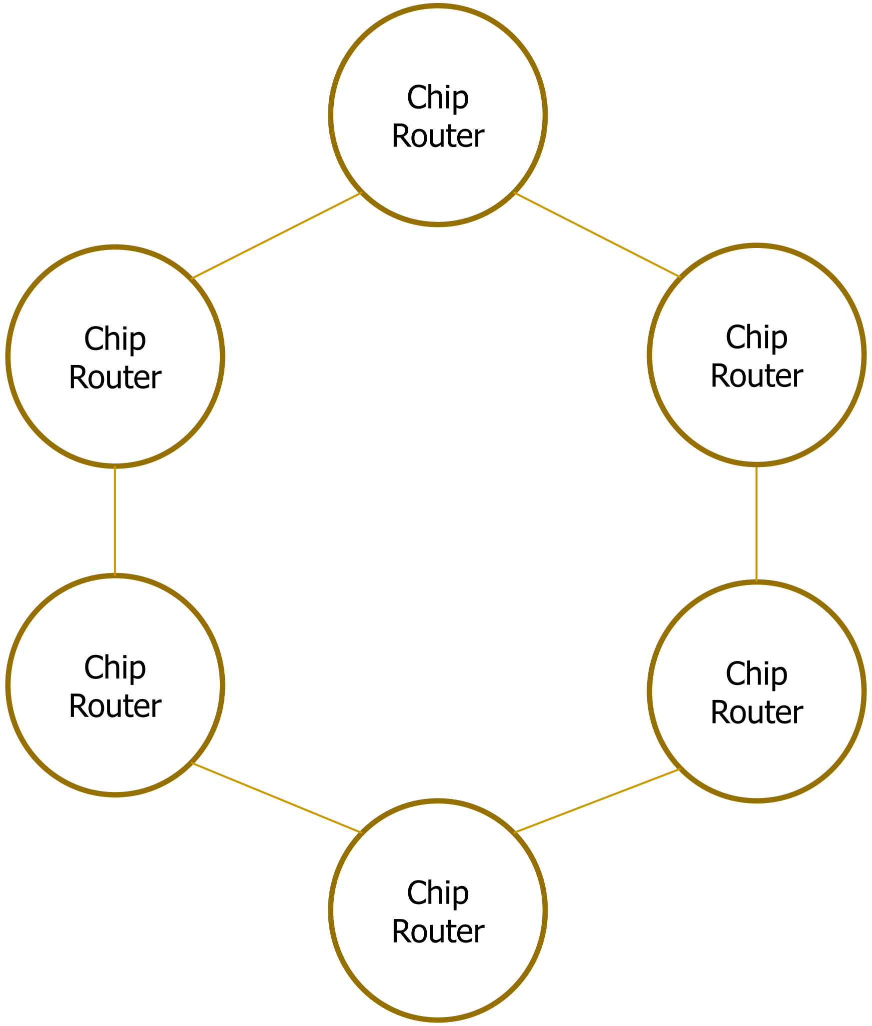
Discussion: Worm Length

- What are the consequences on latency, power, throughput etc...
- Long Worms in Network with
  - High Contention
  - Low Contention
- Short Worms in Network with
  - High Contention
  - Low Contention

Discussion: Network Type

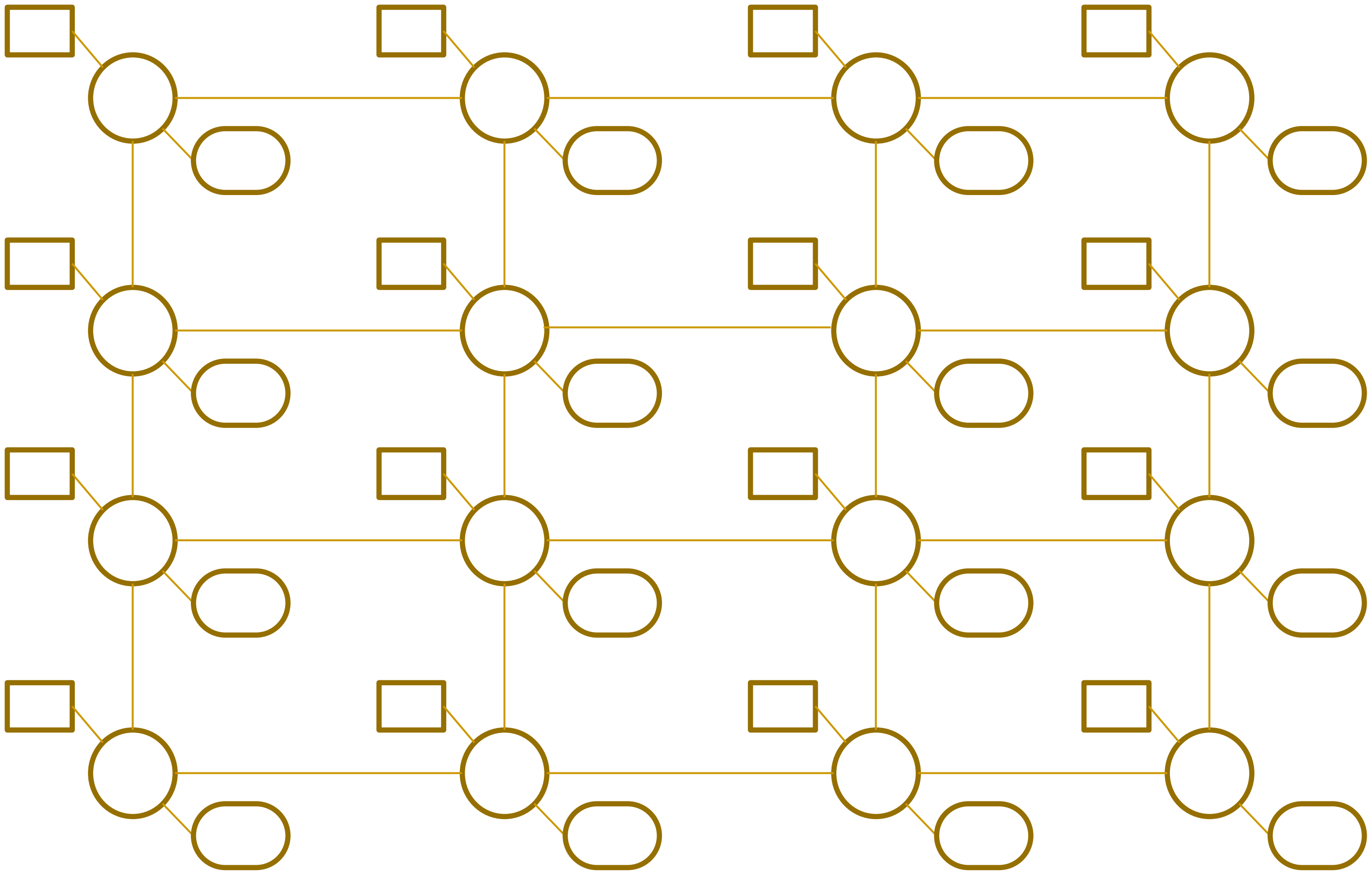
- Mesh
- Circle

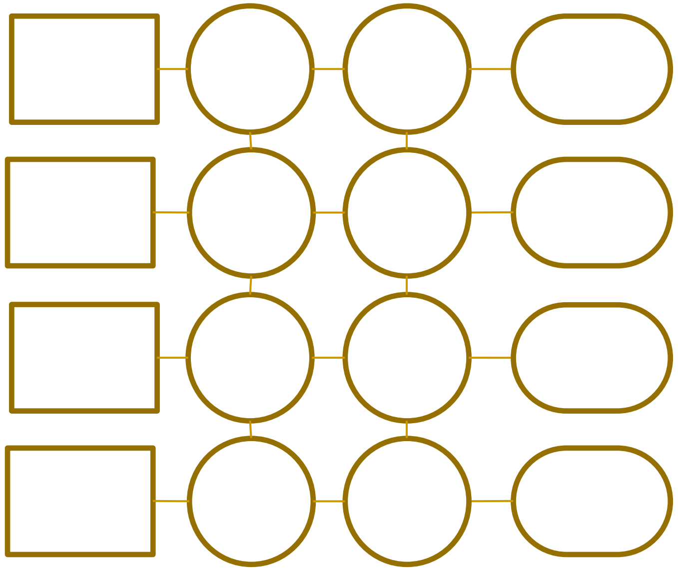




Discussion: Network Type

- Two Router per Element
- One Router per Element





A Case for Bufferless Routing in On-Chip Networks

Paper by

Thomas Moscibroda
Microsoft Research

Onur Mutlu
Carnegie Mellon University

ISCA 2009
Austin, Texas

Was presented to you by Bastian Amrhein