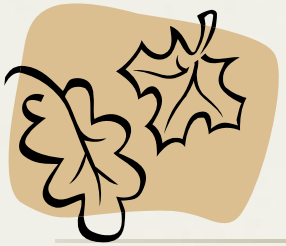
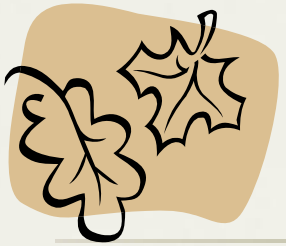
Power-efficient Tree-based Multicast Support for Networks-on-Chip

Wenmin Hu[1],
Zhonghai Lu[2], Axel Jantsch[2], Hengzhu Liu[1]
[1]National University of Defense Technology, P.R China
[2]Royal Institute of Technology, Sweden



Outline

- Motivation
- Multicast scheme
- Implementation
- Path search Algorithm
- Evaluation
- Conclusion



Need for multicast

- Parallel algorithm:
  - parallel search , parallel graph .etc
- Single program multiple-data programming model
- Data-parallel programming mode:
  - Replication , barrier synchronization
- Distributed shared-memory system:
  - Coherence Protocols

Insufficient of unicast router

- Implement multicast by multiple unicast
  - long startup latency for message
  - Redundantly in bandwidth occupying

Figure 1 is referred from [Jerger ISCA08], shows that even low amount of multicast traffic can leads Very poor performance

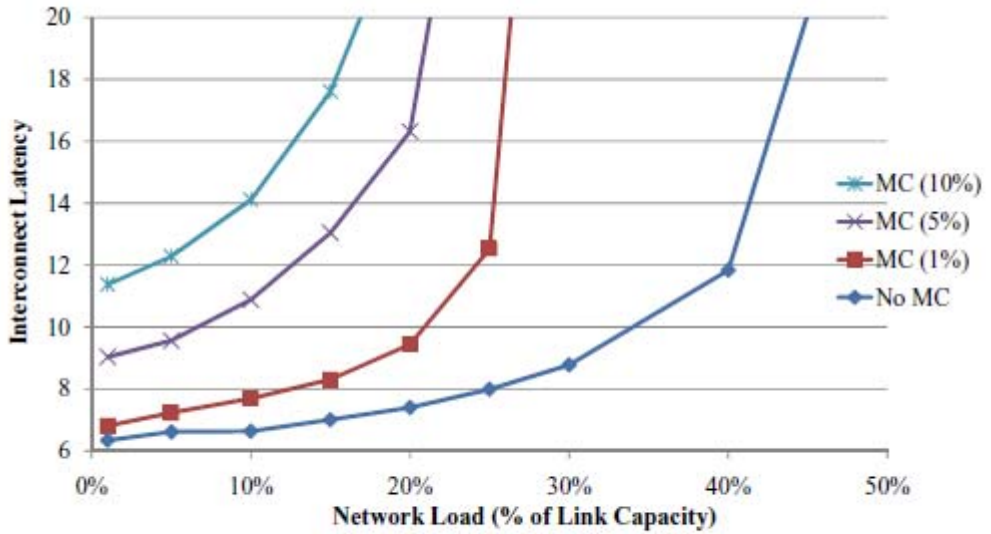
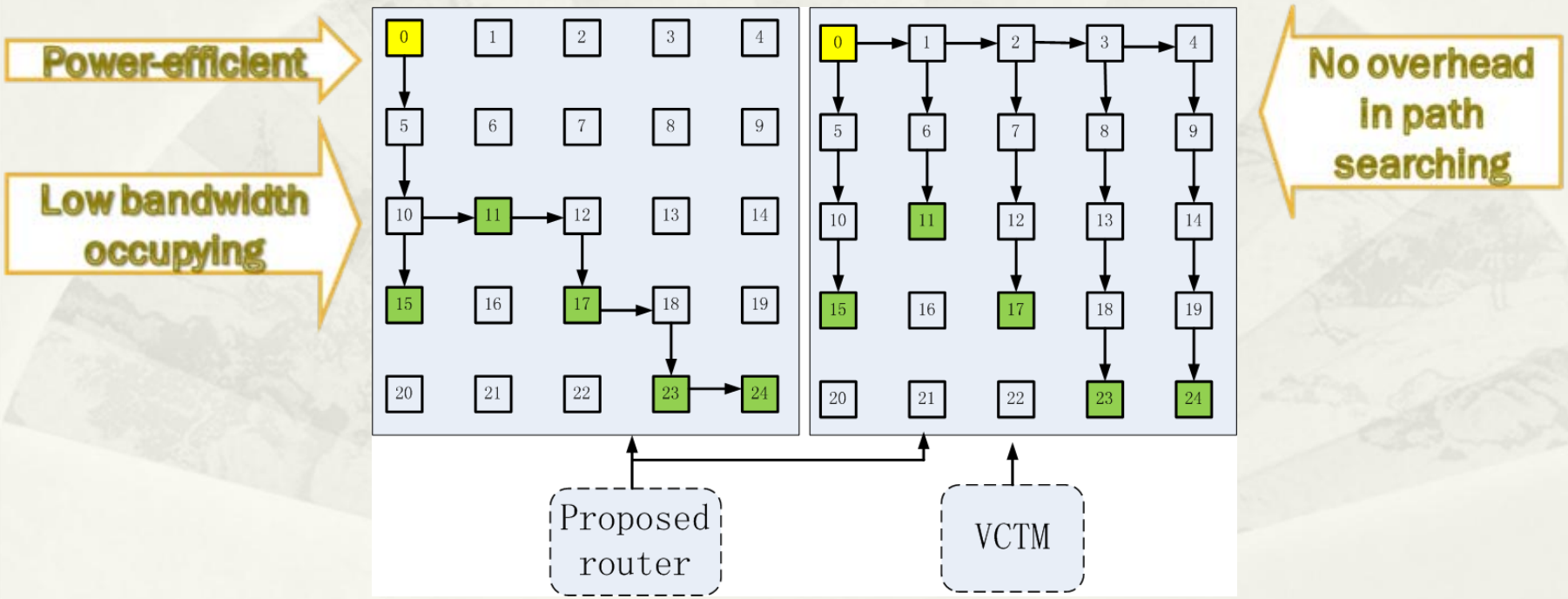
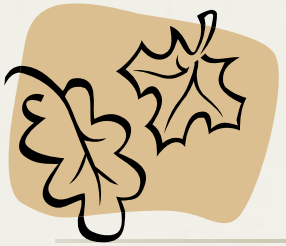


Figure 1. Performance of multicasts on packet-switched interconnect

VCTM VS proposed Router

- VCTM:
  - N. E. Jerger, L. S. Peh, M. Lipasti, Virtual Circuit Tree Multicasting: A case for On-chip hardware Multicast, Proceedings of the 35th annual international symposium on Computer architecture (ISCA), pp. 229–240, 2008.





Outline

- Motivation
- Multicast scheme
- Implementation
- Path search Algorithm
- Evaluation
- Conclusion

Packet format

* Packet for data transfer:

Head/ Body /Tail/HBT	Paket Type	Set/clear	SRC	MCT#	DST	VCID	Payload
2 bits	3 bits	1 bit	6 bits	4 bits	6 bits	2 bits	Payload
00: Head	000 UC	x		xxx			
00: Head	011 MC NORMAL	x			xxx		

* UC: unicast data packet
* MC NORMAL: multicast data packet

Packet format

* Packet for path setup:

Head/ Body /Tail/ HBT	Paket Type	Set/ clear	SRC	MCT#	DST	VCID	Payload
2 bits	3 bits	1 bit	6 bits	4 bits	6 bits	2 bits	Payload
11 : HBT	001 MC_SET_1	0 : set 1 : clear					6 bits ' 2 nd DST
11 : HBT	010 MC_SET_2	0 : set 1 : clear					xxx
11 : HBT	101 MC_ SETUP RPLY	0 : set 1 : clear					

* MC_SET_1: first period, routes to the intermediate node, no updating multicast table in router traversed
* MC_SET_2: second period, routes to the destination node, updating multicast table in router traversed
* MC_SETUP_RPLY: to inform source node of success in sub-path setup

Packet format

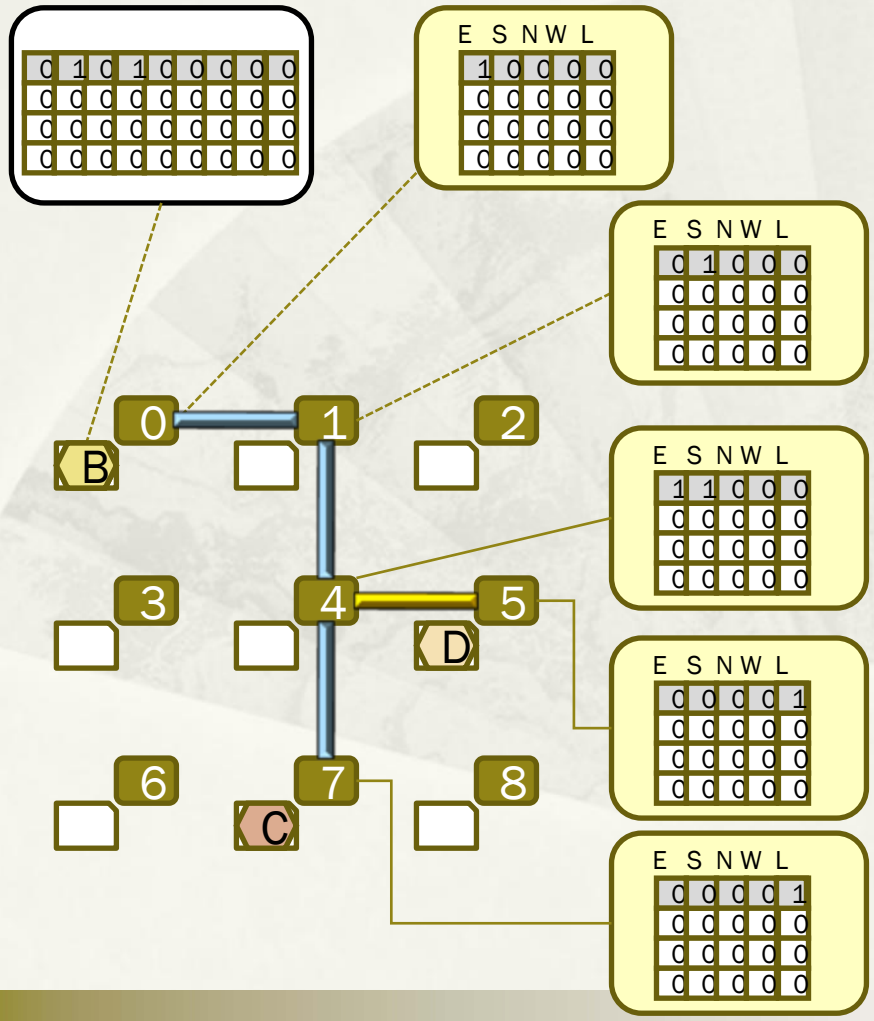
- Packet for path setup:

Head/ Body /Tail/ HBT	Paket Type	Set/clear	SRC	MCT#	DST	VCID	Payload
2 bits	3 bits	1 bit	6 bits	4 bits	6 bits	2 bits	Payload
11: HBT	100 MC_CLEAR	x			xxx		
11: HBT	110 MC_CLEAR_RPLY						

- MC_CLEAR: routes like multicast data packet, but would clear corresponding multicast table entry in the router traversed
- MC_CLEAR_RPLY: to inform source node of success in evicting multicast tree

Example of path setup

Destination Set Content Addressable Memory



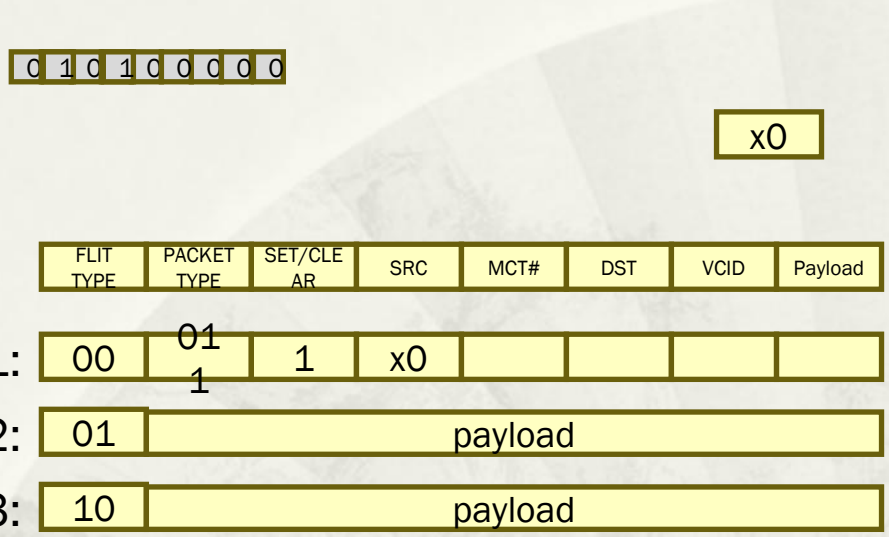
	FLIT TYPE	PACKET TYPE	SET/CLEAR	SRC	MCT#	DST	VCID	Payload
A:	11	010	1	x0	x0	x7		
B:	11	001	1	x0	x0	x4		X5
C:	11	101	1	x7	x0	x0		
D:	11	101	1	x5	x0	x0		

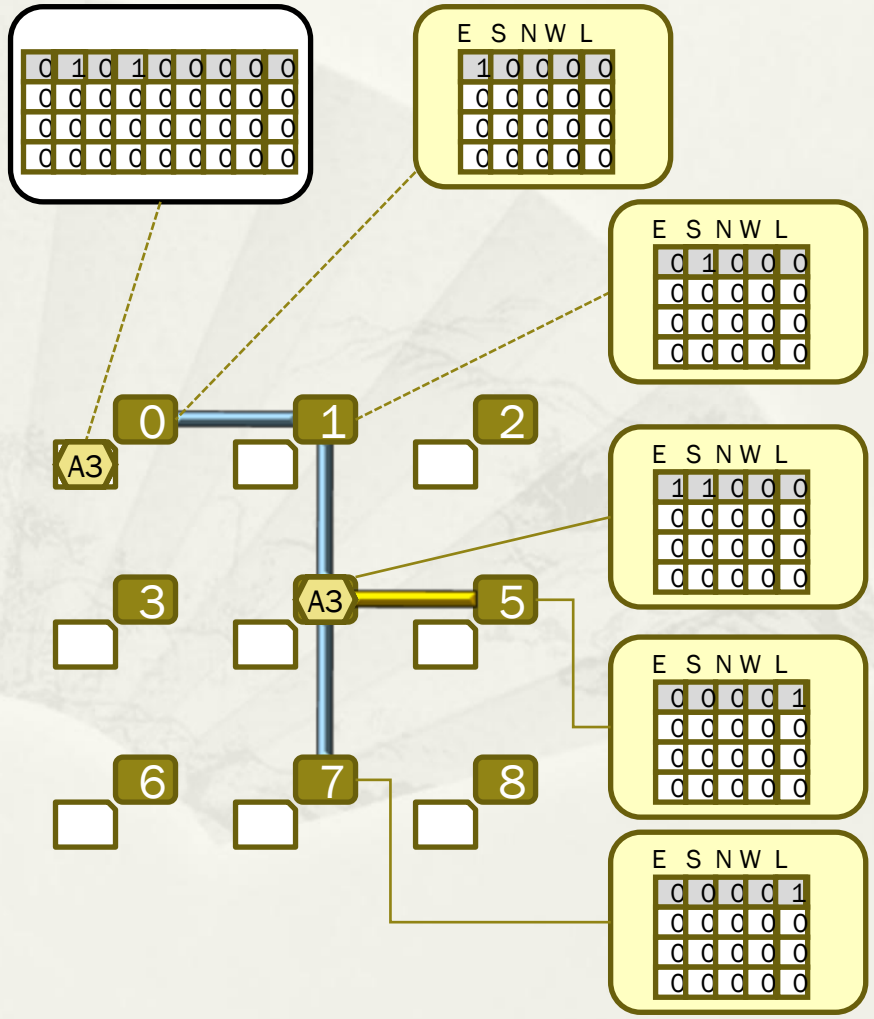
Source 0 to destination set {5,7}

Setup packet: A and B

Reply packet: C and D

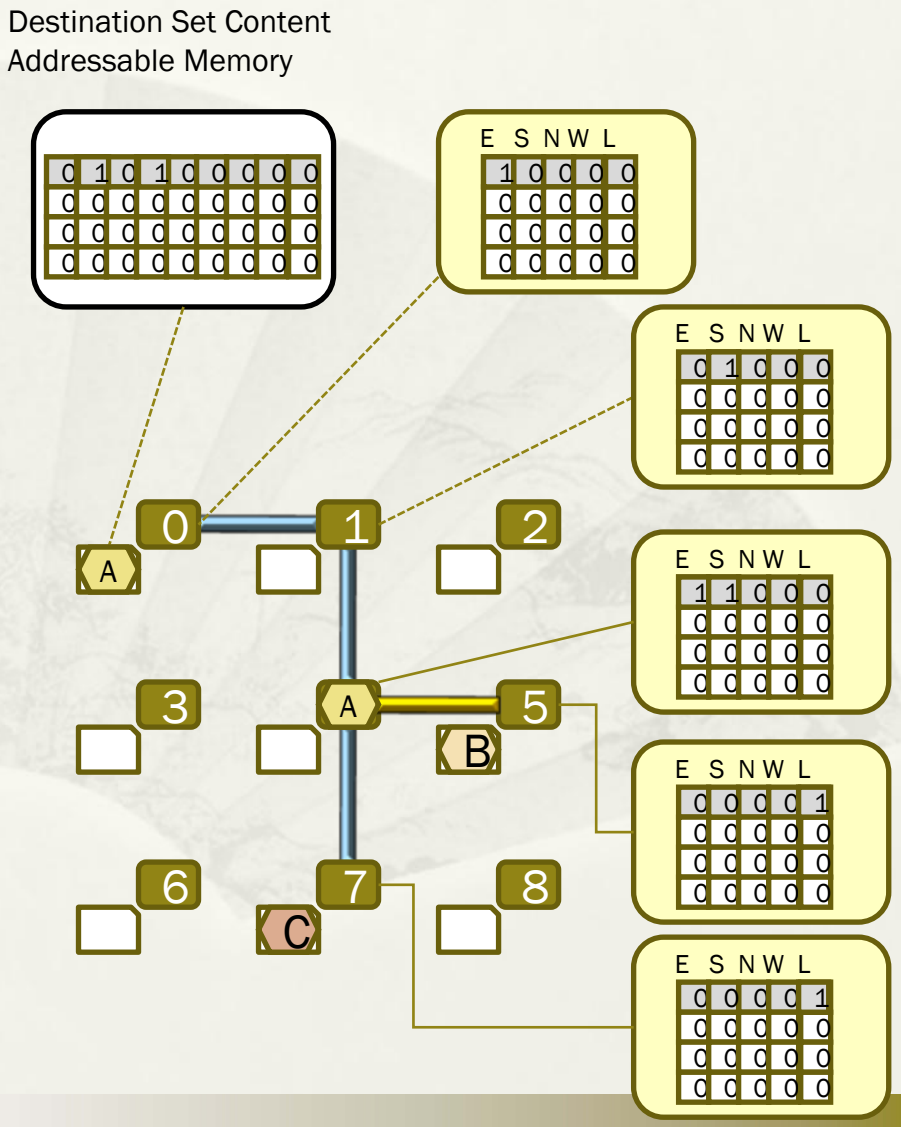
Example of data transfer

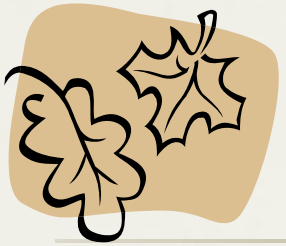




multicast tree evicting

	FLIT TYPE	PACKET TYPE	SET/CLEAR	SRC	MCT#	DST	VCID	Payload
A:	11	100		x0	x0			
B:	11	110		x5	x0	x0		
c:	11	110		x7	x0	x0		

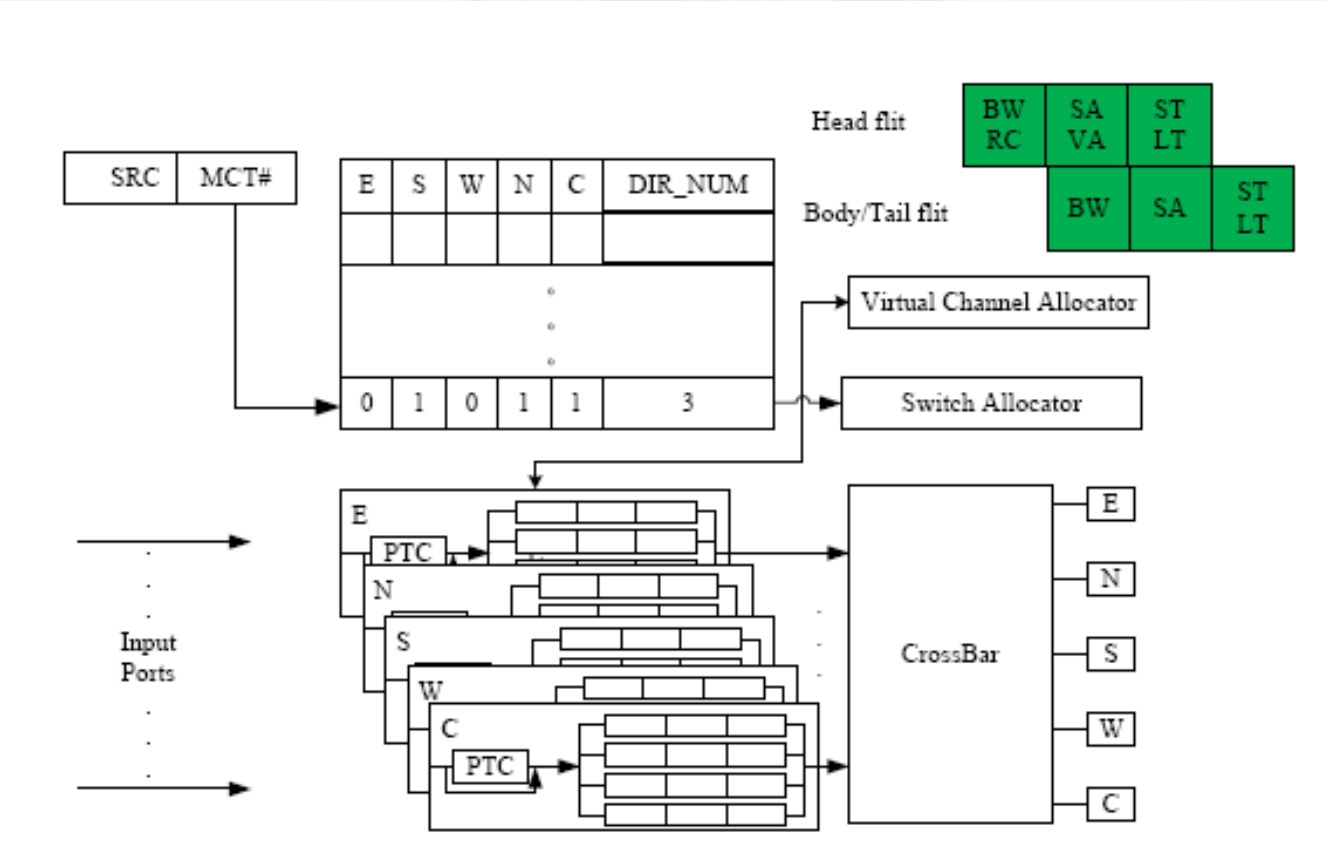




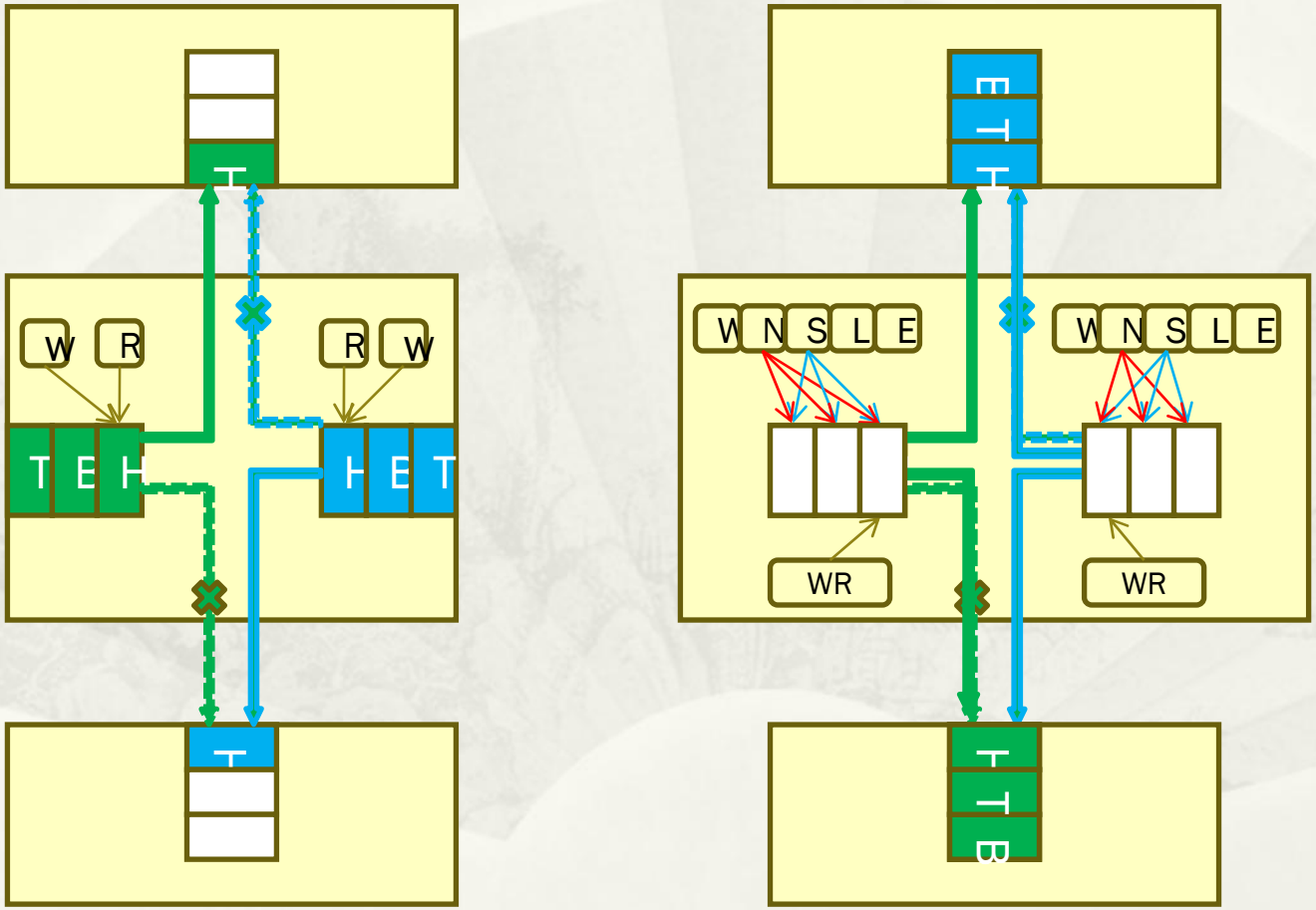
Outline

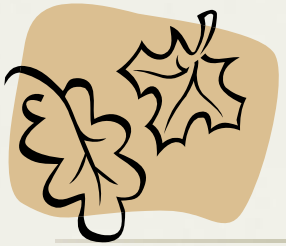
- Motivation
- Multicast scheme
- Implementation
- Path search Algorithm
- Evaluation
- Conclusion

Router architecture



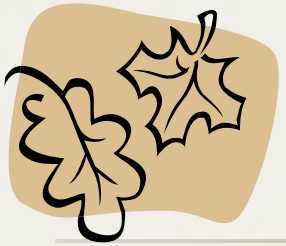
Deadlock free





Outline

- Motivation
- Multicast scheme
- Implementation
- Path search Algorithm
- Evaluation
- Conclusion



Path search algorithm

- To find (branch node, leaf node) pairs that can form the whole tree
- Tree shape optimization
- Deadlock free
- flexibility
  - Optimized algorithm: pre-configured before application running
  - Simple algorithm: dynamically configuration

Optimized tree(OPT)

- West first turn model!!
- Similar to minimal spanning tree algorithm

Algorithm: Generate the optimized multicast tree based on west-first turn model
Input: Destination set D, Source node $s : (x_0, y_0)$;
Output: Pair setD_{pair};
Define: $k(a, b) = |a.y - b.y| + |a.x - b.x|$;
Initial: $D_{node} \leftarrow s, D_{pair} \leftarrow \emptyset$;

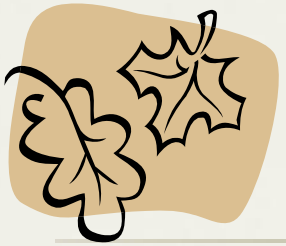
1: Find the node $v \in D, \forall a \in D, v.y \leq a.y$. Add (s, v) into D_{pair}, removed v from D, add the nodes on the path from s to v into D_{node}
2: **while** D is not empty **do**
3: $D_{pair\_tmp} \leftarrow \{(u, v) | u \in D_{node}, v \in D, that \forall a \in D_{node}, \forall b \in D, k(u, v) \leq k(a, b)\}$
4: Select $(u, v) \in D_{pair\_tmp}$, that $\forall (a, b) \in D_{pair\_tmp}$, $v.y \leq b.y$
5: Add (u, v) into D_{pair}, remove v from D, add the nodes on the path from u to v into D_{node}
6: **end while**

Left-XY-Right-Optimized Tree (LXYROPT)

- West first turn model!!
- hops from each destination node to source node is minimum

Algorithm: Generate the LXYROPT multicast tree based on west-first turn model
Input: Destination set D, Source node $s : (x_0, y_0)$;
Output: Pair set D_{lpair}, D_{mrpair};
Define: $k(a, b) = |a.y - b.y| + |a.x - b.x|$;
Initial: $D_{node} \leftarrow s, D_{lpair} \leftarrow \emptyset, D_{mrpair} \leftarrow \emptyset$;

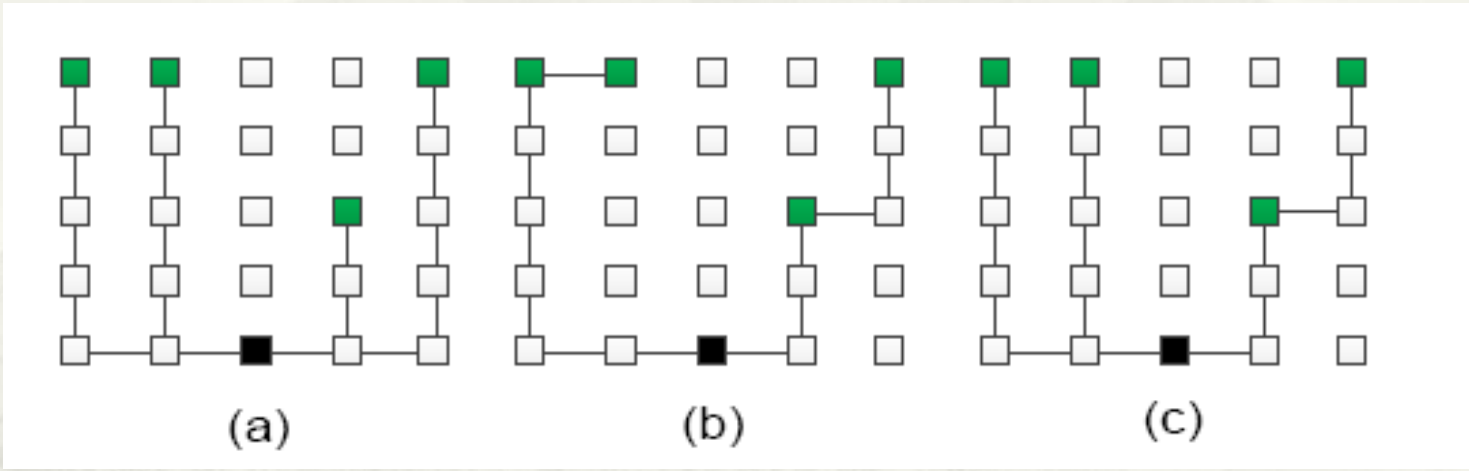
1: $D_{left} \leftarrow \{(x, y)|(x, y) \in D, y < y_0\}$
2: $D_{mid-right} \leftarrow \{(x, y)|(x, y) \in D, y \geq y_0\}$
3: **while** D_{left} is not empty **do**
4: Find a node $v \in D_{left}$, Add (s, v) into D_{lpair}, remove v from D_{left}
5: **end while**
6: **while** $D_{mid-right}$ is not empty **do**
7: $D_{pair\_tmp} \leftarrow \{(u, v)|u \in D_{node}, v \in D_{mid-right},$ that $k(s, v) = k(s, u) + k(u, v)$
8: Select $(u, v) \in D_{pair\_tmp}$, that $\forall(a, b) \in D_{pair\_tmp}$, $k(u, v) \leq k(a, b)$
9: Add (u, v) into D_{mrpair}, remove v from $D_{mid-right}$, add the nodes on the path from u to v into D_{node}
10: **end while**



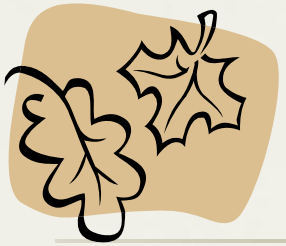
XYT(VCTM)

- Source node is defined as the intermediate node
- No overhead in path search
- Fit for dynamically multicast path setup

Example of three algorithms



XYT(VCTM) OPT LXYROPT



Outline

- Motivation
- Multicast scheme
- Implementation
- Path search Algorithm
- Evaluation
- Conclusion

Network configuration

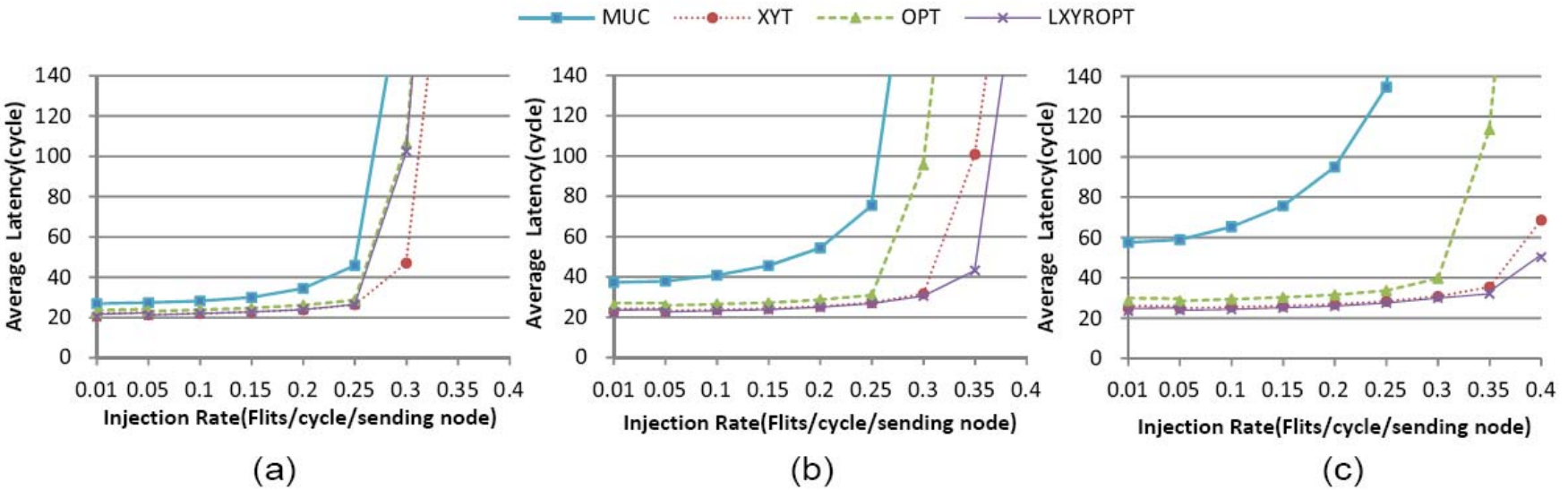
* A cycle-accurate router model based on SystemC

TABLE I
NETWORK PARAMETERS

Topology	8×8 mesh
Routing	Multicast: Tag ID based Unicast: XY
Packet Size	1 flit: Multicast setup packet 3 flits: others
Virtual Channels	4
Buffers per Channel	3
Router ports	3 For corner 4 For border 5 For others
Number of MCT entries	16

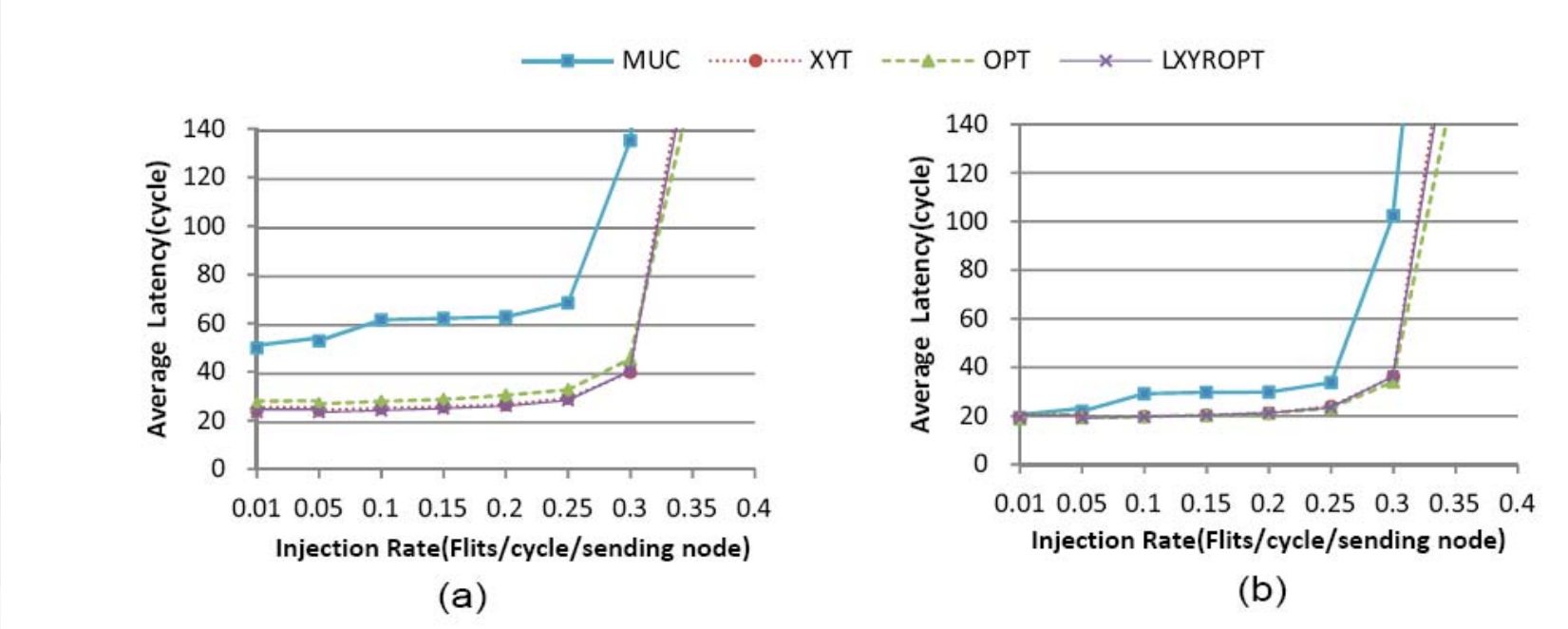
Multicast profile

- Source node is uniform distribution
- Destination node is uniform distribution
- Scenarios
  - 16 source node , each to 5 nodes
  - 8 source node, each to 10 nodes
  - 4 source node, each to 20 nodes



Unicast and multicast mixed traffic profile

* Multicast is accounted for 20%

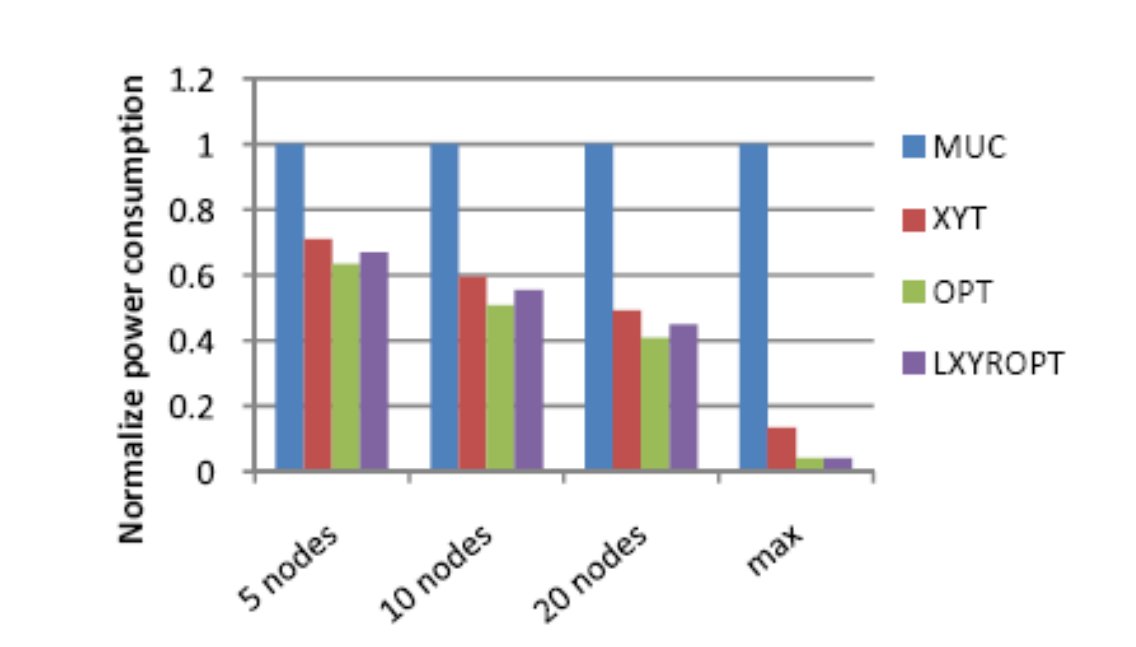


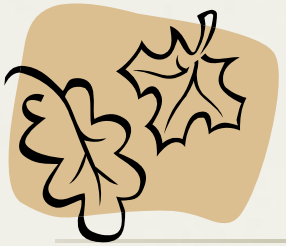
Multicast traffic

Unicast traffic

Power consumption

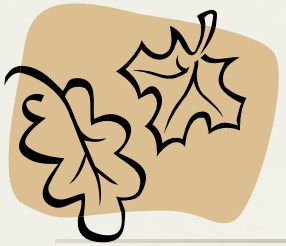
- Library of noxim
- Count the number of routing ,selection, forward, incoming and standby during the simulation





Outline

- Motivation
- Multicast scheme
- Implementation
- Path search Algorithm
- Evaluation
- Conclusion



Conclusion

- A hardware supports any-shaped multicast path
- Two power-efficient path search algorithms
- Flexibility: support XYT(VCTM)
- Simple deadlock free scheme
- Open door for path search algorithm optimization

Tank you!!! Questions?