

JUNCTION BASED ROUTING: A SCALABLE TECHNIQUE TO SUPPORT SOURCE ROUTING IN LARGE NOC PLATFORMS

**Shabnam Badri, Rickard Holsmark and Shashi Kumar
JÖNKÖPING UNIVERSITY, SWEDEN**

NoCArc Workshop, Vancouver, Canada, 2012-12-01



Outline

- ❖ Goals and Motivations
- ❖ NoC Background
- ❖ Introduction to JBR
- ❖ Performance Evaluation of JBR
- ❖ Conclusions



Goals and Motivations

Goal:

Improvement of the communication between components of a system which is integrated on a single chip.

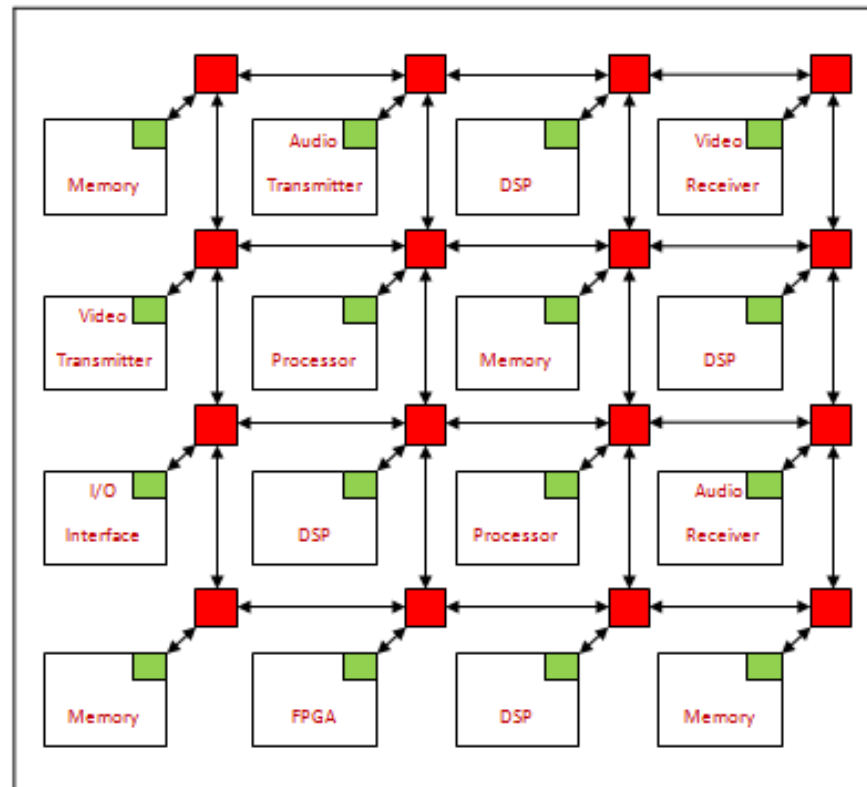
Motivations:

- In the embedded systems using such a chip, the communication patterns can be profiled off-line and routing can be well planned.
- Source routing is very suitable in such contexts.
- Source routing has one serious drawback of overhead for storing the path information in header of every packet.
- This disadvantage becomes worse as the size of the network grows.
- A technique, called Junction Based Routing (JBR), can handle this problem.



Network on Chip (NoC)

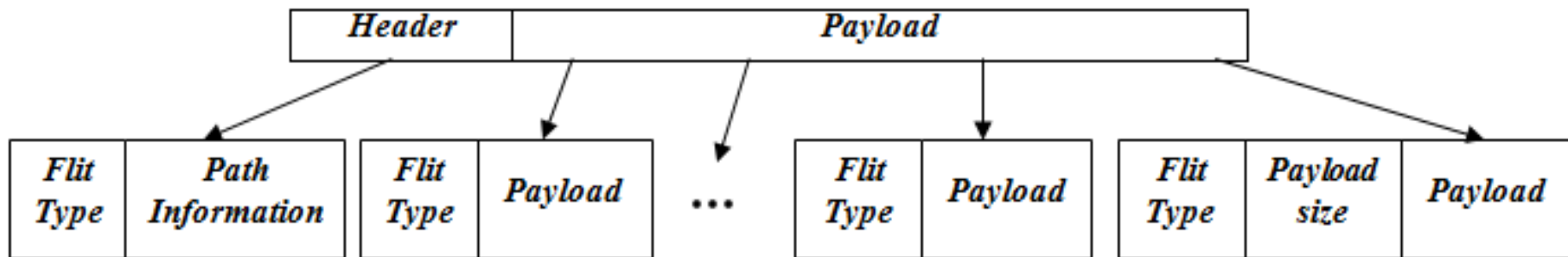
- The numbers of cores that can be integrated are growing linearly with the increase in chip capacity.
- A dominant paradigm for synthesis of multi-core SoCs.
- A packet switched network.
- A NoC-based system is usually considered scalable.





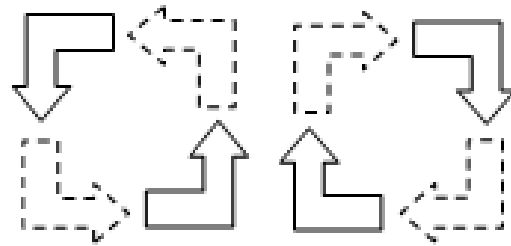
Routing Techniques

- **Switching**
 - Latency in the network strongly depends on the chosen switching technique.
 - Packet switching and circuit switching.
 - Store and forward, wormhole and cut-through switching.
- **Routing**
 - *Source vs. Distributed* routing, *Deterministic vs. Adaptive*, *Static vs. Dynamic* routing, *Minimal vs. Non-minimal* routing.
 - Application Specific Routing.

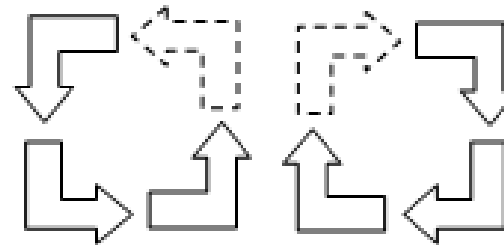




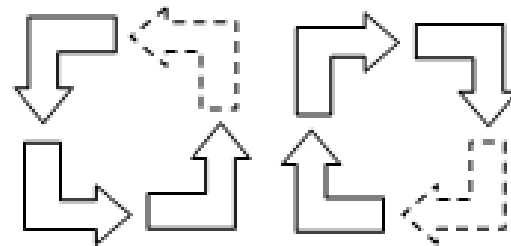
Turn-Model Routing Algorithms



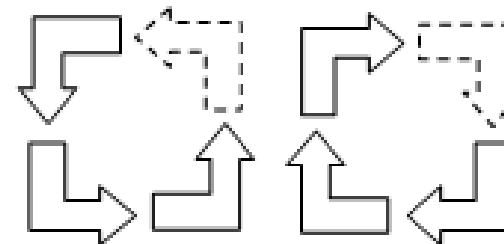
xy



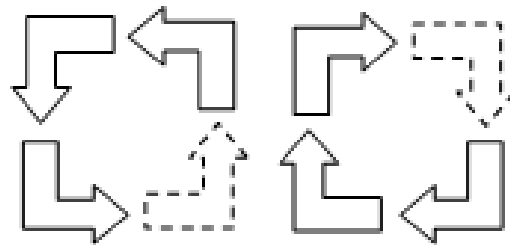
North-Last



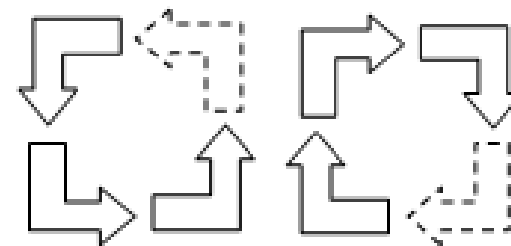
West-First



Negative-First



Odd-Even (Even Column)



Odd-Even (Odd Column)



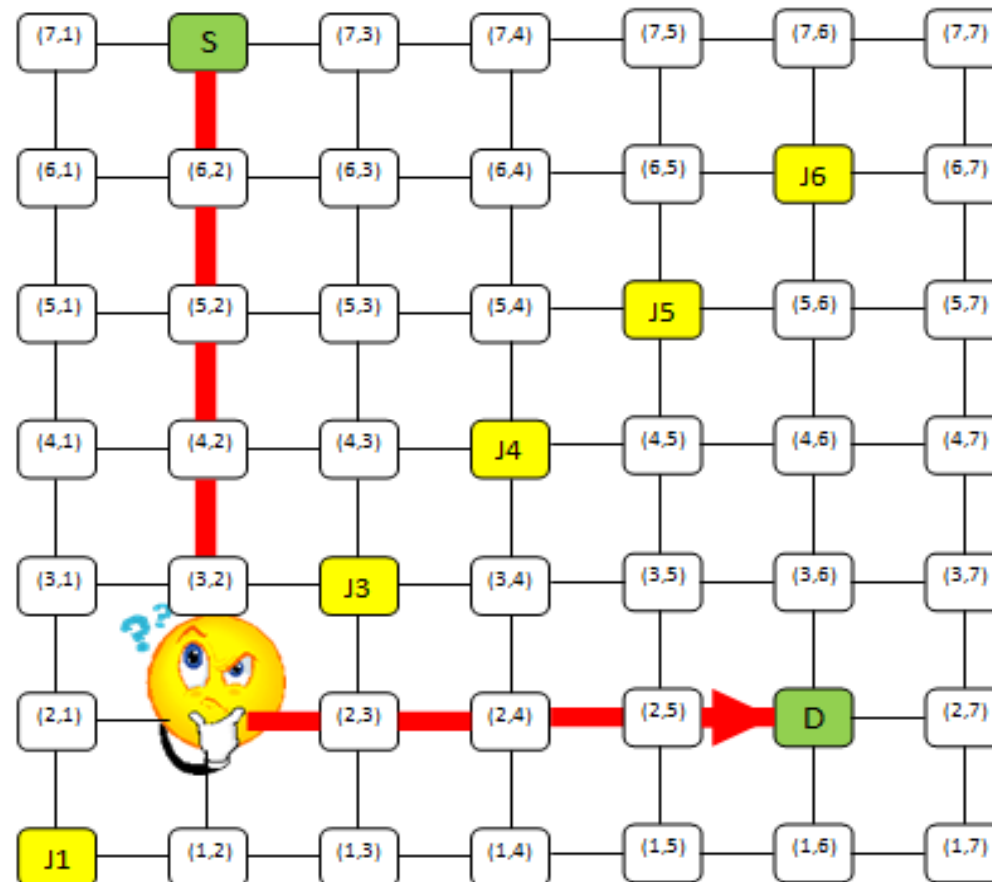
Related Work

- A large number of deadlock-free distributed routing algorithms for NoCs have been proposed.
- Source routing has been used in interconnection networks for IBM SP1 multiprocessor.
- Hierarchical organization of networks and hierarchical routing for large on-chip networks.
- **JBR: an alternative.**



Junction-Based Routing

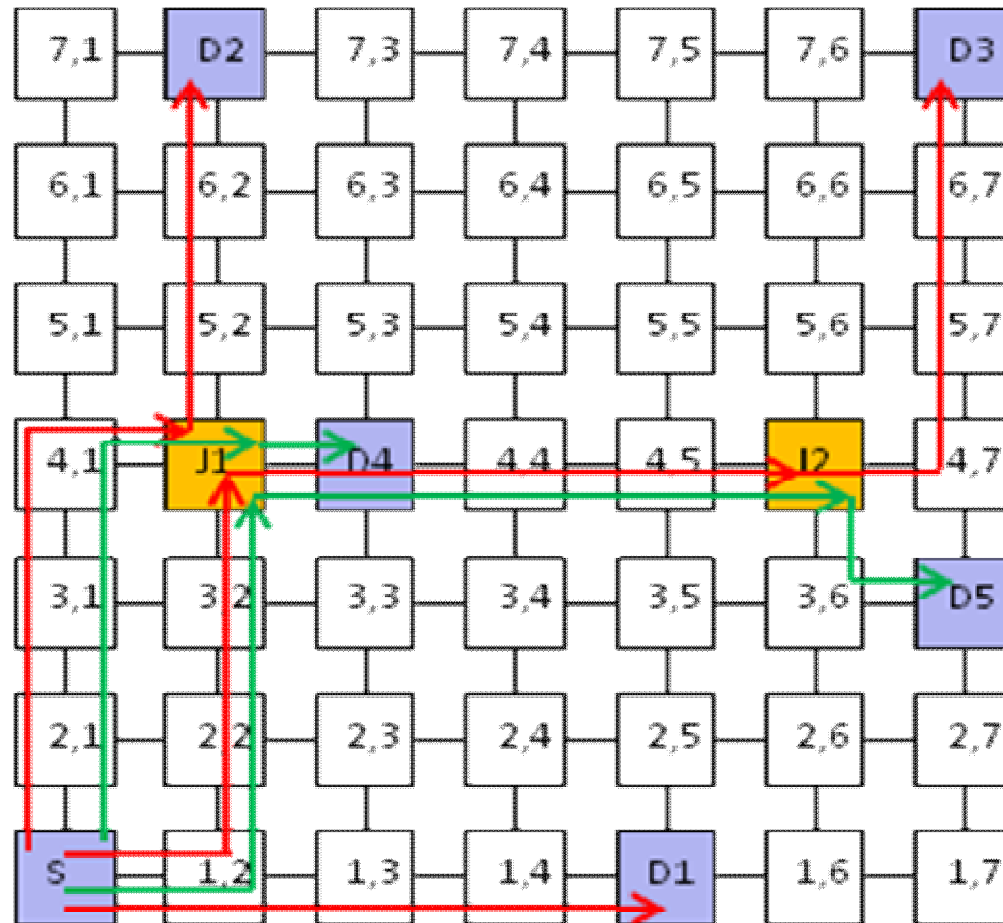
- The idea is derived from the railway networks.
- A large distance can be covered by going through intermediate temporary destinations (called Junctions) such that each sub-path is bounded by a hop limit.





An Illustration of Using Junctions

- This work considers 2-D mesh topology NoC for application of JBR.
- The idea of JBR is general and will be applicable to all topologies- regular or irregular.





Path Information

Wormhole routing is used.

(7,7)	Payload
-------	---------

a) Distributed Routing

(1,1) (1,2) (1,3) (1,4) (2,4) (3,4) (4,4) (5,4) (6,4) (7,4) (7,5) (7,6) (7,7)	Payload
---	---------

b) Source Routing

When the packet leaves the resource node:

(1,1)(2,1)(2,2)(3,2)(4,2)	Payload
---------------------------	---------

When the packet leaves the first junction node:

(4,2)(4,3)(4,4)(4,5)(4,6)	Payload
---------------------------	---------

When the packet leaves the second junction node:

(4,6)(5,6)(6,6)(7,6)(7,7)	Payload
---------------------------	---------

c) Junction-Based Routing



Header Overhead of JBR

- Header, body and end flits.
- Number of bits of data that can be carried by the header flit.
- H : limit on the maximum number of hops in JBR.

FT	DA	TD	Path Information	Payload
-----------	-----------	-----------	-------------------------	----------------

FT	Payload
-----------	----------------

FT	Payload Size	Payload
-----------	---------------------	----------------

H	7	6	5	4	3
Number of Bits of Data	11	13	15	17	19



Comparison of Header Overhead

- In a 7x7 mesh network with a hop count limit of 7 hops, the size of the memory required for path storage in every resource is almost half of the size of the memory that is needed in pure source routing.
- 2-bit clockwise router port address encoding scheme for encoding routing information.

Mesh Size	Distributed Routing	Source Routing	JBR (H=4)
5x5	6 bits	18 bits	8+6+1 = 15bits
6x6	6 bits	22 bits	15 bits
7x7	6 bits	26 bits	15 bits
8x8	6 bits	30 bits	15 bits
10x10	8 bits	38 bits	17 bits
16x16	8 bits	62bits	17 bits



Path Length Overhead

- For a 7X7 NoC and a hop count limit of 7 hops and a flit size of 34 bits:
 - 6 bits for DA field, 2 bits for FT and 14 bits for path information field.
 - A possibility of accommodating 11 bits of payload.
 - 32 bits of payload can be transported in the body flit.
 - It is possible to accommodate up to 24 bits in the end flit.

- **The overhead in JBR grows very slowly and therefore is more scalable.**

- The path overhead in terms of bits to specify path for an $N \times N$ mesh network:

- Distributed Routing: $2\lceil \log_2 N \rceil$

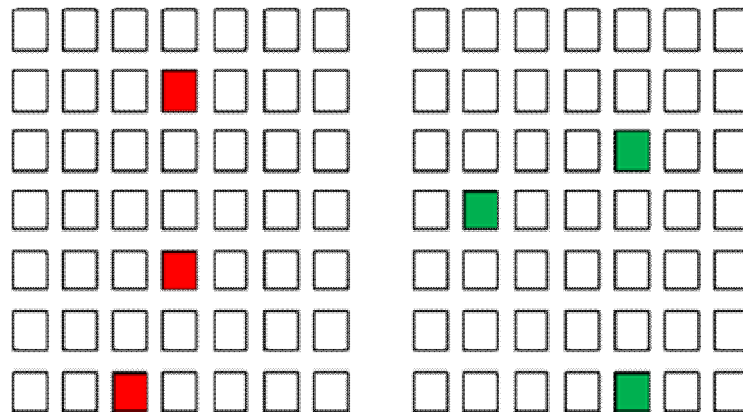
- Source Routing: $2(2N - 1)$

- JBR: $2H + 2\lceil \log_2 N \rceil + 1$



Number and Position of Junctions

- A minimum number of junctions are required to be placed in the network to achieve full reachability. We also need to position the junctions in the network such that:
 - There is a path from every node to at least one junction with path length less than the hop count limit.
 - There is a path from one junction to at least one more junction with path length less than the length limit (except for a trivial case when the network has only a single junction).
 - If we draw a graph in which every junction is a node and a pair of junctions have an edge between them if and only if the path length between them is less than the path length limit. This graph must be connected. This condition is necessary for ensuring reachability of any node from every other node in the network.

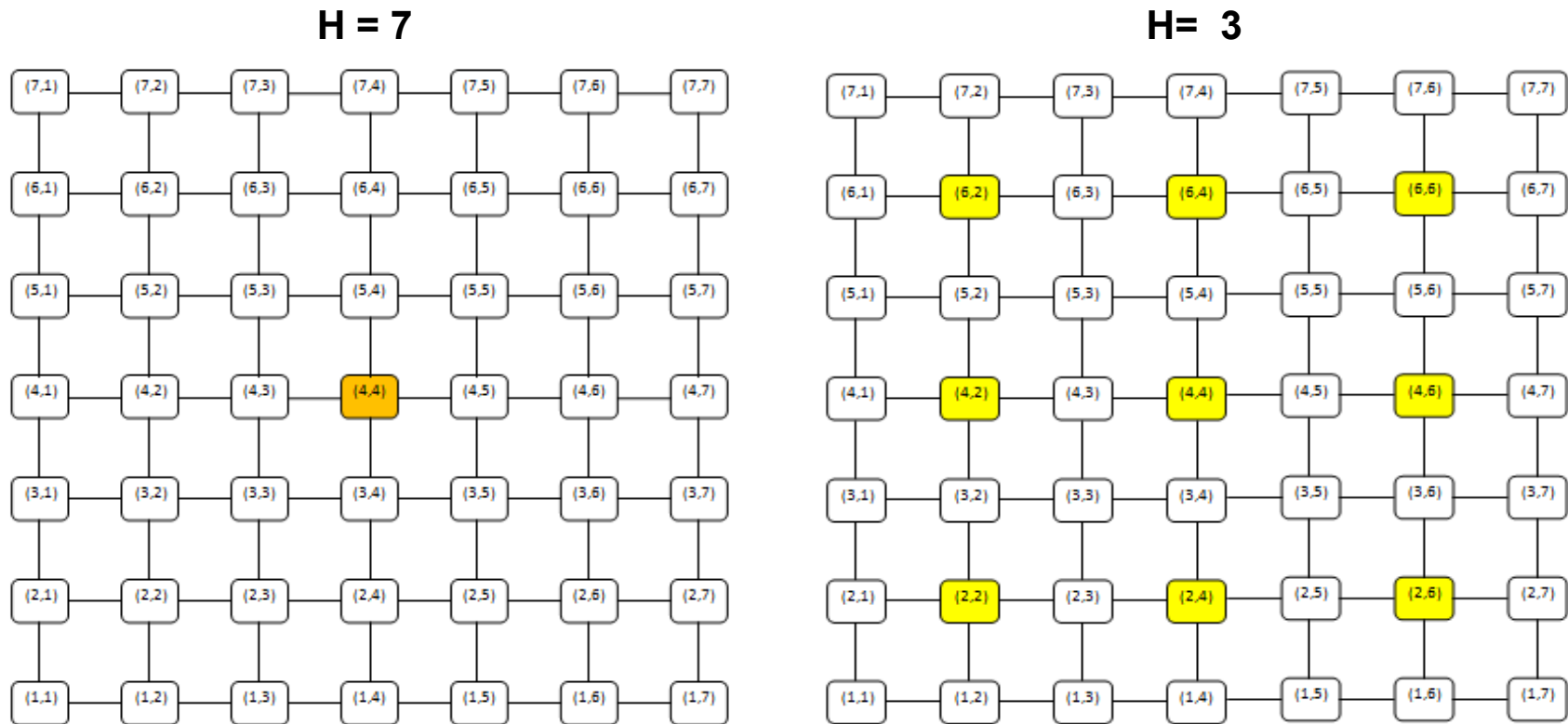


Two configurations of three junctions for a 7x7 NoC and an H of 5.



Junctions vs. Hop Count Limit

- An algorithm has been developed to find number and position of junctions for a given hop count limit for mesh of any size.



- The number of junctions is not comparable with the total number nodes in the network and the number of junctions grows slowly with decreasing the hop count limit or increasing the network size.



Number of Junctions vs. Hop Count Limit

Hop Count Limit (H)	Number of Junctions (NJ)	Number of Configurations	NJ/NN	Number of Bits for Path Header
13	0	1	0	33
12	1	45	0.02	31
11	1	37	0.02	29
10	1	25	0.02	27
9	1	13	0.02	25
8	1	1	0.02	23
7	1	1	$1/49=0.02$	$7*2+6+1=21$
6	2	40	$2/49=0.04$	$6*2+6+1=19$
5	3	80	0.061	17
4	5	691	0.102	15
3	9	1	0.183	13
2	49	1	1	11

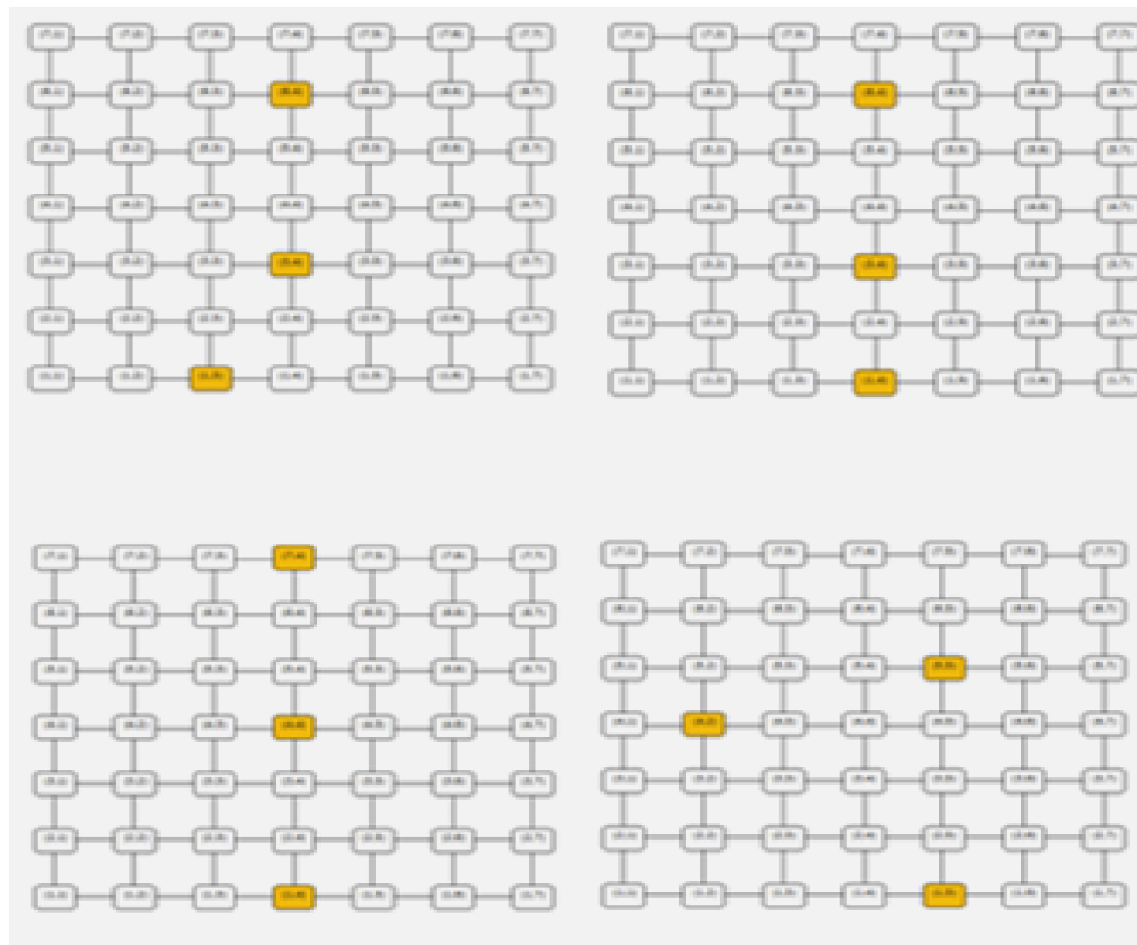
Mesh Size	Minimum Number of Junctions (H=6)
7x7	2
8x8	3
9x9	3
10x10	4

Mesh Size	Minimum Number of Junctions (H=6)	Minimum Number of Junctions (H=5)
7x7	2	3
8x8	3	4
9x9	3	4



Multiple Configurations of Junctions for a Given Path Length

Satisfaction of some other criteria like layout uniformity or optimization of performance in the context of application specific communication.





Increase in Path Length

- The average increase in communication overhead

- Overhead =
$$\frac{\sum_{i=1}^M \sum_{j=1}^M V_{ij} (JD_{ij} - D_{ij})}{\sum_{i=1}^M \sum_{j=1}^M V_{ij} D_{ij}}$$

- JD_{ij} = Distance between node i and node j using Junction based routing
- D_{ij} = Minimum distance between node i and node j
- V_{ij} = Communication volume between node i and node j
- M is the total number of nodes in the network



Overhead is very small!!

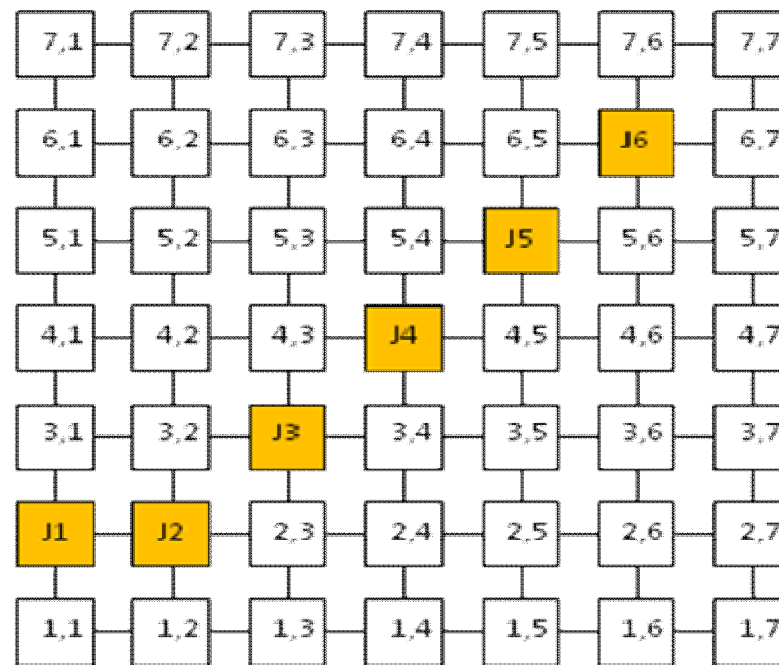
On a 7x7 mesh NoC with path length limit of 6 hops where the communication volume for each pair is a random number in the range of 1 to 10:

- For uniform random traffic, the average increase in overhead for different configurations varies from 0.05% to 3%.
- For traffic favoring locality, the average overhead for different configurations vary from 0.01% to 0.09%.



Path Computation

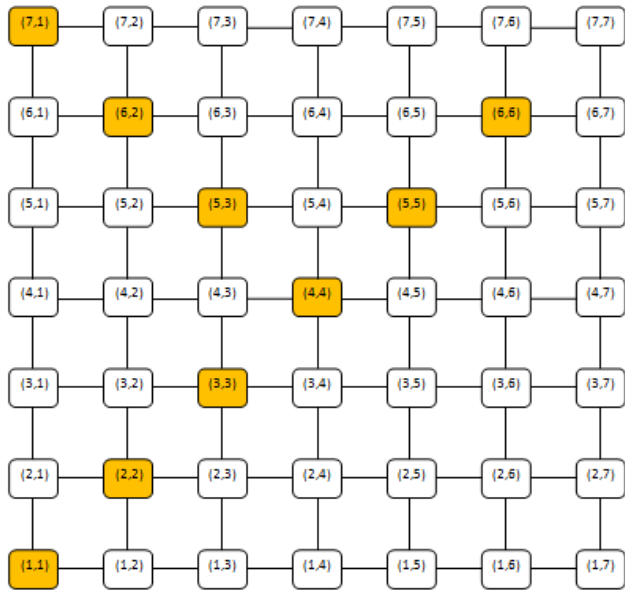
- Considering the routes allowed by a particular deadlock-free routing in the procedure of determining number and position of junctions.
- More junctions in the network.
- **Using Turn-Model routing algorithms and minimal paths solves the problem of increasing in paths lengths.**
- Using information about traffic patterns based on the communication requirements of the application during the path selection process.



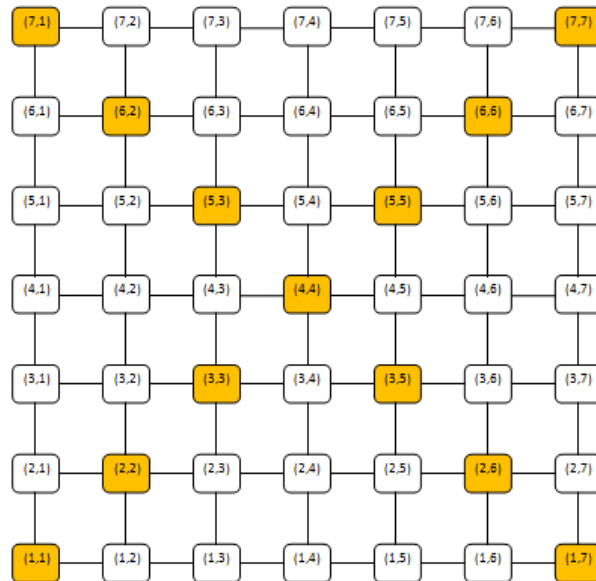
Odd-Even Routing Algorithm and a Hop Count Limit of 7



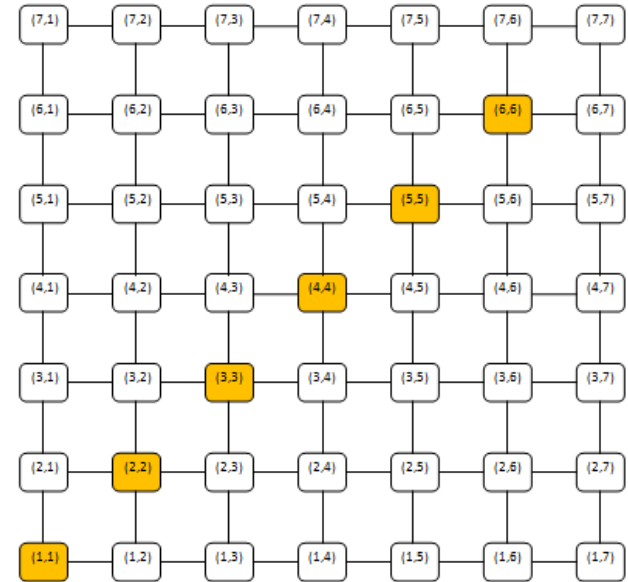
Configurations of Junctions which Support Deadlock-Free Routing



West-First



XY



Negative-First



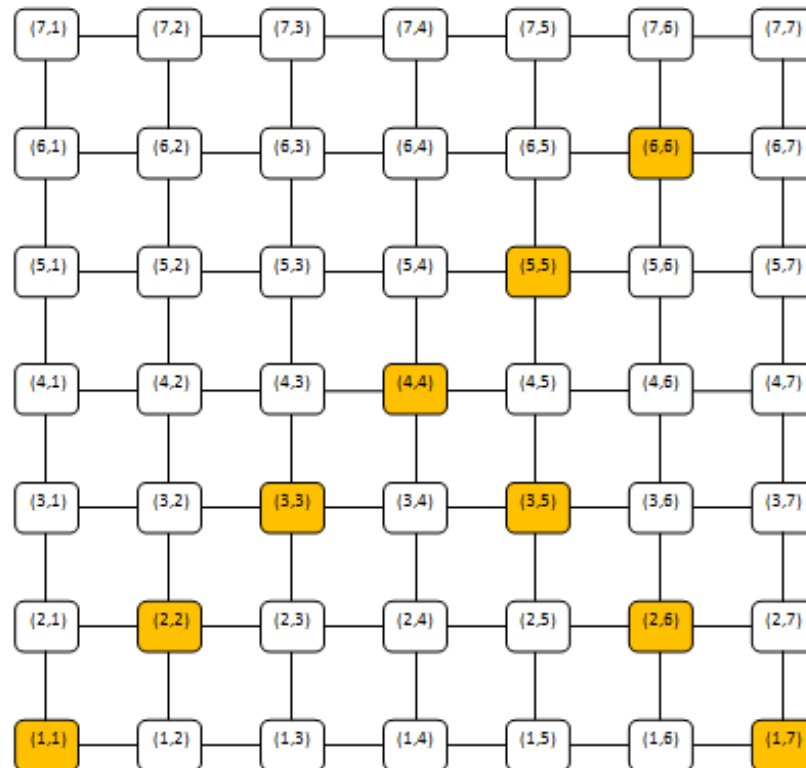
North-Last Routing Algorithm

- We define junction ratio as follows:

$$NJ/NN = \text{Number of Junctions} / \text{Number of Nodes} = 9/49 = 0.18$$

$$PJBR/PSR = \text{Number of Paths in JBR} / \text{Number of Paths in Source Routing} = 0.93$$

- A high value shows that JBR retains high path adaptivity.





Odd-Even Routing Algorithm



Different routing algorithms require different number of junctions but **it is still a small fraction of the total number of nodes. For the same value of hop-count, the ratio of junctions to nodes decreases as the size of NoC increases.**



Load Balance among Links

- A tool computes the required number of junctions and their positions for a mesh NoC for any given routing algorithm.
- Input parameters are network size, hop count limit, routing algorithm and traffic pattern or application specific communication information.
- Output parameters are one path for all communicating pairs and load distribution parameters.
- A cost for each pair reflects the potential of the communication to cause load imbalance among links:

$$\text{Communication Cost} = \frac{(\text{Communication Bandwidth} * \text{Distance})}{\text{Path Adaptivity}}$$

- In a particular set up of 7x7 NoC using Negative-First routing algorithm:
 - The standard deviation of link loads were reduced by **16.5%** as compared to random selection of paths.
 - There was a reduction of **22.5%** traffic on the link with the maximum load.



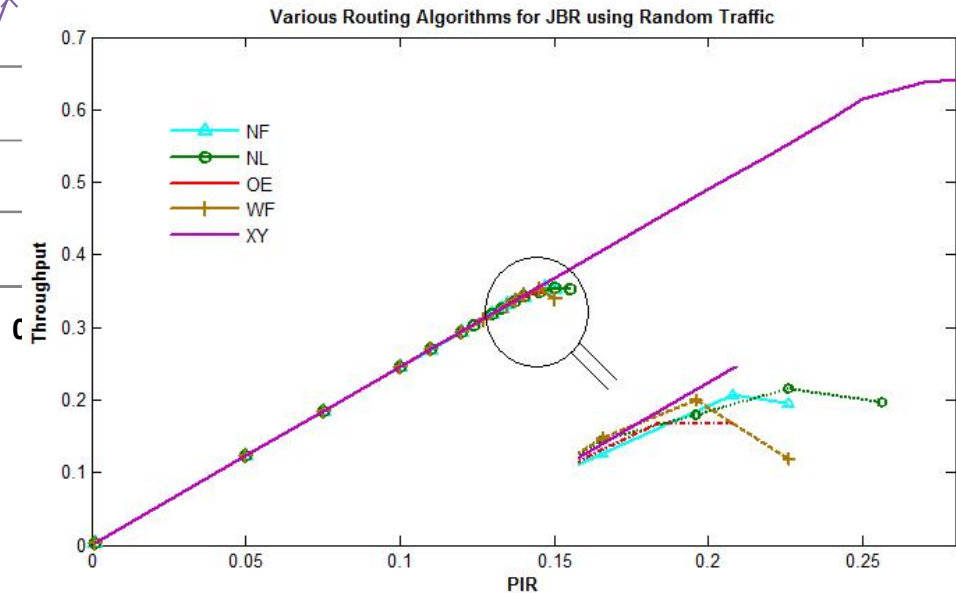
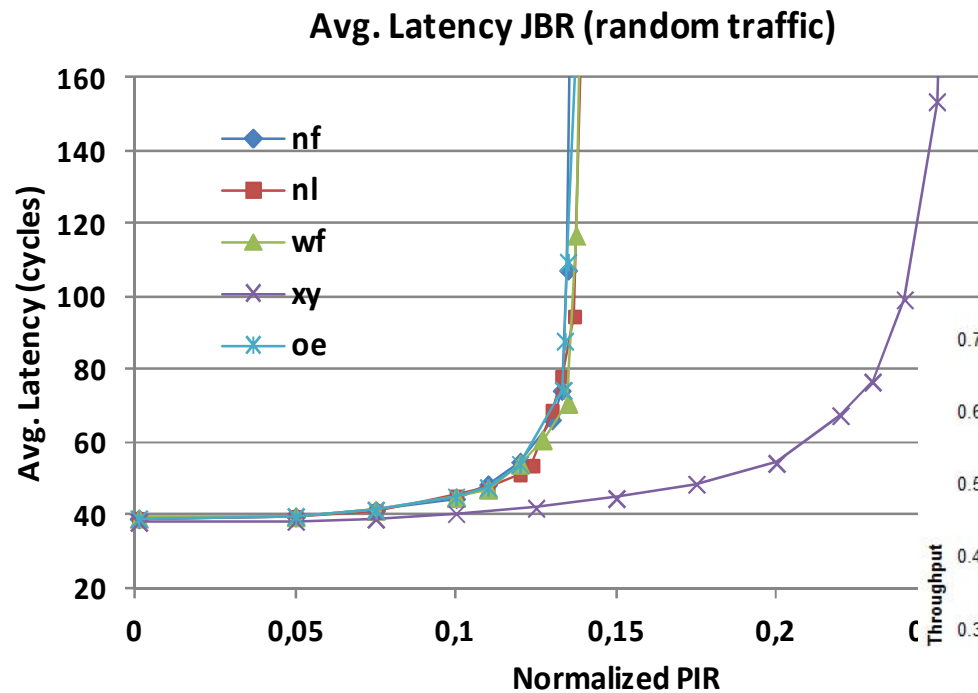
Evaluation of JBR

- A NoC simulator developed in SDL.
- Wormhole switching on a 7x7 mesh NoC for all experiments.
- JBR may require **fewer flits** to transport a given amount of payload.
- A header flit requires a minimum of 3 clock-cycles for traversing a router.
- One additional clock-cycle in case that packets require the service (i.e. a new path) of a junction.



Average Packet Latency in Random Traffic

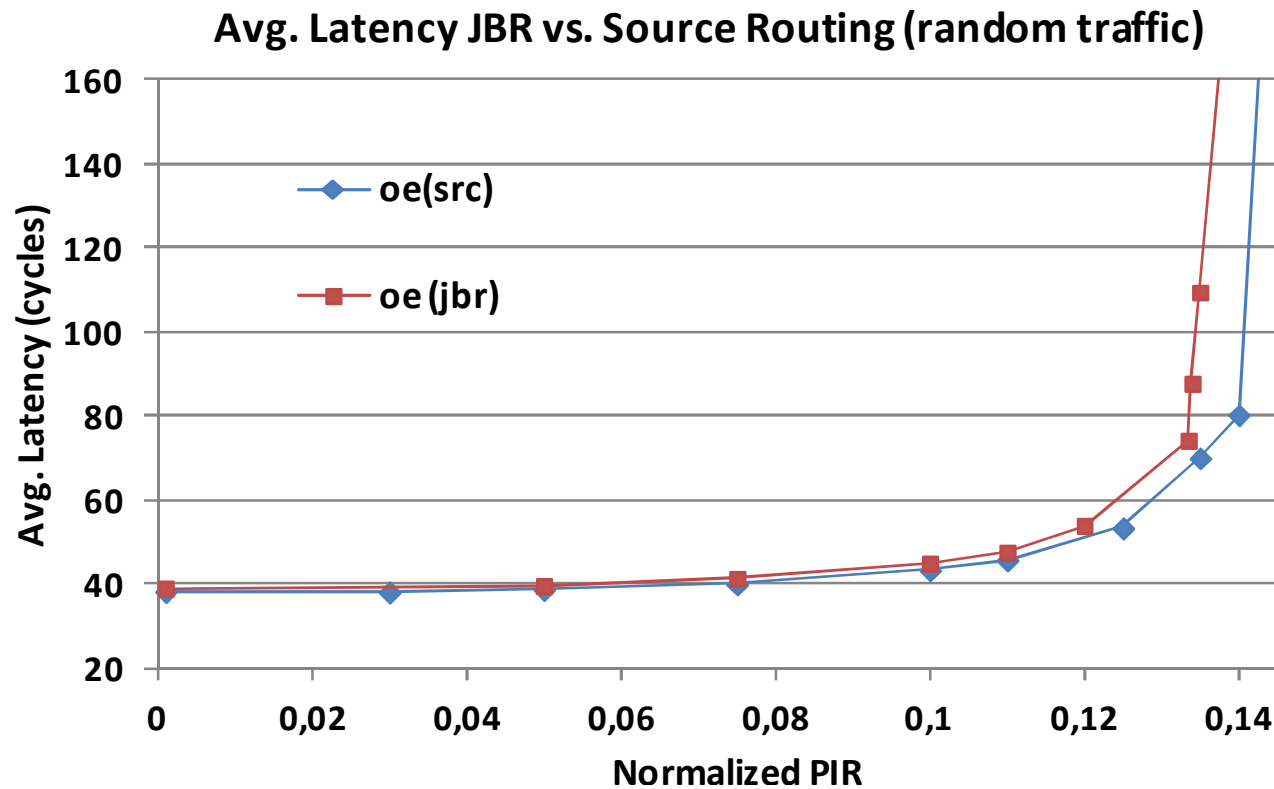
- The packet size of 10.
- **JBR correlates well with earlier work, where XY has shown superior performance in both distributed and source routing.**



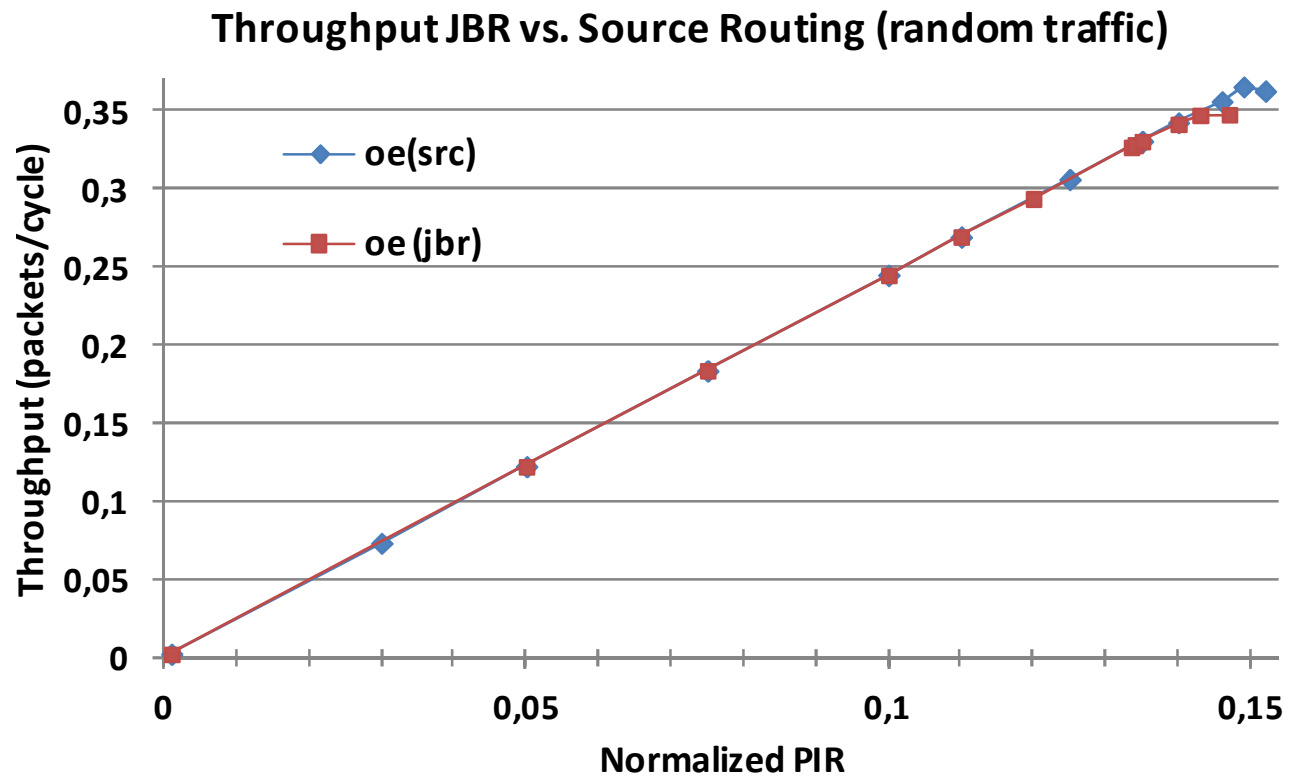


JBR and Source Routing

- The additional delay in junction routers does not cause a drastic performance penalty.
- The packet size equals 10 for both of them.



JBR and Source Routing (Throughput)

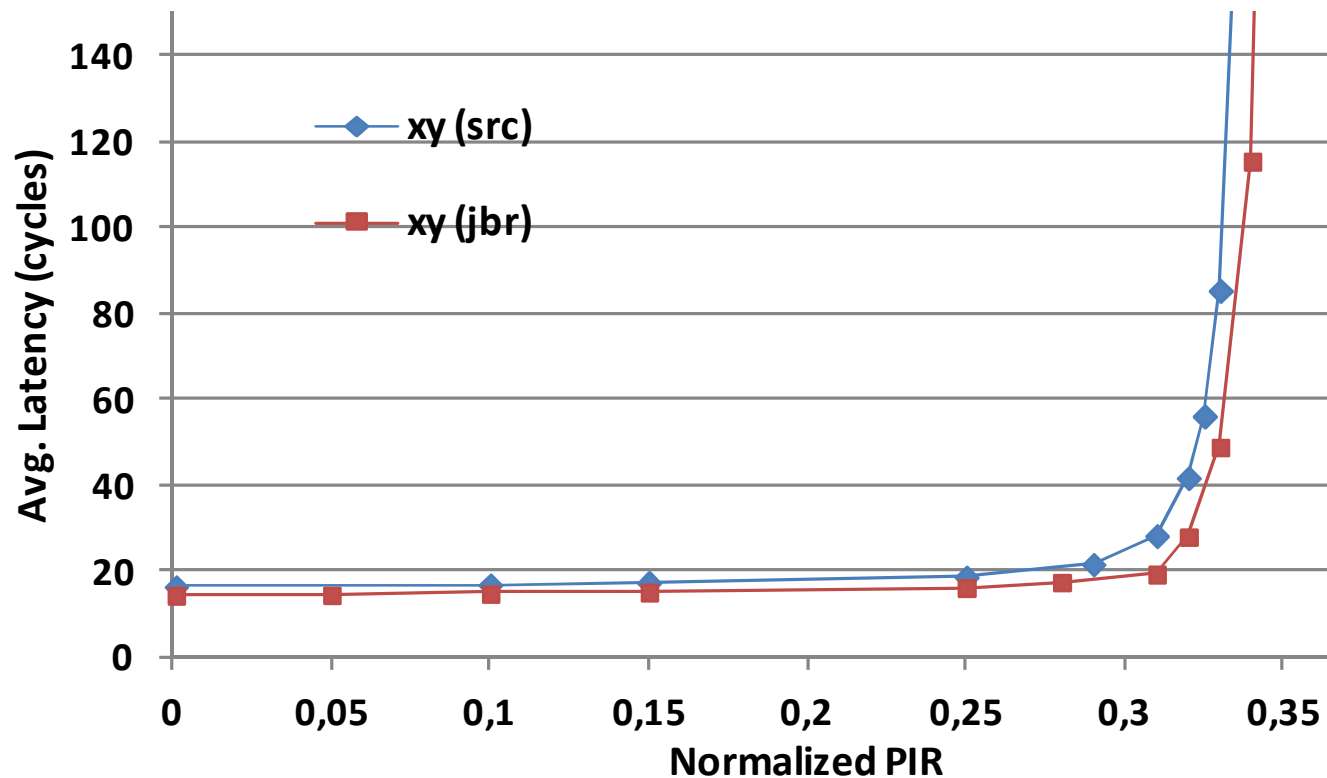




Packets with Small Payloads

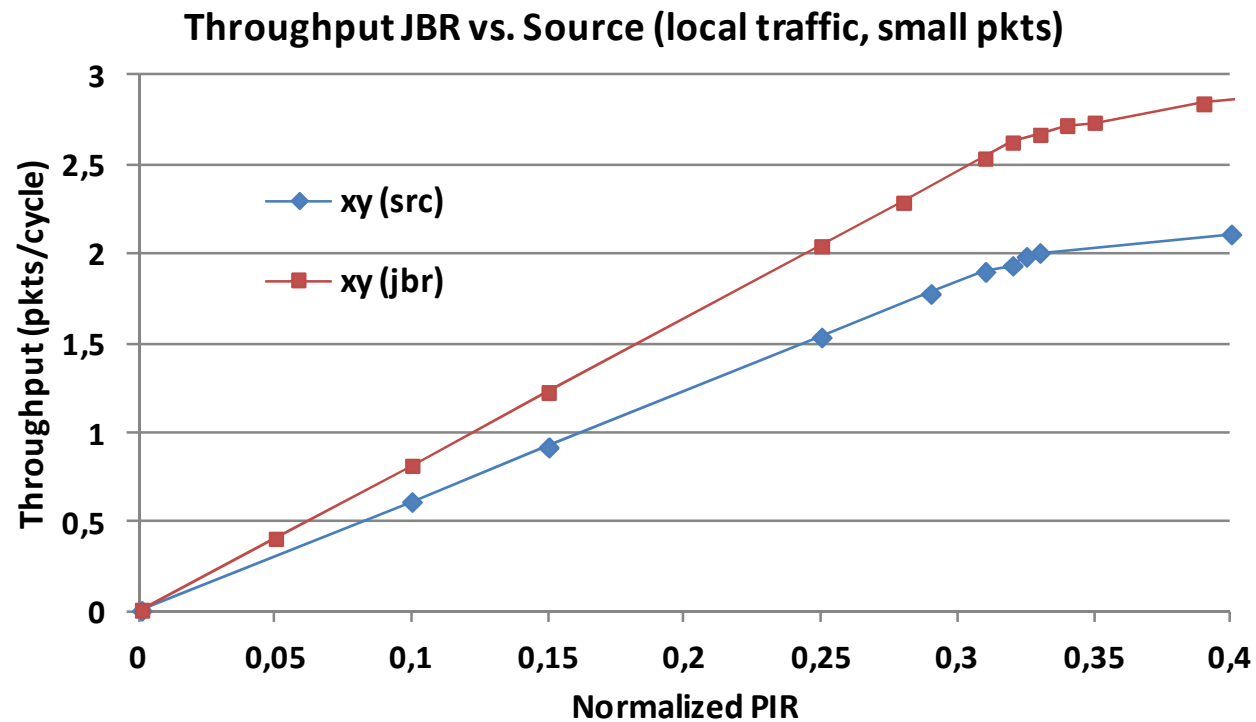
- **Better utilization of network resources.**
- Considering a 7x7 NoC with a hop count limit of 7 hops, the header flit can carry up to 11 bits of payload and with a hop count limit of 4 hops, the header flit can carry up to 17 bits of payload.
- The packet size equals 3 flits for JBR and 4 flits for source routing (H=7)

Avg. Latency JBR vs. Source (local traffic, small pkts)





Packets with Small Payloads (Throughput)





Architectural Implications

- Path Table (PT): A major new memory component.
- Path Modifier: Because of this extra functionality in the router, pipelining its operation and its control becomes significantly more complex.
- Another alternative for routers in JBR: a single router design with a mode option to make it function as a junction or as a normal router. **A single router design allows us to omit the PT in the core or RNI.**
- The size of the table is of major concern.



Conclusions

- ✓ JBR makes source routing in large NoCs systematic, scalable and efficient.
- ✓ A tool has been developed to search for appropriate junction positions which can support deadlock-free routing when paths are computed using turn model based deadlock free routing.
- ✓ JBR has slightly worse performance as compared to pure source routing for packets with large payload.
- ✓ JBR performs better than source routing for packets with small payload.
- ✓ Paths computed using static XY routing algorithm give better performance as compared to the situation when paths are computed using adaptive routing algorithms for random traffic.



Future Work

- Prototyping a router to support JBR.
- Development of techniques to compress the path tables in routers.
- Using distributed routing schemes and/or using topologies other than regular mesh.

Thank you