香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

A Thermal-aware Application specific Routing Algorithm for Network-on-chip Design

Zhi-Liang Qian and Chi-Ying Tsui

VLSI Research Laboratory

Department of Electronic and Computer Engineering

The Hong Kong University of Science and Technology

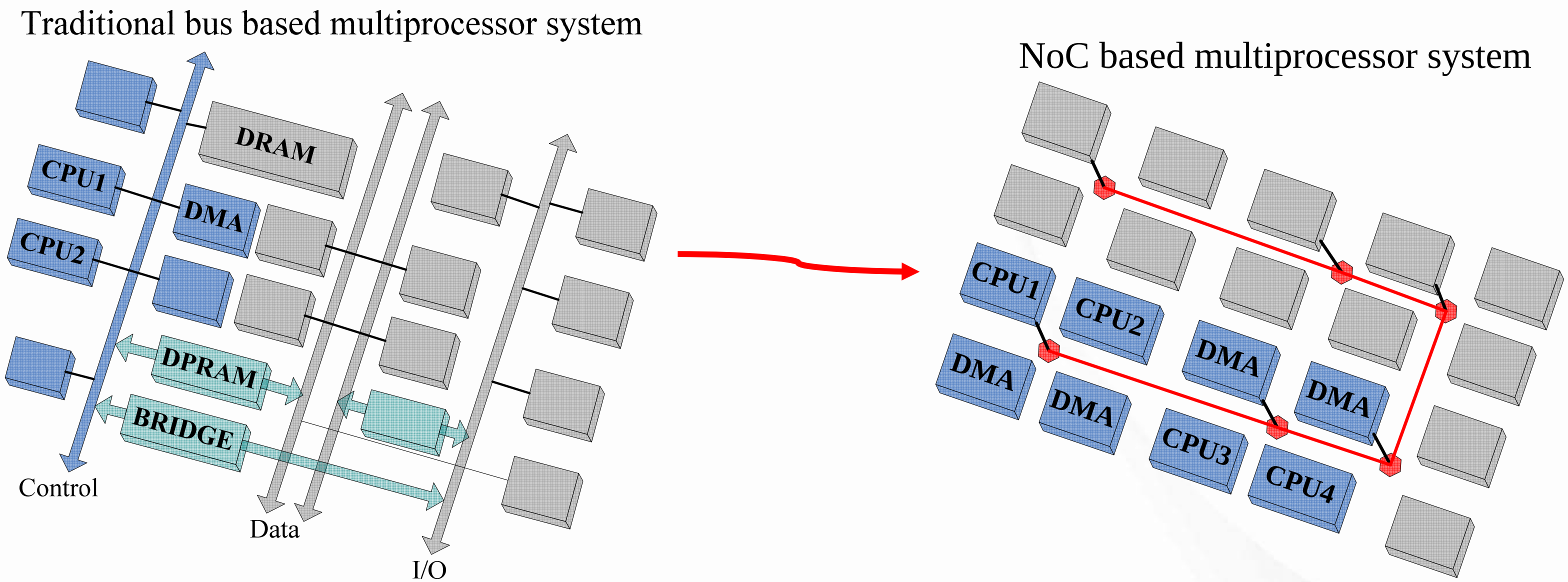
Jan 27th, Yokohama, Japan, ASP-DAC 2011

Outline

- **Introduction**
- Motivation
- Application-specific and thermal-aware routing overview
- Proposed routing algorithm
- Router Microarchitecture
- Experimental results
- Conclusions

Network-on-Chips (NoC)

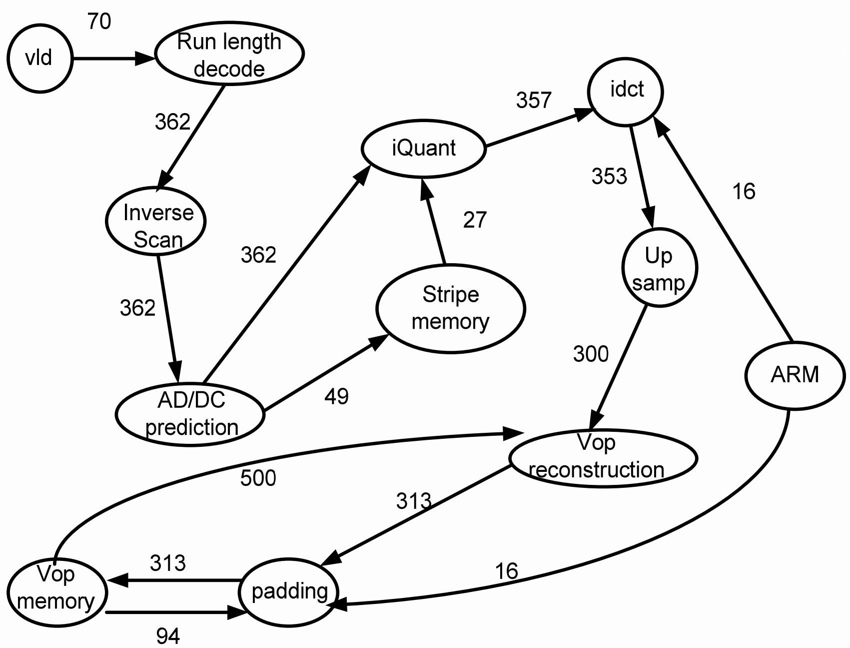
- Network-on-chips (NoC) : a scalable and modular solution for multiprocessor system-on-chip (MPSOC) design.

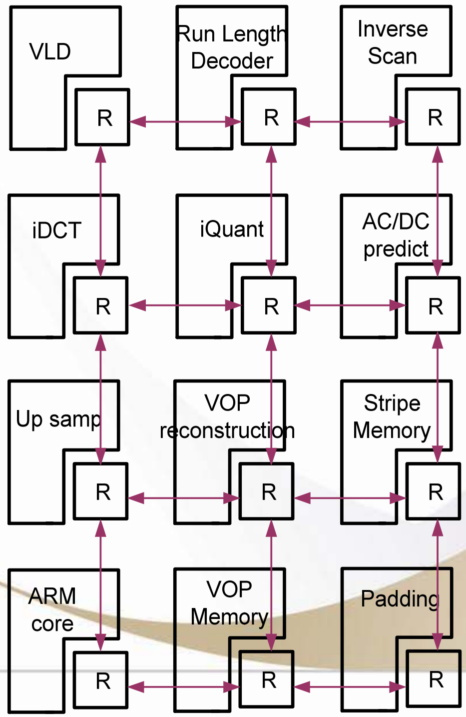


- Advantages of NoC : scalability / latency, power consumption / throughput / reliability etc.

Application-specific NoC

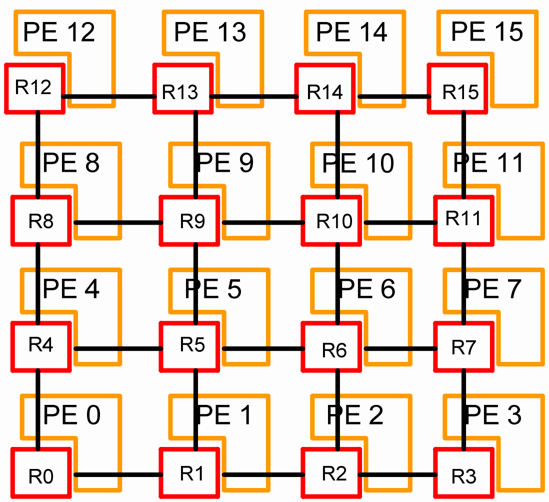
- NoC designed given a target application domain
  - The application is characterized by a given communication task graph (CTG).
  - Traffic information (communication pairs and volume) are obtained through profiling.
  - An example task graph and tile mapping for VOPD (Video Object Plane Decode) application:





Design Challenges for NoC

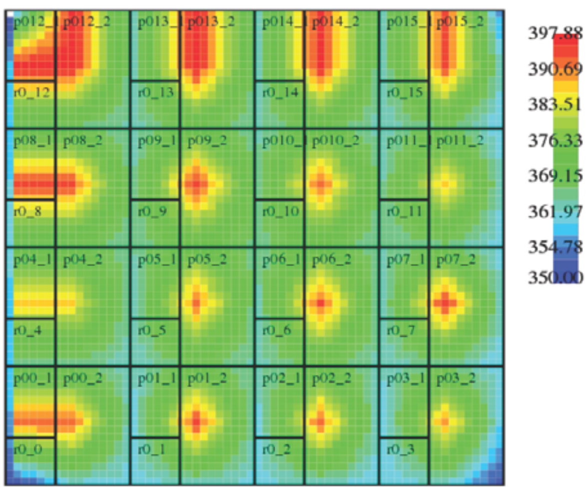
- Design constraints and objectives :
  - Energy and power consumption
  - Latency and throughput
  - Bandwidth requirement
  - Hardware implementation etc.
- Temperature and peak power have become the dominant constraints



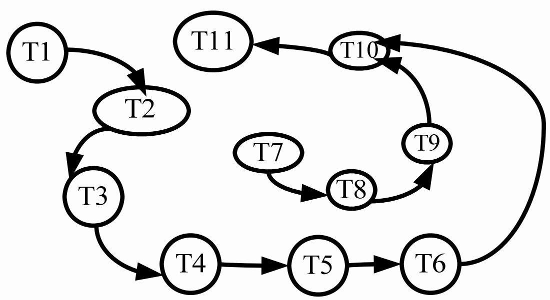
A typical runtime thermal profile

Temperature hotspot:

1) Degraded performance
2) Reduced reliability

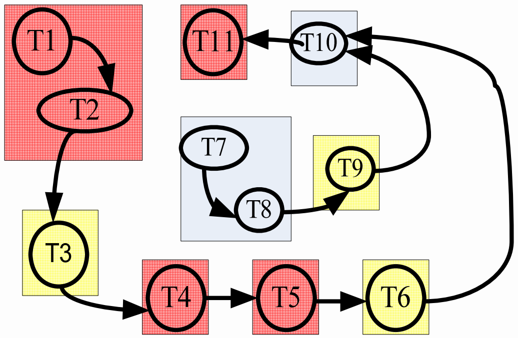


Application-specific NoC Design Flow



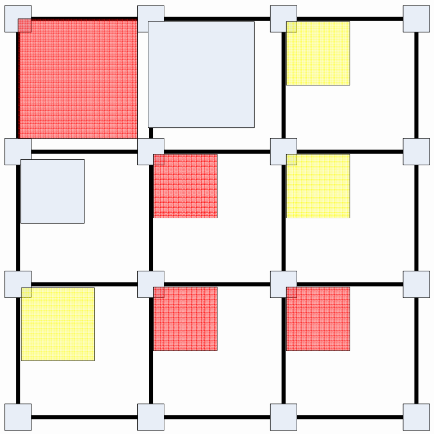
Task scheduling and allocation

Energy aware task scheduling/mapp ing [1]



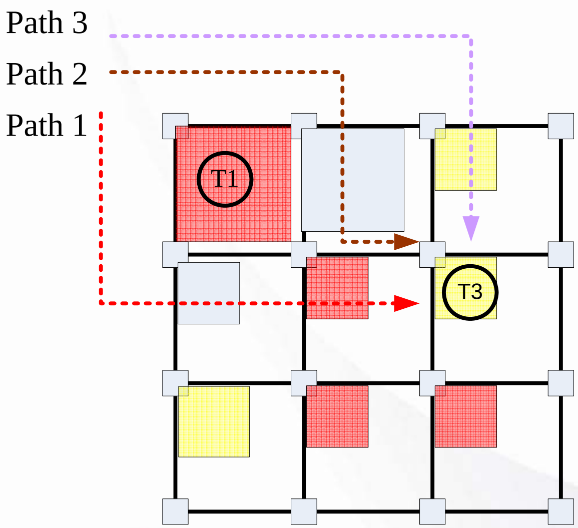
Mapping and Floorplanning

Energy /bandwidth / thermal aware placement [2]



Routing design

This work



[1] D.Bertozzi et.al " NoC synthesis flow for customized domain specific multiprocessor system-on-chip" IEEE Transactions on Parallel and Distributed Systems.16(2), pp.113-129, 2005

[2] H.Jingcao et.al " Energy- and performance- aware mapping for regular NoC architectures" IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 24(4), pp.551-562, 2005

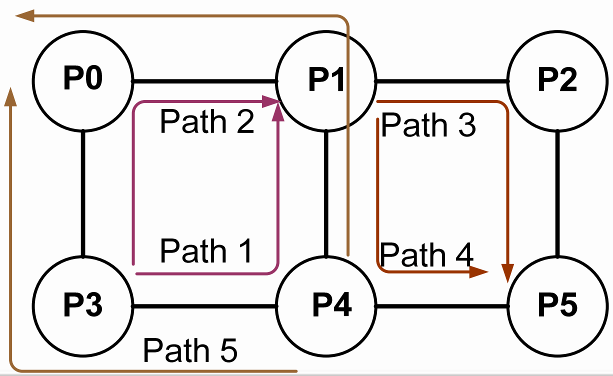
Previous Work

- Network-on-chip routing algorithm design:
  - Fault tolerant routing [3]
  - Bandwidth aware routing [4]
  - Limitations: temperature and thermal issues are not taken into consideration
- Thermal-aware NoC routing :
  - Ant-colony routing algorithm [5]
  - Thermal-region based routing [6]
  - Limitations: Generic routing algorithm is used; complex control schemes in [5]; deadlock avoidance issue in [6]

[3] D.Fick et.al “ A highly resilient routing algorithm for fault-tolerant NoCs” In Proc. DATE, pp.21-26, 2009
[4] M.Palsi et.al “Bandwidth-aware routing algorithms for networks-on-chip platforms” IET, Computer & Digital Techniques. 3(5). pp.413-429, 2009
[5] M.Daneshtalab et.al “NoC Hot Spot minimization Using AntNet Dynamic Routing Algorithm” In Proc. ASAP, pp. 33-38, 2006
[6] L.shang et.al “Temperature-Aware On-chip Networks” IEEE Micro. 26(1), pp. 130-139,2006

Adaptive Routing

- Deterministic routing : only one path provided for every communication pair
  - Path 1 ,Path 3 and Path 5 are provided (XY routing)
  - Simple but may introduce congestion and hotspots
- Adaptive routing : several paths dynamically selected within the router
  - Multiple paths can be used for routing
  - Distribute traffic more evenly



- Adaptive and minimal path routing is adopted in this work：
  - Reducing hotspot temperature
  - Maintaining the latency and throughput

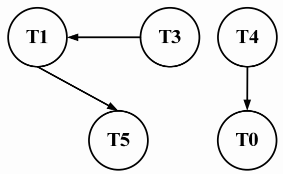
Outline

- Introduction
- **Motivation**
- Application-specific and thermal-aware routing overview
- Proposed routing algorithm
- Router Microarchitecture
- Experimental results
- Conclusions

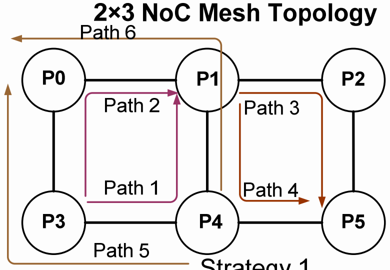
A Motivation Example

- Routing algorithm can be exploited to reduce the hotspot temperature
  - Communication network consumes a significant power budget (e.g. 39% in [8])

Communication graph



2×3 NoC Mesh Topology



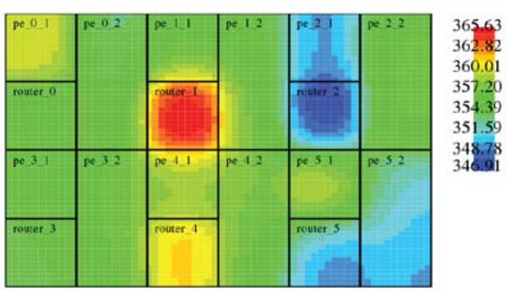
Strategy 1

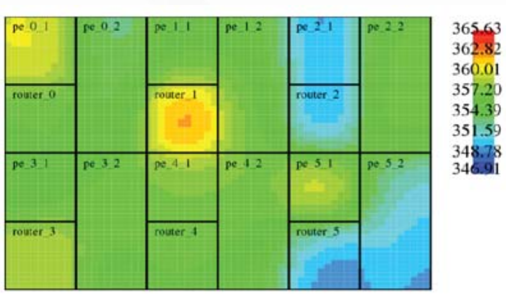
Two strategies of routing implementation

Communication pair	Path used in routing	Number of packets sent over the path	
		Strategy 1 (uniform traffic distribution)	Strategy 2 (allocating traffic non-uniformly)
P3 to P1	P3->P4->P1	500	400
	P3->P0->P1	500	600
P1 to P5	P1->P2->P5	500	850
	P1->P4->P5	500	150
P4 to P0	P4->P3->P0	500	900
	P4->P1->P0	500	100

Temperature profile simulation by Hotspot 5.0

Strategy 2



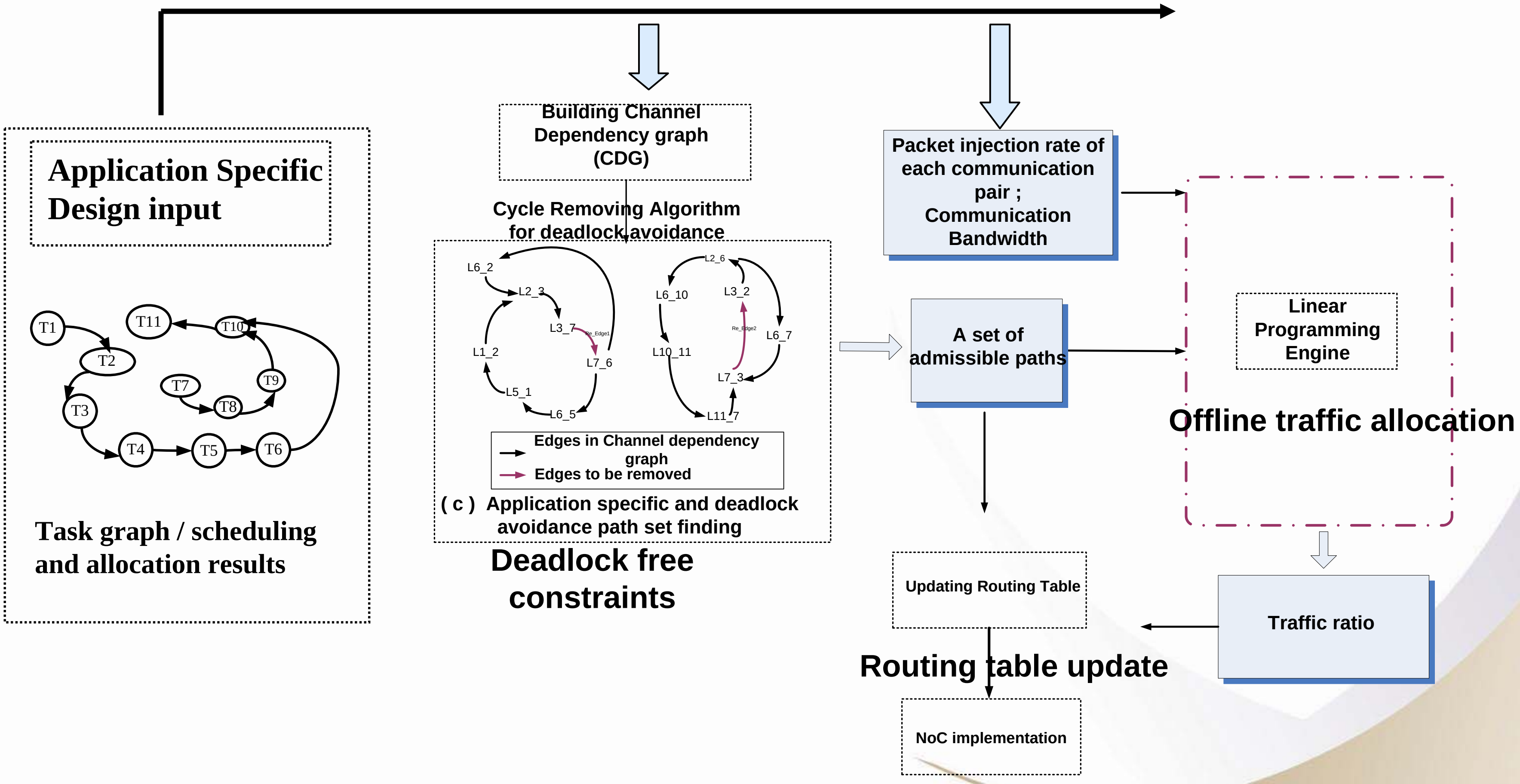


[8] S.Vangal et.al " A 5.1GHz 0.34mm2 router for Network-on-Chip Applications" In proc. IEEE Symposium on VLSI Circuit. pp. 42-43,2007

Outline

- Introduction
- Motivation
- **Application-specific and thermal-aware routing overview**
- Proposed routing algorithm
- Router Microarchitecture
- Experimental results
- Conclusions

Overview of the Proposed Application-specific Routing Algorithm



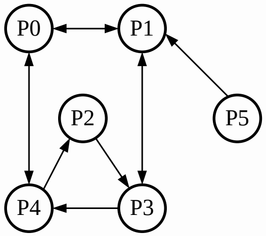
Outline

- Introduction
- Motivation
- Application-specific and thermal-aware routing overview
- **The proposed routing algorithm**
- Router Microarchitecture
- Experimental results
- Conclusions

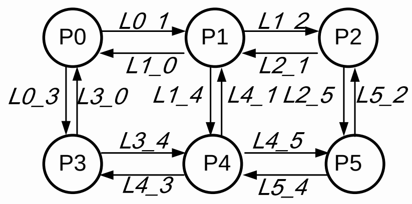
Deadlock Free Path set finding algorithm

- Using application traffic information can improve the adaptivity

Communication Graph

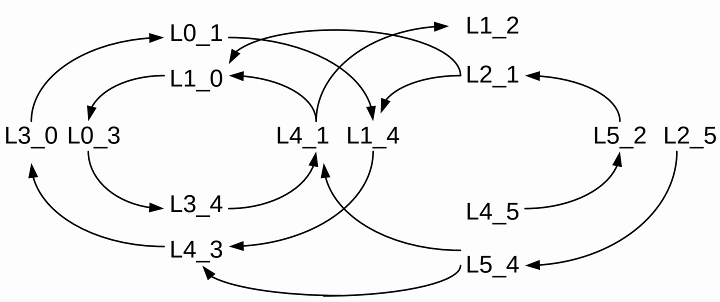


Topology Graph

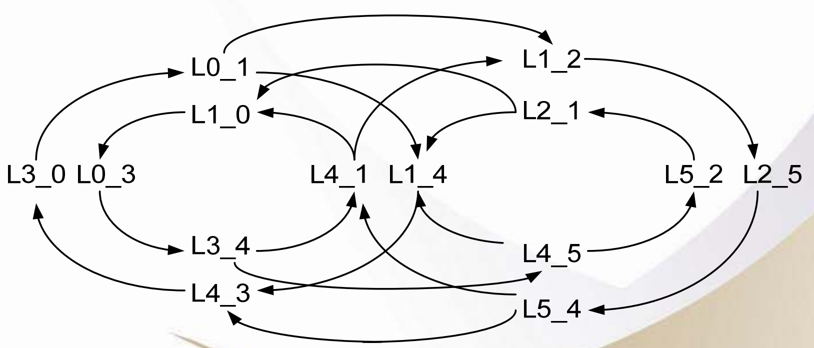


Total minimal paths: 18
Application specific :16
Westfirst :13
Northlast :14
Negativefirst :15
Oddeven :14

Application Channel Dependency Graph (CDG)

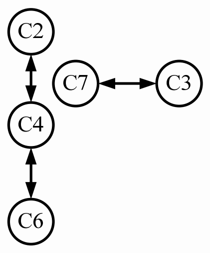


Channel Dependency Graph (CDG)

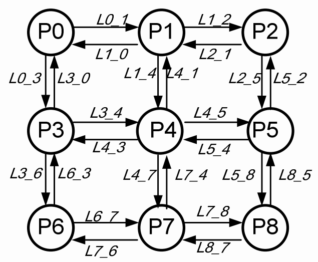


Deadlock Free Path set finding algorithm

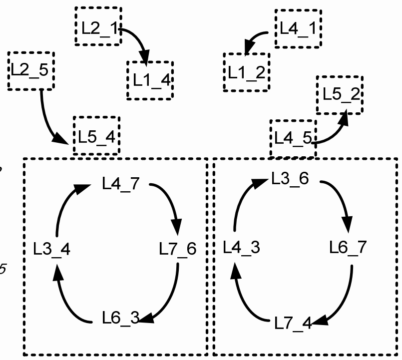
- Here we use an application specific and deadlock free path set finding similar to that used in [7]:



Core Communication graph



NOC mesh

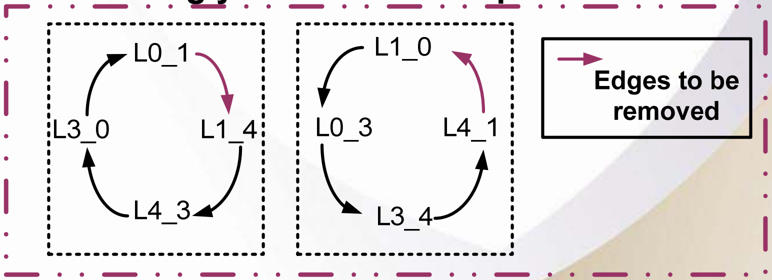


Channel Dependency Graph (CDG)
of the application communication

Find Circles in CDG

Circle removing algorithm

Step 3: Removing edge to break cycles in Strongly connected components



Deadlock free Path set

[7] M.Palesi et.al "Application Specific Routing algorithms for Network on chip" IEEE Transactions on Parellel and Distributed Systems. 20(3), pp. 316-330, 2009

Deadlock Free Path Set Finding Algorithm

- We modified the cost function from [7] :
  - Maximize the flexibility of re-divert traffic to even out the power distribution

$$\max \alpha = \max \frac{1}{|C|} \sum_{c \in C} \alpha_c W_c = \max \frac{1}{|C|} \sum_{c \in C} \beta W(c) \times \frac{|\Phi_{S_{edge}}(c)|}{|\Phi(c)|}$$

C: the set of communication pairs in the application

c: one communication pair

$\alpha_c = \frac{\#\, of \ \ paths \ provided \ \ for \ c}{\#\, of \ \ total \ \ paths \ existed \ in \ network}$: adaptivity of the communication pair c

α : average adaptivity of the communication

S_{edge} : the set of edges to be removed in the channel dependency graph (CDG) to break cycles

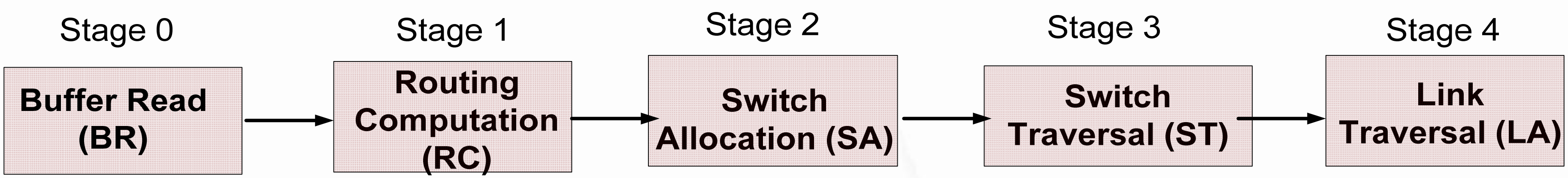
$\Phi(c)$: set of all minimal paths for communication c

$\Phi_{S_{edge}}(c)$: set of all minimal paths for communication c after edges in S*edge* being removed

Optimal Traffic Ratio Calculation

- Router energy consumption model:

Canonical Router Pipeline in NoCs

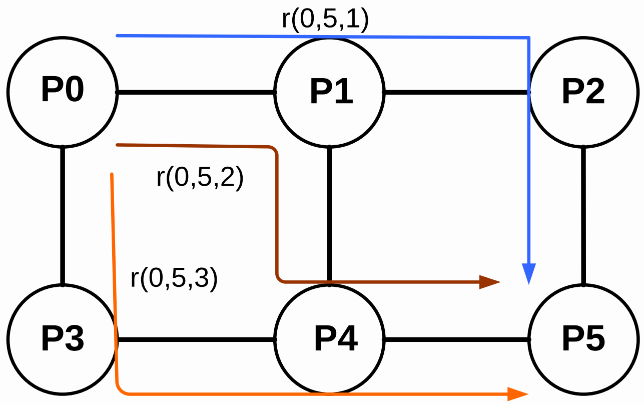


$$\underbrace{\Delta E_{r\_i}}_{\text{Energy consumption for routing a single packet}} = (\underbrace{E_{buffer\_rw}}_{\text{Buffer read and write}} + \underbrace{E_{forward}}_{\text{Forward a single flit}}) \times \underbrace{S_{packet}}_{\text{Packet size}} + \boxed{\underbrace{E_{rc}}_{\text{Routing computation}} + \underbrace{E_{sel}}_{\text{Output port selection}} + E_{vc}}$$

Wormhole head flit

Optimal traffic ratio calculation

- Problem formulation for optimal traffic ratio :
  - **Variables:** r(i , j, k) -- the ratio of using the kth path for sending packets between tile i and tile j



Three deadlock free paths are available for P0->P5:

Path 1: P0->P1->P2->P5

Path 2: P0->P3->P4->P5

Path 3: P0->P3->P4->P5

The path (a, b, k) passing through tile i

  - **Tile energy:** $E_i = E_{p\_i} + \Delta E_{r\_i} \times \boxed{\sum_{T_i} r(a,b,k) \times \underline{p(a,b)}}$

Traffic rate from source a to destination b

Optimal traffic ratio calculation

- LP Problem formulation for optimal traffic ratio:
  - Objective function: $obj \Rightarrow \min(\max(E_i))$
  - Problem constraints:

    - Traffic splitting constraints : Summation of all the traffic allocation ratios between a given pair (i, j) should equal to one.

$$\sum_{k=1}^{L_{i,j}} r(i, j, k) = 1 \quad \forall (i, j) \in C$$

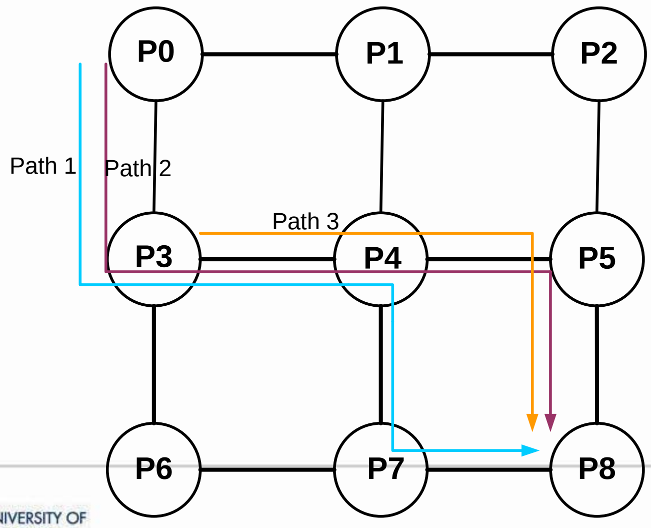
$$r(i, j, k) \geq 0 \quad \forall i, j \in [1, N], k \leq L_{ij}$$

    - Bandwidth constraints : the aggregate bandwidth should not exceed the link capacity

$$\sum_{T_i \cap T_j} \frac{r(a,b,k) \times p(a,b) \times S_{packet\_bit}}{T} \leq C_{ij}$$

Converting and Combining the Path Ratios

- Routing tables are used in the routing
  - routing decisions are made locally within the router
  - for minimal path routing , at most two candidate ports are available
  - the path ratios are converted into the local probability stored in the routing table for output port selection
  - Two types of routing table formats
    - Source destination pair
    - Destintation only



Routing table format in P4 (source-destination pair)

	Source id	Input port	Dst id	Output port	ratio
Path 1	0	W	8	S	P1
Path 2	0	W	8	E	P2
Path 3	3	W	8	E	P3

Routing table format in P4 (destination only)

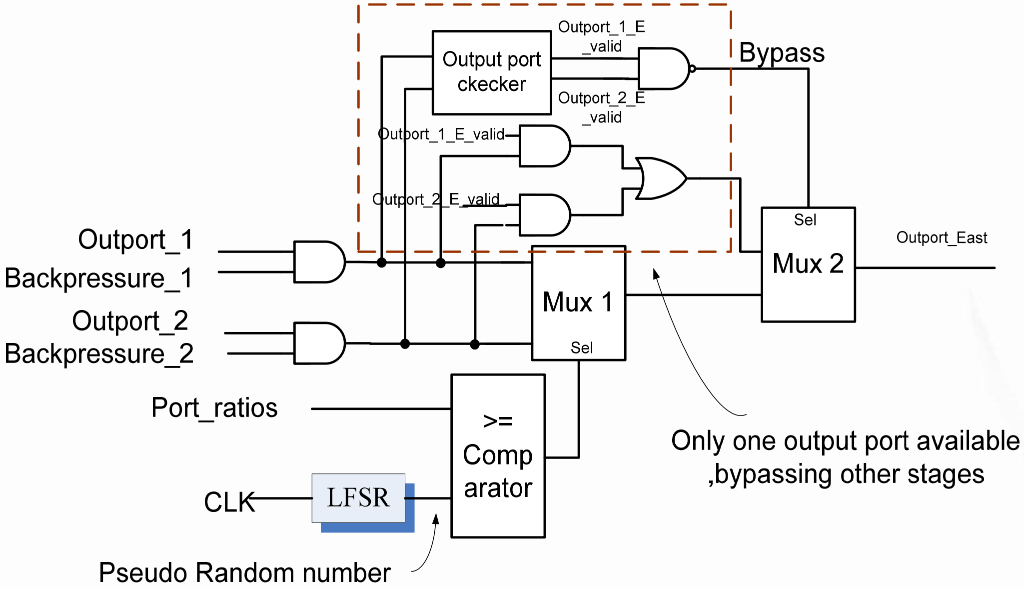
	Input port	Dst id	Output port	ratio
Path 1	W	8	S	P1
Path 2 **Path 3**	W	8	E	P2

Outline

- Introduction
- Motivation
- Application-specific and thermal-aware routing overview
- Proposed routing algorithm
- **Router Microarchitecture**
- Experimental results
- Conclusions

Router Microarchitecture

- Output selection hardware design



If (output_1 and output_2 available)

τ = output of LFSR

If τ < p(o1)

return o1

else

return o2

If (only one port available)

return the available one

– A pseudo random number generator using linear feedback register (LFSR) is employed

– If one output port is not available for routing due to limited buffer space etc. , the back pressure signal will disable the corresponding port from selection

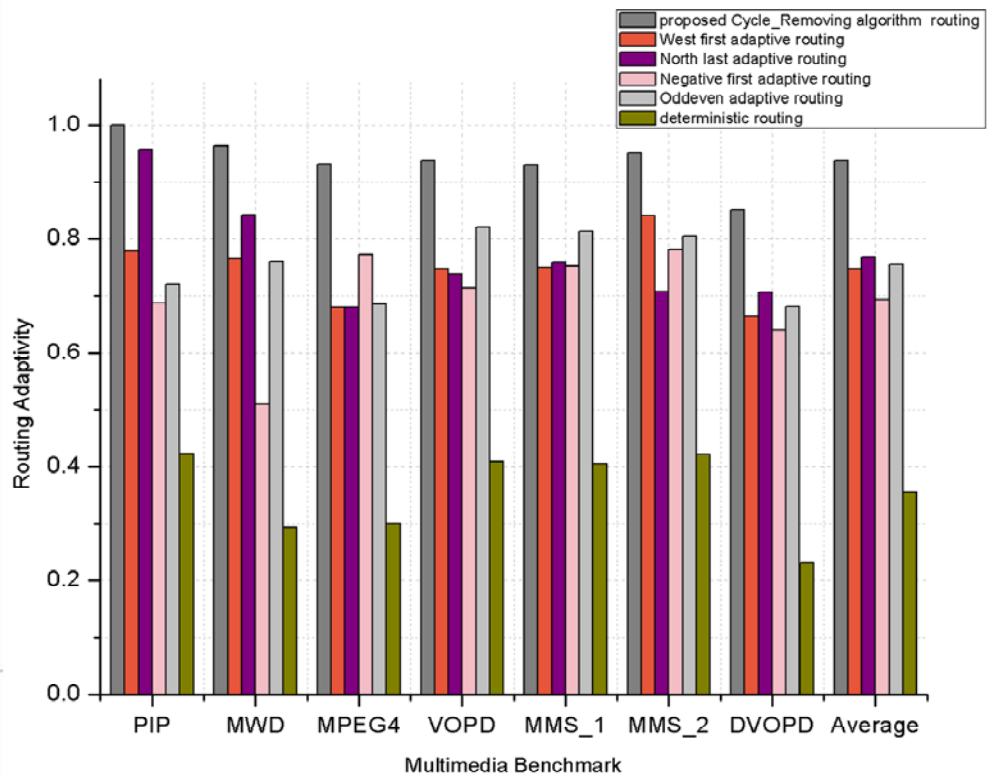
Outline

- Introduction
- Motivation
- Application-specific and thermal-aware routing overview
- Proposed routing algorithm
- Router Microarchitecture
- **Experimental results**
- Conclusions

Experimental Results

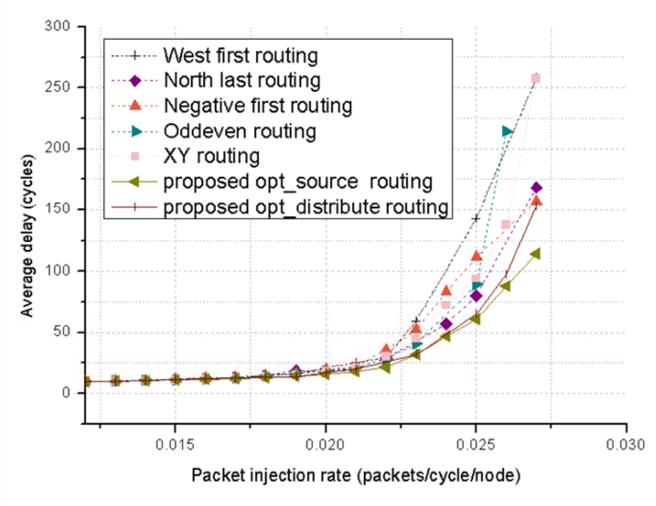
- Simulation environment setup
  - A C++ program is developed for the thermal-aware routing algorithm
  - A cycle accurate, flit-based NoC simulator,extended from Noxim, is used for simulation
  - Both synthetic traffic and real benchmarks are used for the simulation
    - MPEG4, VOPD, MMS

- Adaptivity comparison
  - 20%-30% more paths are available by consideration the application traffic information
  - Higher adaptivity will help to distribute the traffic more uniformly

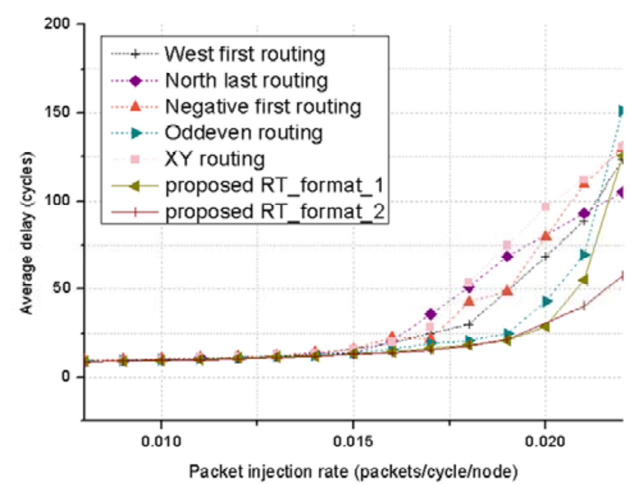


Latency Simulation- Synthetic Traffic

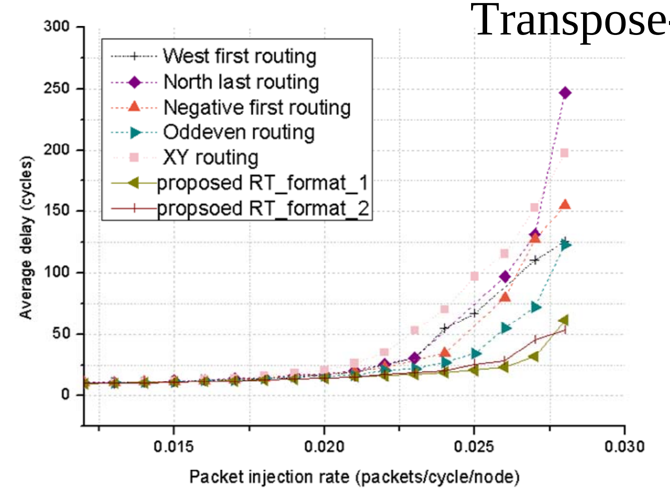
Random traffic



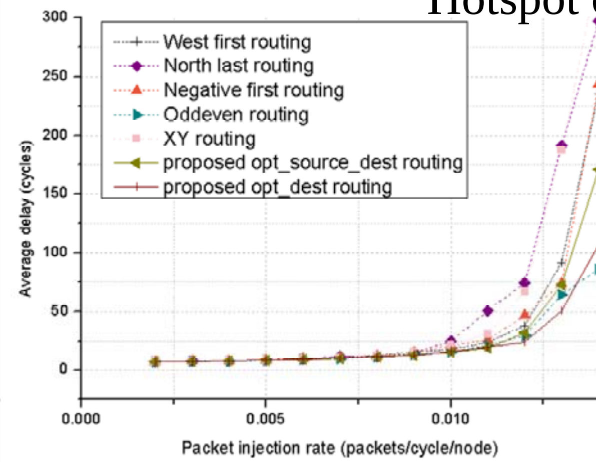
Transpose-1 traffic



Transpose-2 traffic

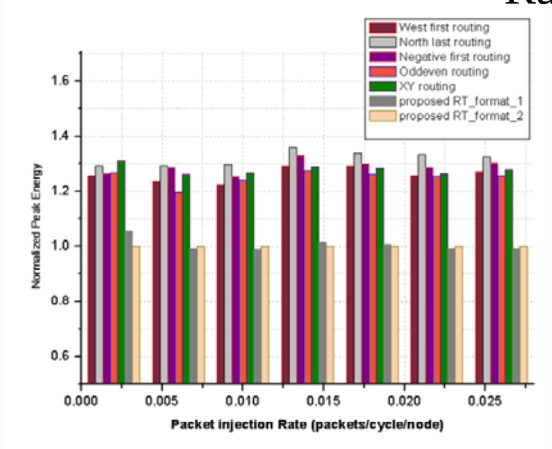


Hotspot center traffic

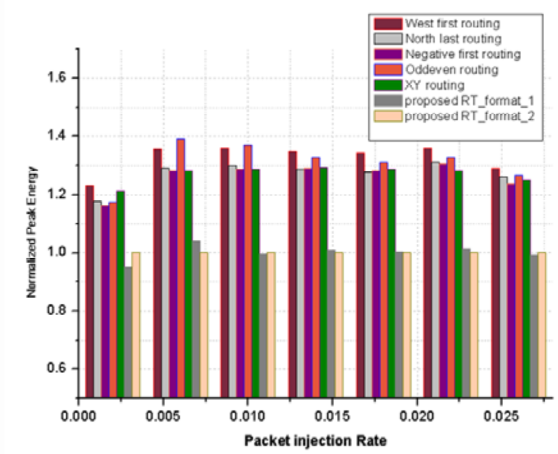


Peak Energy Simulation- Synthetic Traffic

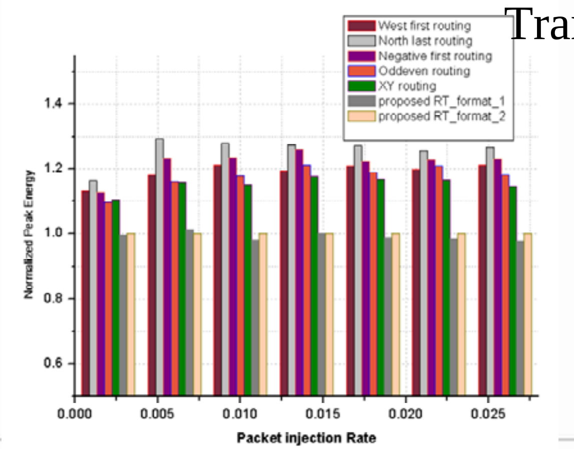
Random traffic



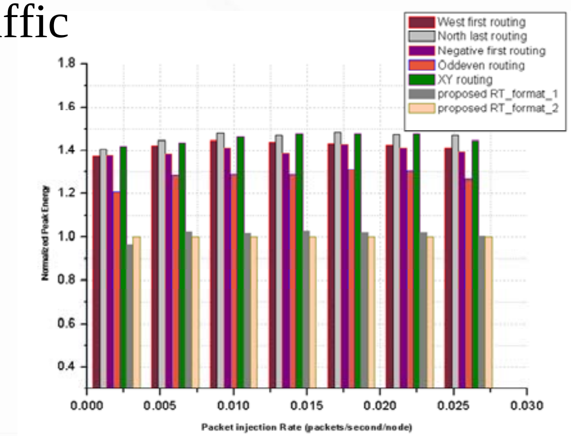
Transpose-1 traffic



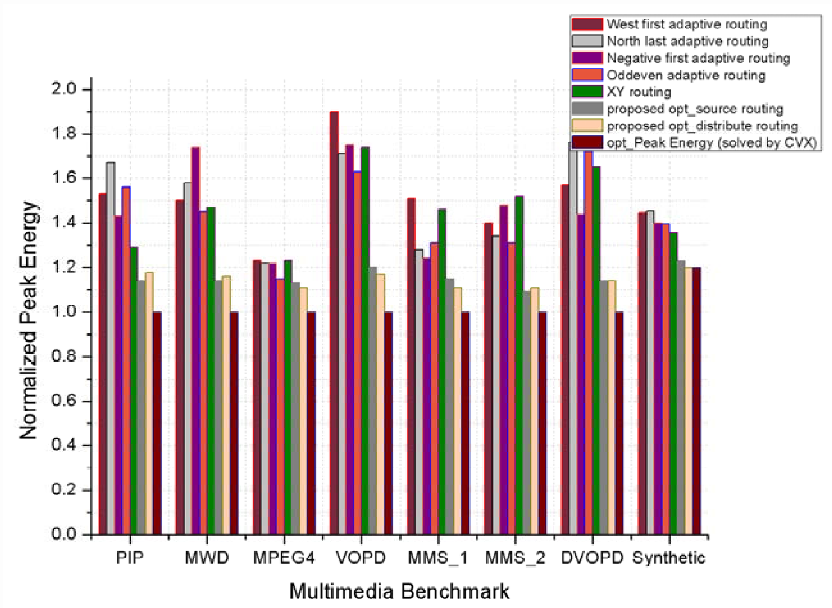
Transpose-2 traffic



Hotspot center traffic

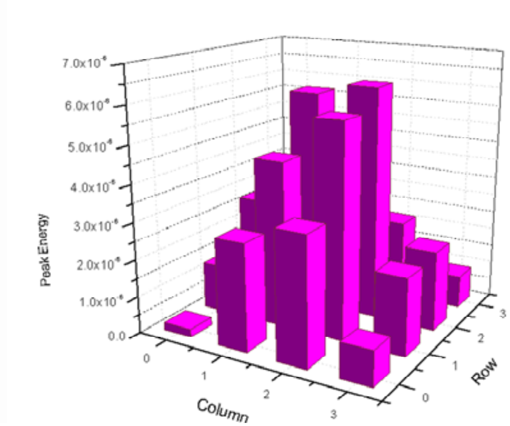


Peak Energy Simulation- Real Benchmark Traffic

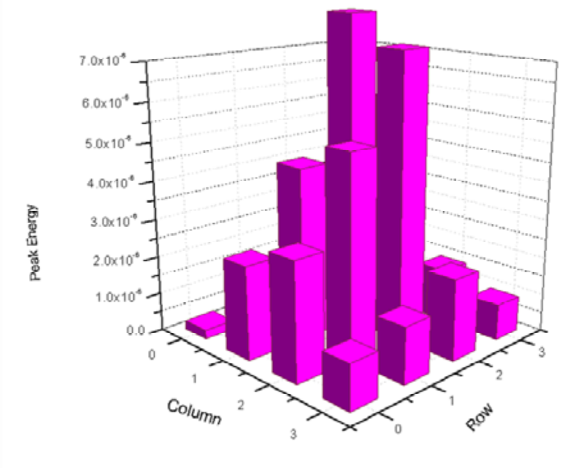


– In average, 16.6% peak energy reduction can be achieved

- Peak energy profile



Proposed



Oddeven

Peak Energy Reduction with Different PE/Router Ratios

- Tile's energy depends on both the routers and the processing element
  - We evaluate the effectiveness of the routing algorithm of reducing the peak energy when the energy ratio $r_e = \frac{Average\ processor\ energy}{Average\ router\ energy}$ varies.

	Peak energy reduction											
Synthetic Traffic	**Uniform random**		**Hotspot-center**		**Transpose-1**		**MMS-1**		**VOPD**		**Average**	
Average Energy ratio (r_e)	**vs. XY**	**vs. OE**	**vs. XY**	**vs. OE**	**vs. XY**	**vs. OE**	**vs. XY**	**vs. OE**	**vs. XY**	**vs. OE**	**vs. XY**	**vs. OE**
0.67	**17.4%**	**12.8%**	**15.7%**	**17.6%**	**15.6%**	**17.9%**	**17.7%**	**11.8%**	**16.5%**	**28.9%**	**16.6%**	**17.6%**
1.00	**15.7%**	**15.3%**	**14.4%**	**16.2%**	**13.3%**	**14.2%**	**15.7%**	**10.4%**	**12.3%**	**23.6%**	**14.3%**	**15.7%**
1.67	**10.6%**	**8.6%**	**12.3%**	**14.0%**	**11.2%**	**13.8%**	**10.6%**	**6.5%**	**9.3%**	**17.7%**	**10.9%**	**12.0%**
2.00	**11.9%**	**9.3%**	**11.6%**	**15.0%**	**11.1%**	**13.8%**	**10.6%**	**6.5%**	**9.3%**	**17.7%**	**10.9%**	**12.0%**
2.67	**9.4%**	**7.7%**	**10.2%**	**13.0%**	**8.9%**	**11.1%**	**10.4%**	**7.0%**	**7.6%**	**14.8%**	**9.3%**	**10.5%**
3.00	**8.8%**	**7.0%**	**9.6%**	**10.9%**	**8.8%**	**10.6%**	**9.7%**	**6.5%**	**7.6%**	**14.2%**	**8.9%**	**9.7%**
3.67	**8.1%**	**6.5%**	**8.7%**	**9.9%**	**7.8%**	**9.2%**	**7.9%**	**5.2%**	**7.1%**	**13.0%**	**7.9%**	**8.6%**
4.00	**7.9%**	**6.1%**	**8.3%**	**9.4%**	**7.0%**	**9.0%**	**7.3%**	**4.7%**	**6.5%**	**12.0%**	**7.4%**	**8.1%**

Outline

- Introduction
- Motivation
- Application-specific and thermal-aware routing overview
- Proposed routing algorithm
- Router Microarchitecture
- Experimental results
- **Conclusions**

Conclusions

- In this paper, we propose an application-specific and thermal aware routing algorithm for network on chips.
- Given the application traffic characteristics, a set of deadlock free paths with higher adaptivity is first obtained for routing.
- A LP problem is formulated to allocate the traffic properly among the paths.
- A table based router is also proposed to select the output ports according to the ratios
- From the simulation results, the peak energy reduction can be as high as 16.6% for both synthetic traffic and industry benchmarks.