Maurizio's Research Activity (At a glance)

Maurizio Palesi

UniCT (www.unict.it)

- 12 Faculties
- Faculty of Engineering
  - ➔ 7 Departments
    - ✓DAU: Architecure and urbanistic
    - ✓DIEES: Electrical, Electronic and Systemics
    - ✓DICA: Civil
    - ✓DIIM: Informatics and Mathematics
    - ✓*DIIT: Informatics and Telecommunications*
    - ✓DM: Mathematic
    - ✓DMFCI: Phisical and Chemistry

DIIT (www.diit.unict.it)

- **16** professors, **6** researchers, **5** assistant researchers, **15** Ph.D. students
- Research groups
  - ➔ Telecommunication
  - ➔ Computer science
- Topics
  - ➔ Computer architectures, Signal processing, Operating systems, Artificial intelligence, Networking, Industrial informatics, …

Team

- Vincenzo Catania, *Full Professor*
- Giuseppe Ascia, *Associate Professor*
- Maurizio Palesi, *Assistant Researcher*
- Davide Patti, *Ph.D. Student*
- Alessandro G. Di Nuovo, *Ph.D. Student*
- Fabrizio Fazzino, *External Consultant*

Research Topics

- Instruction Level Power Estimation
- Design Space Exploration of Parameterized Systems
- Bus Encoding Techniques
- Area/Power/Performance Tradeoff Analysis of VLIW Architectures
- Network on Chip Architectures

Outline

- Design Space Exploration Techniques
  - ➔ Parameterized SoC platforms
    - ✓ Pruning the design space
    - ✓ Accelerating evaluation of a system configuration
- Network on Chip Architectures
  - ➔ Topological mapping
  - ➔ Application specific routing algorithms
  - ➔ Efficient selection schemes

Design Space Exploration

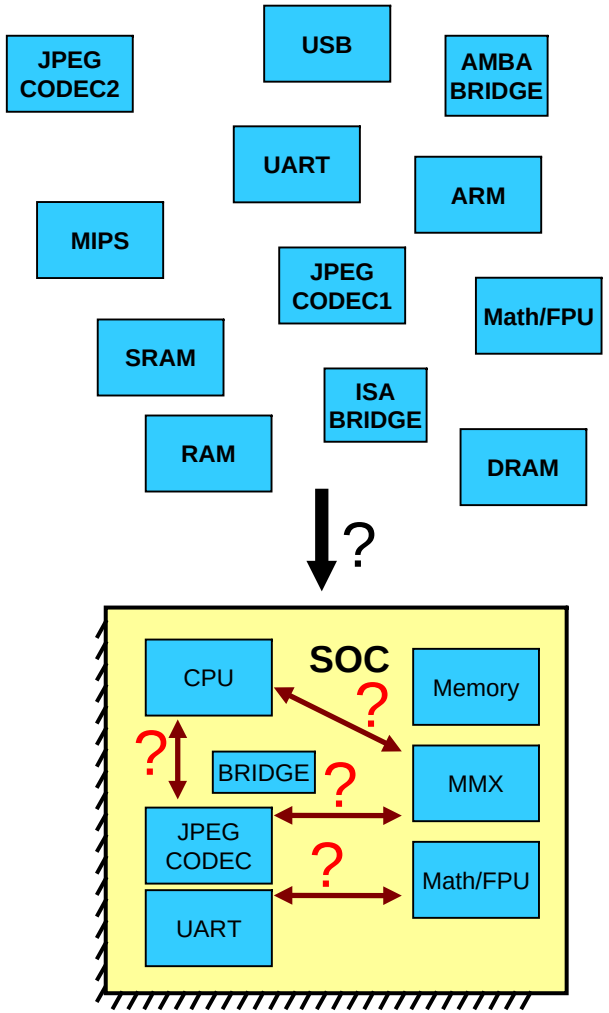
Trends

- Design trends
  - ➔Growing demand for portable devices
  - ➔Growing demand for low power design
  - ➔Increased application complexity
  - ➔Shrinking time-to-market windows
- Technology trends
  - ➔Increased chip capacity
  - ➔Increased I/O pins
  - ➔Improved on-chip integration techniques (storage, digital, analog, digital, …)
  - ➔SOC era

Need for greater design productivity!

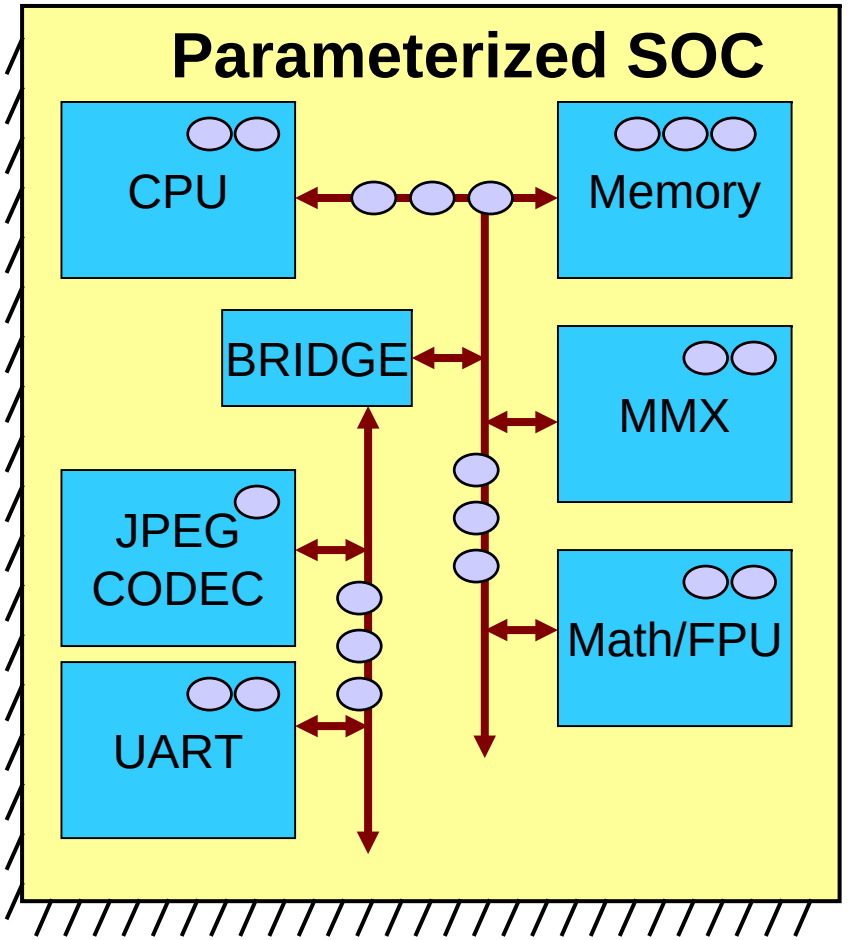
IPs reuse

- One approach: reuse of existing IP
  - ➔IP selection?
  - ➔IP integration?
  - ➔SOC verification?
  - ➔Multi-source IP licensing
  - ➔…



Configure-and-Execute

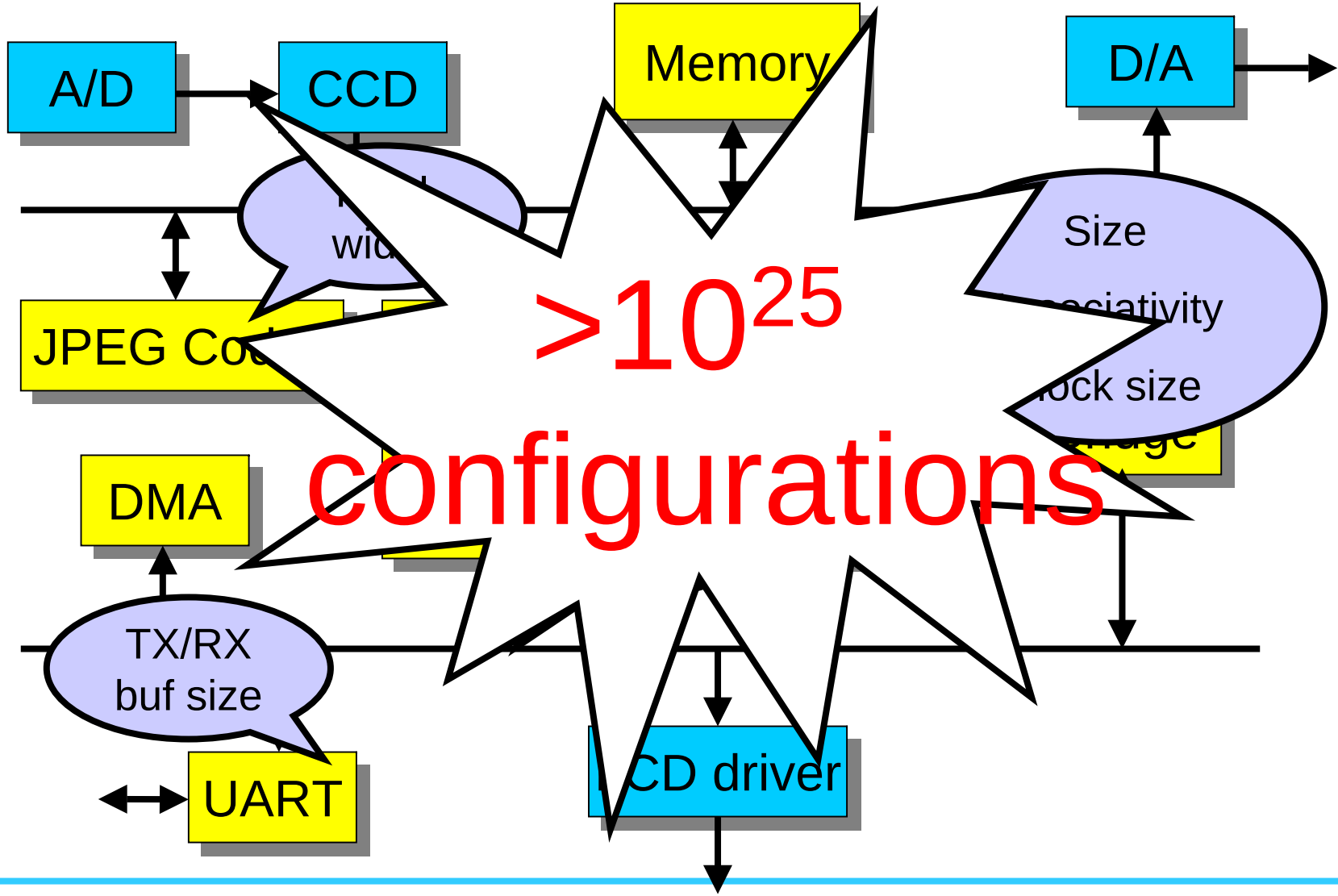
- Alternate approach: reuse of SOC
  - ➔ Designed, integrated, tested
  - ➔ Domain specific
  - ➔ *Parameterized*
- Designed by firms specializing in SOC
- User: map application, then, "*configure-and-execute*"



Parameterized Platforms

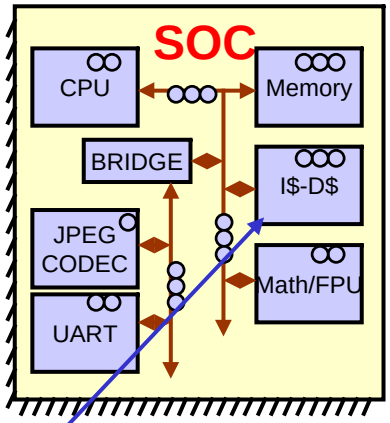
- ATI Technologies – XILLEON™ 220 SOC for Digital Set-top Box Market
- Tensilica – Xtensa™ 1040 configurable processor cores
- Philips Semiconductors – Nexperia™ SOC platforms
- Adelante Technologies – offers complete SOC customizable platforms for DSP domains
- More…

Sample SOC Platform for Digital Camera



DSE: Problem Formulation

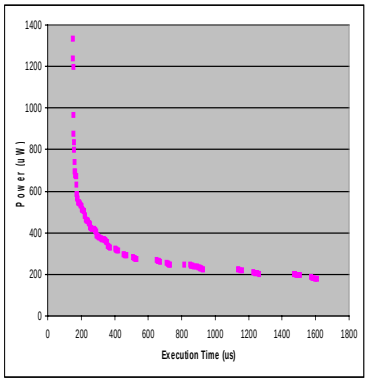
- Given
  - ➔ Parameterized SOC architecture
  - ➔ Fixed application
- Automatically explore the design space
- Find optimal points w/respect to power and performance




```
void main(){
 while(1){
  Receive();
  Decode();
  Display();
 }
}
```
Application

❖Size = {1K, 4K, 8K}
❖Line = {4, 8, 16}
❖Assoc = {1, 2, 4}

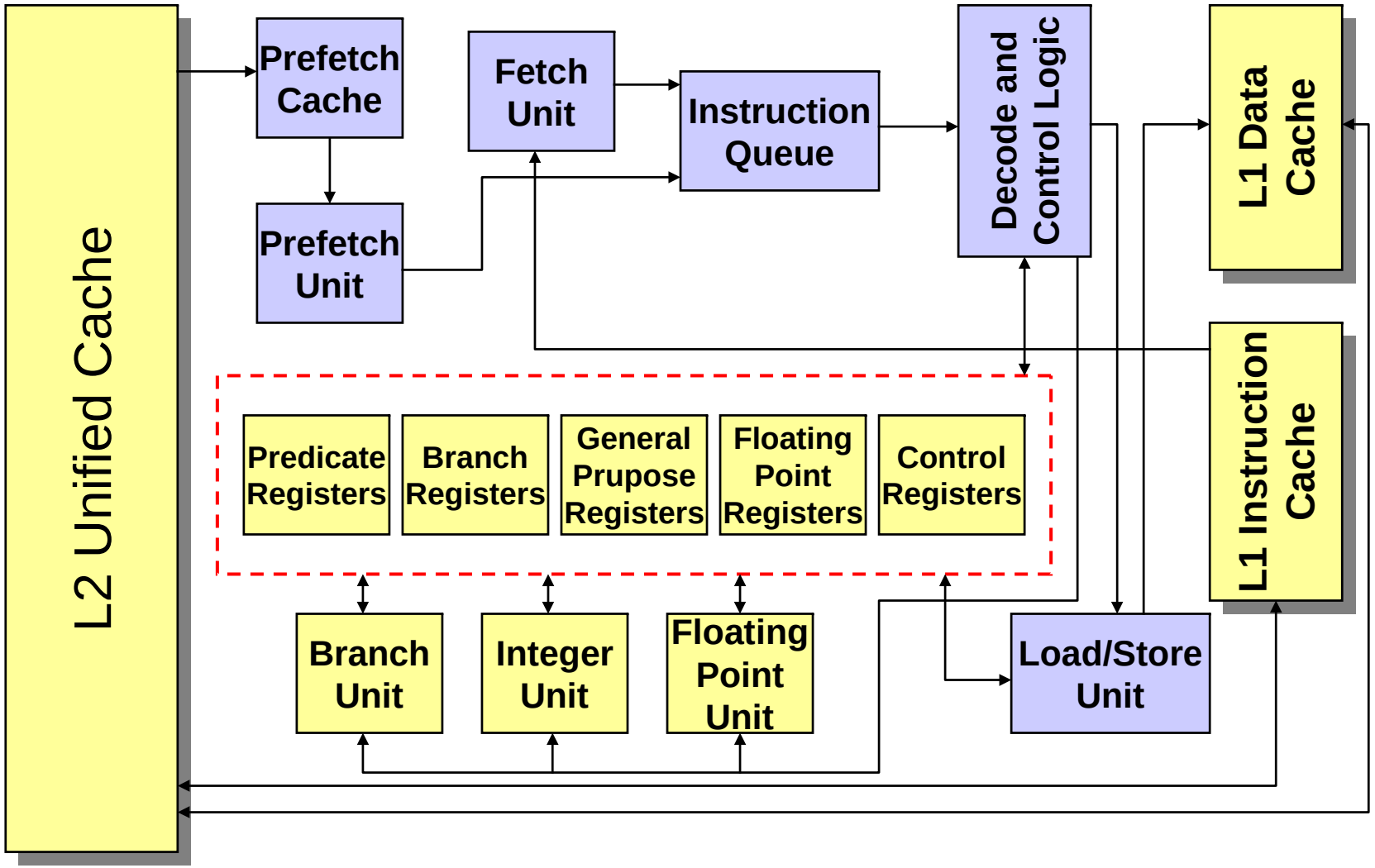
Explore



DSE Approaches

- Dependency analysis (dep) [Givargis *et al.* TVLSI02]
- GA-based DSE (ga) [Palesi *et al.* TCAD05]
- Sensitivity Analysis [Zaccaria *et al.* DAES02]
  - ➔Pareto-based Sensitivity Analysis (pbsa) [Palesi *et al.* IWSOC02]
- Dependency + GA (depga) [Palesi, Givargis CODES02]
- Sensitivity + GA (saga) [Palesi *et al.* ISCS02]

Reference Architecture

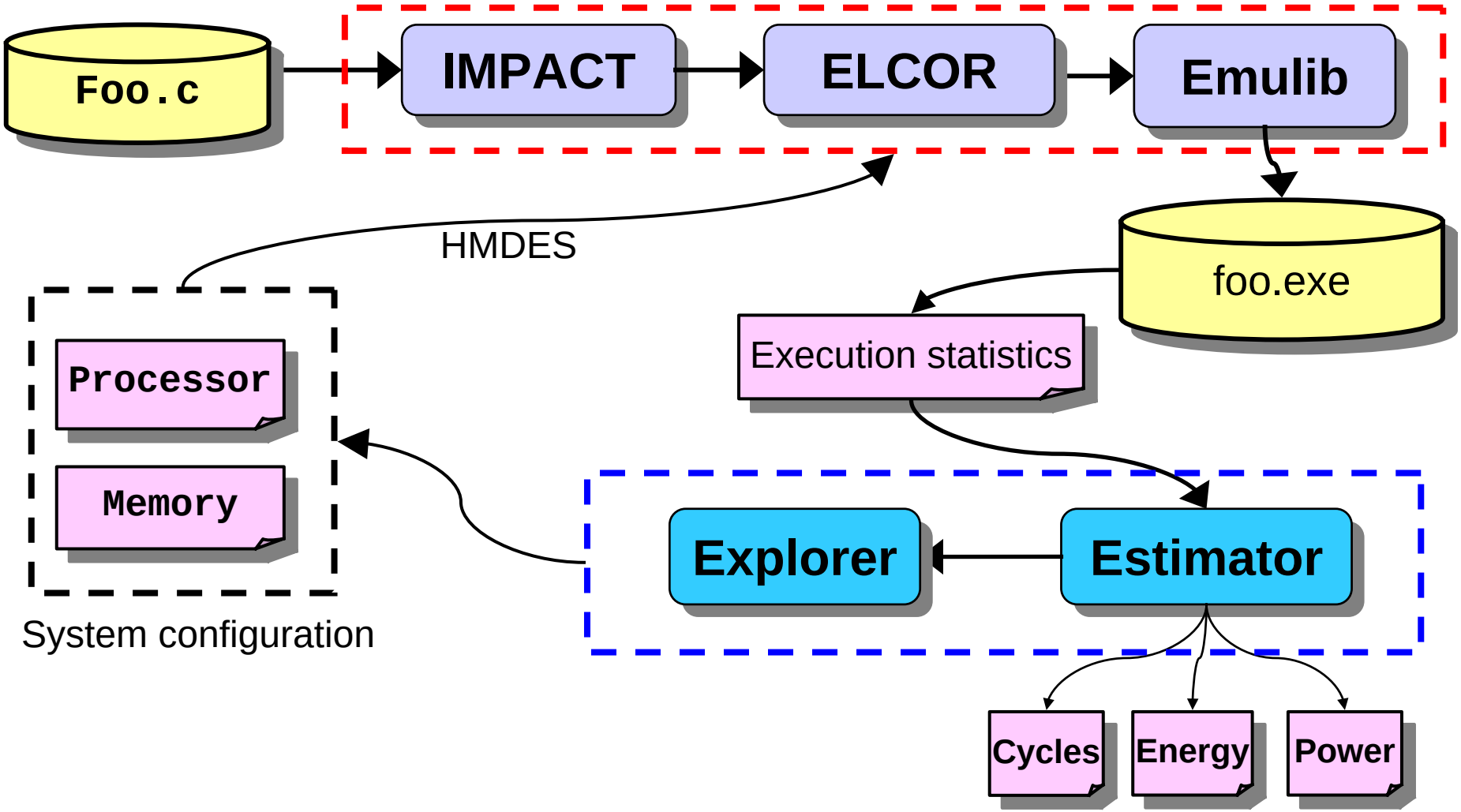


An Open Platform: EPIC Explorer

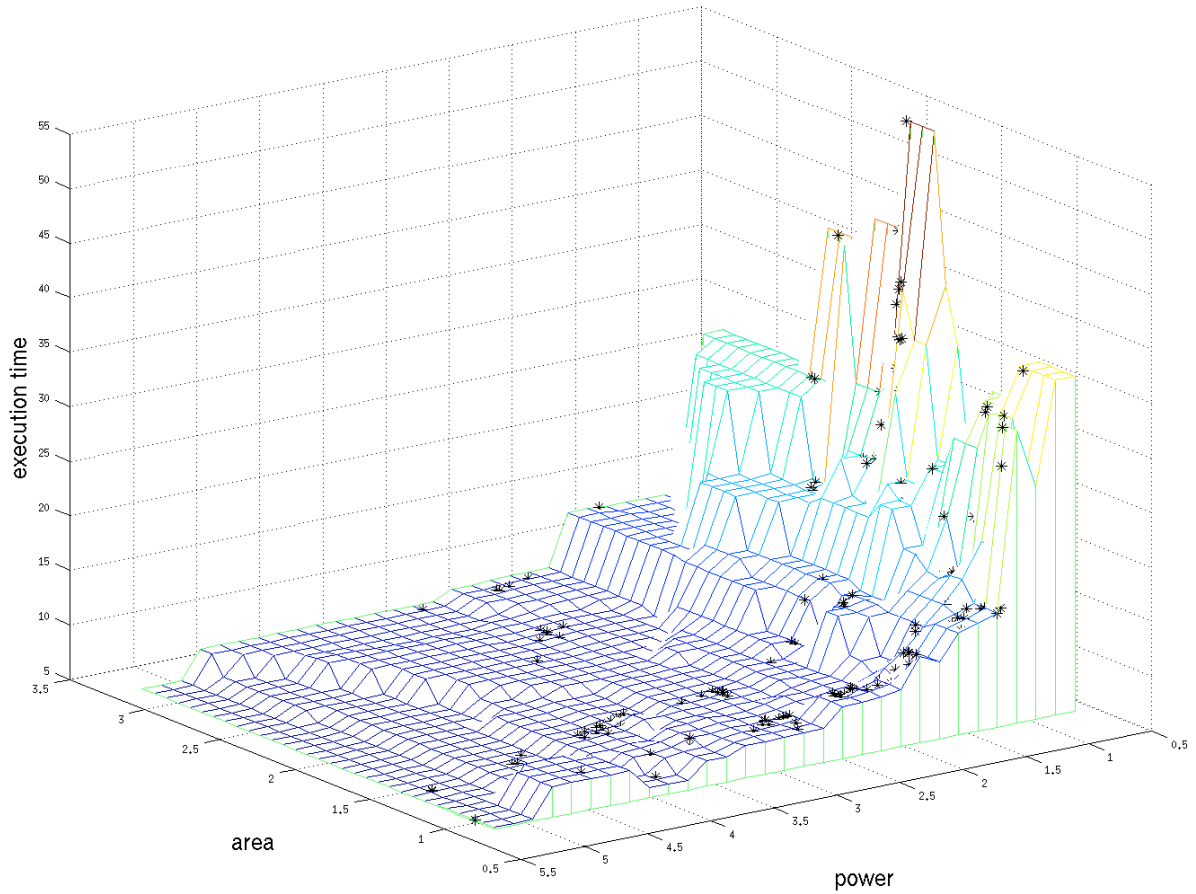
- Interfacing to the **Trimaran** framework that provide VLIW compiler and simulator for dynamic statistics
- Estimator component implementing high level models
- Explorer component implementing multi-objective design space exploration algorithms

- EPIC Explorer
  - ➔http://epic-explorer.sourceforge.net

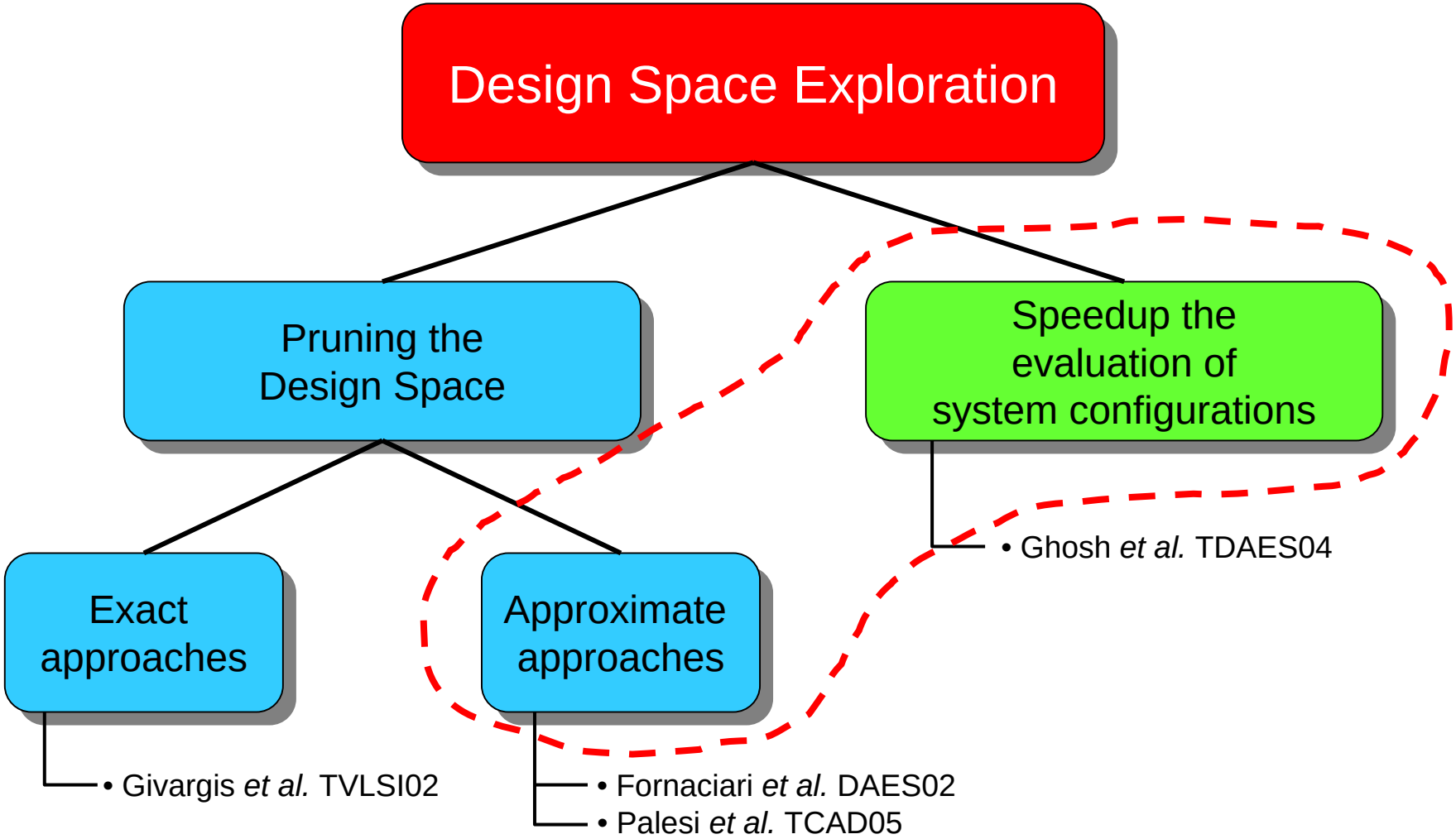
The Exploration Data Flow



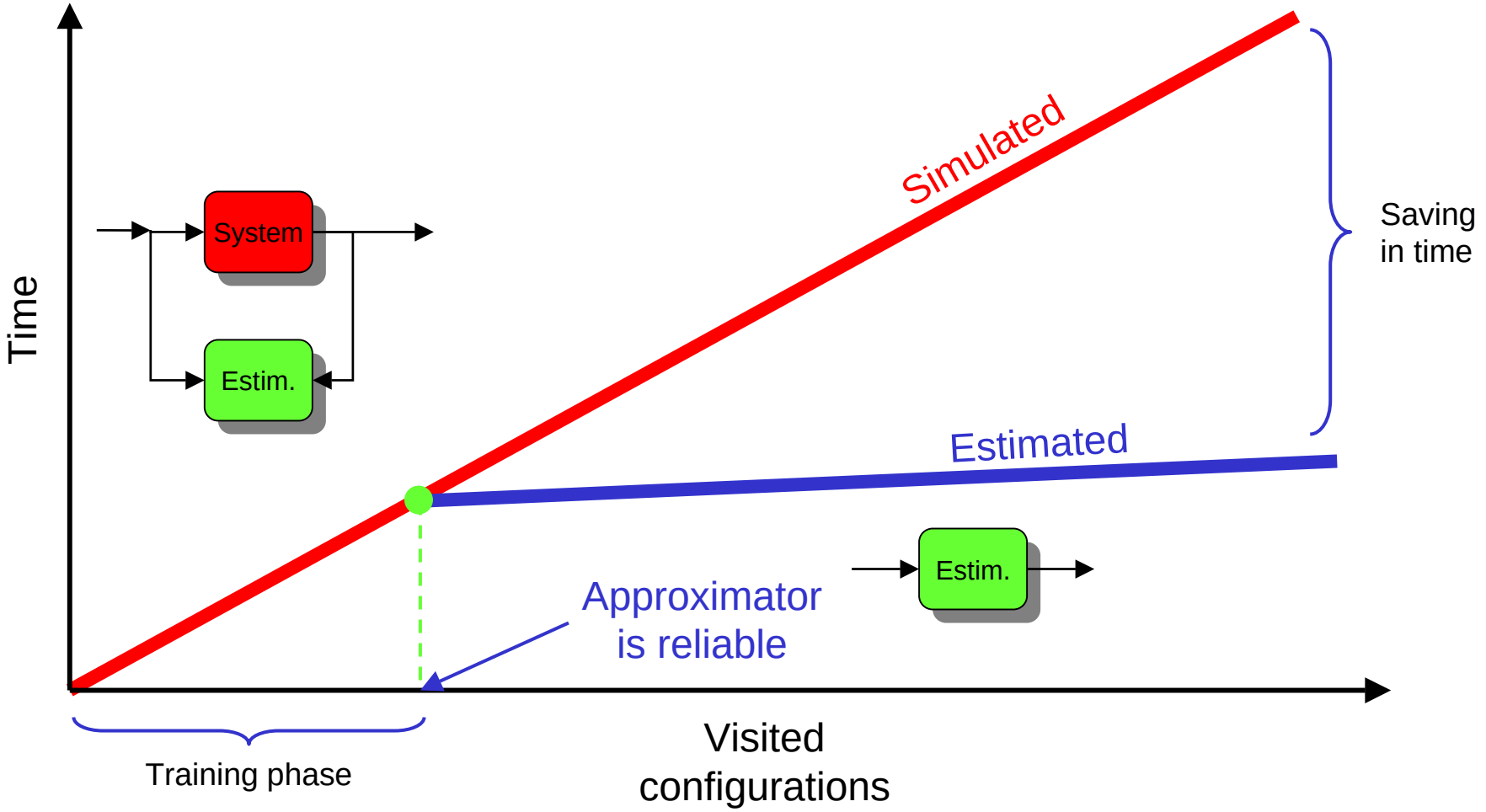
Pareto Surface



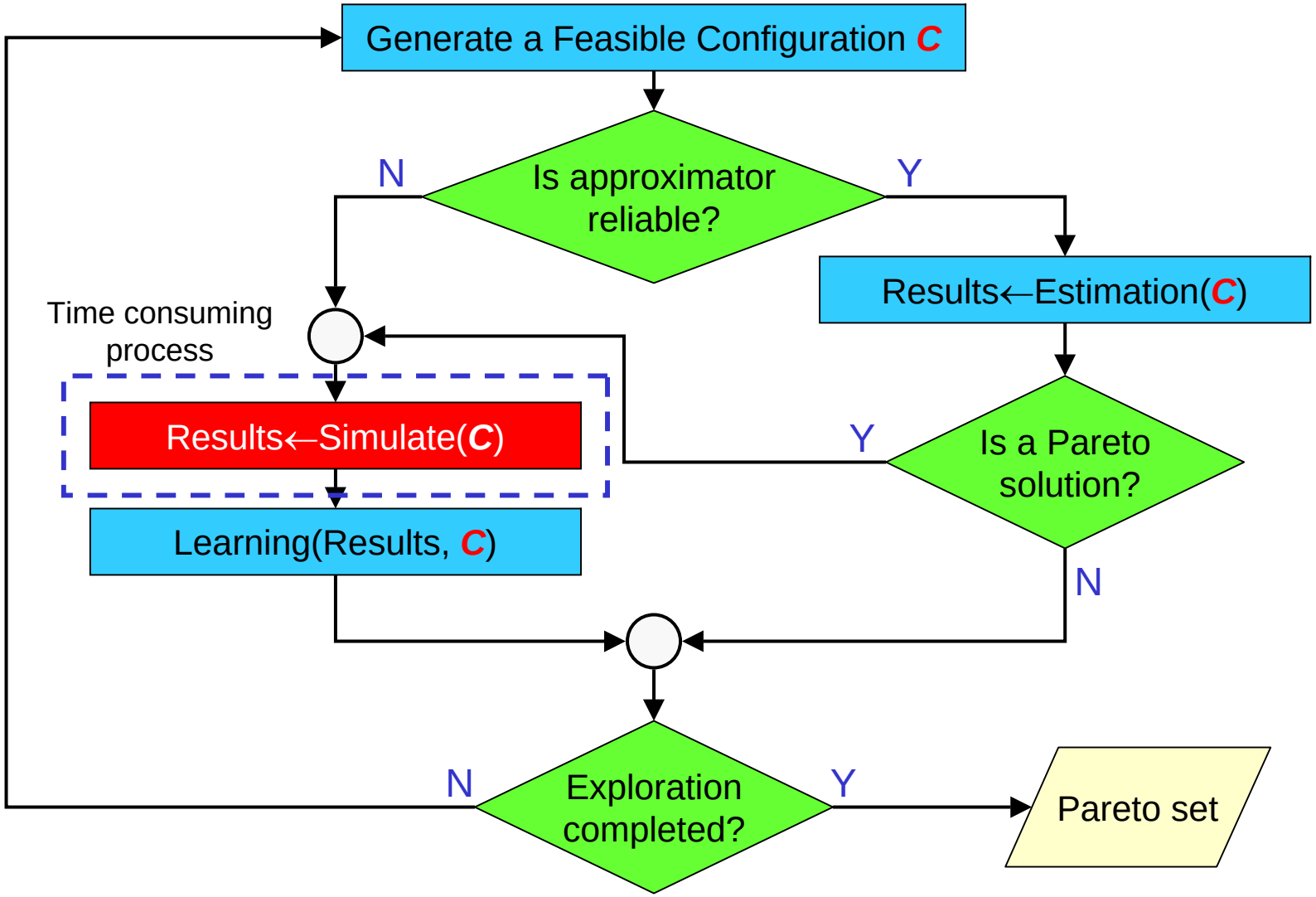
Design Space Exploration



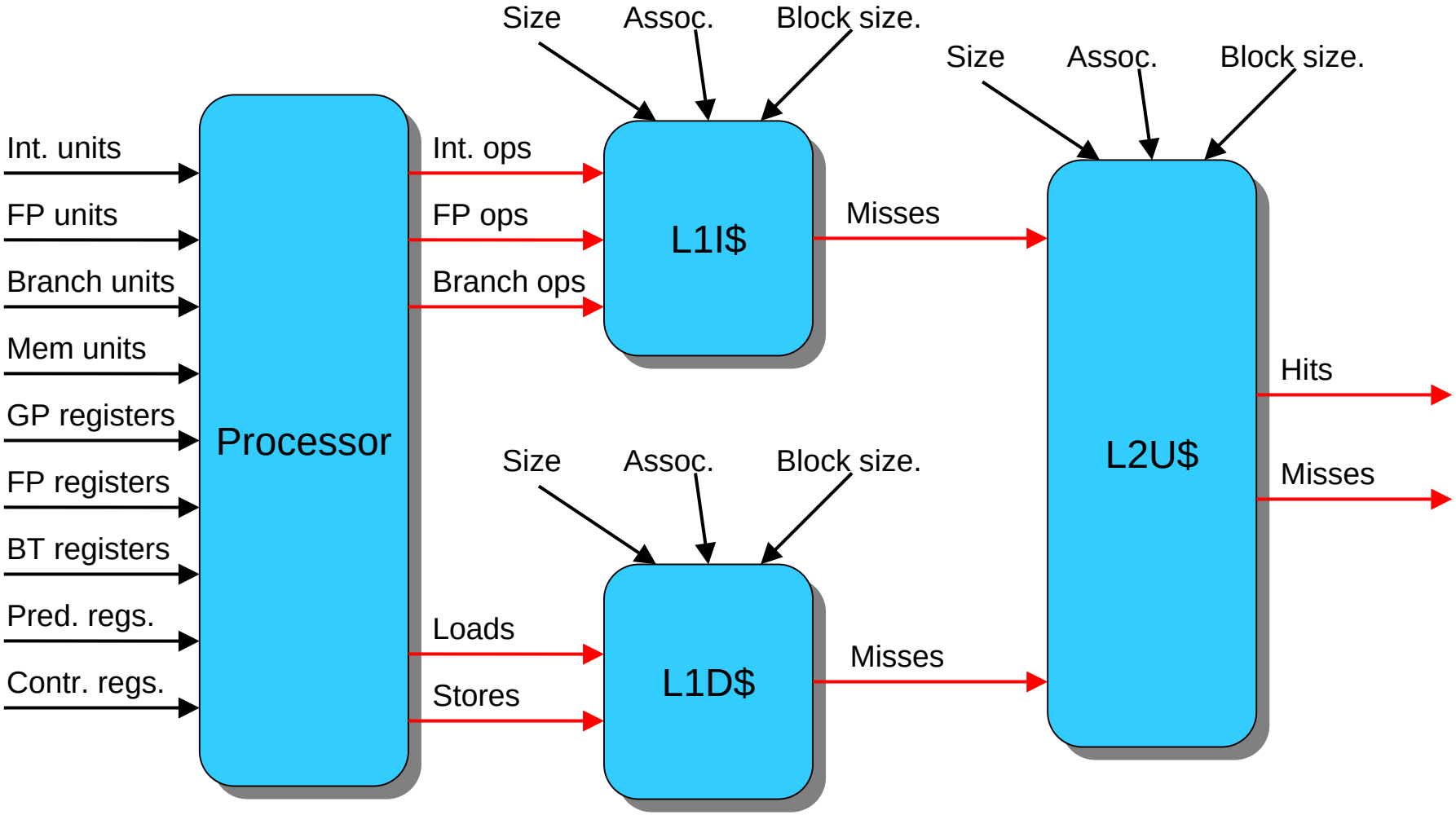
The Basic Idea



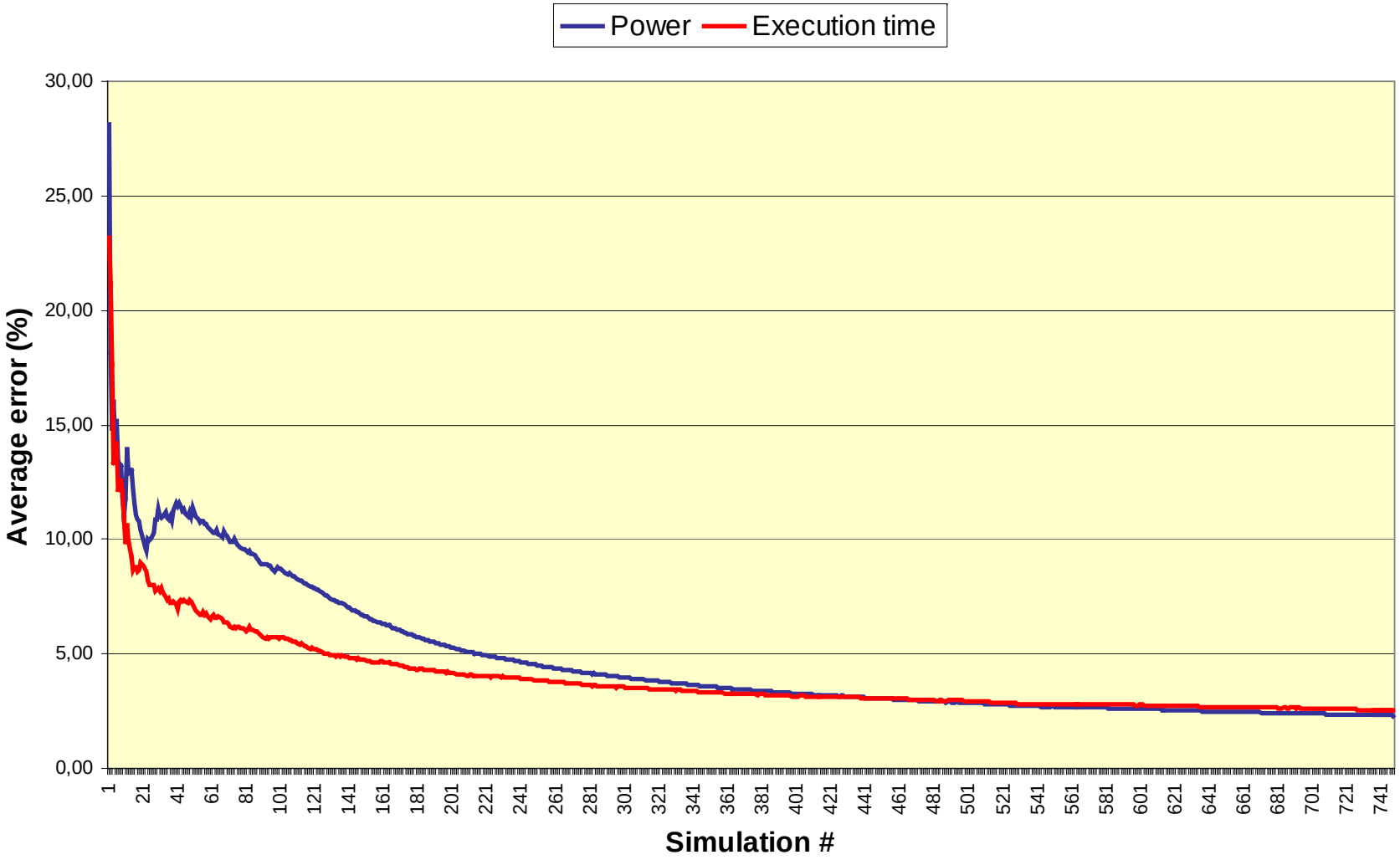
Proposed Exploration Flow



Hierarchical Fuzzy System



Simulation Error

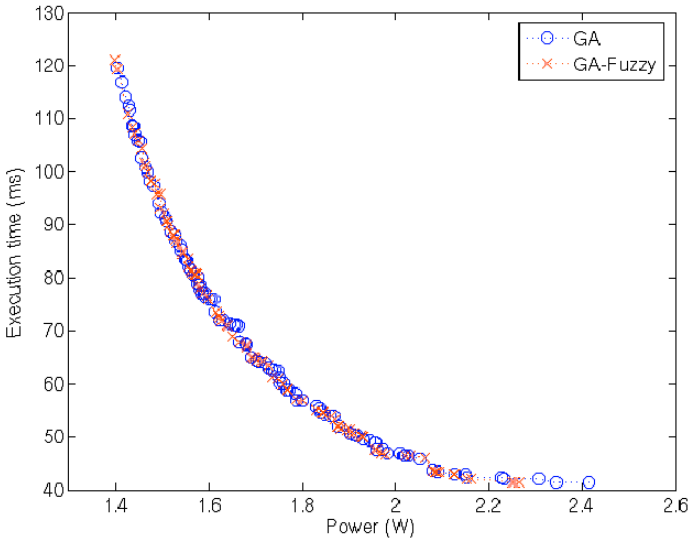


Case Study

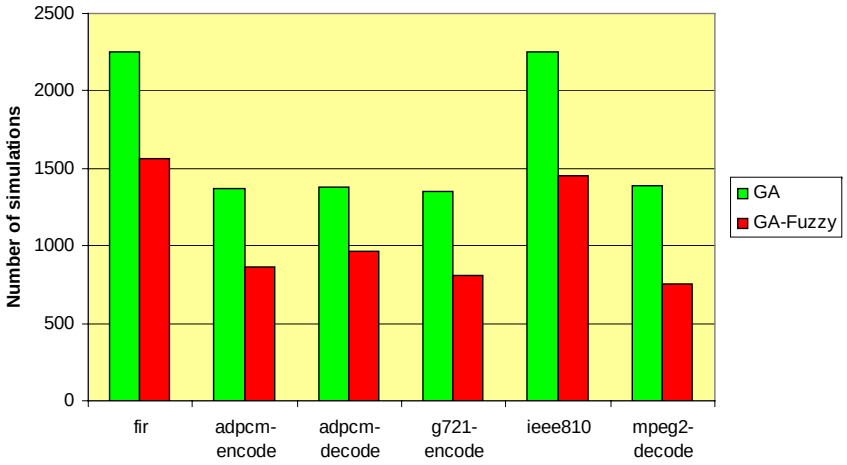
Parameter	Parameter space
GPR/FPR	16, 32, 64, 128
PR/CR	32, 64, 128
BTR	8, 12, 16
Integer/FP units	1, 2, 3, 4, 5, 6
Memory/Branch units	1, 2, 3, 4
L1D/I cache size	1KB, 2KB, …, 128KB
L1D/I cache block size	32B, 64B, 128B
L1D/I cache assoc.	1, 2, 4
L2U cache size	32KB, 64KB, …, 512KB
L2U cache block size	64B, 128B, 256B
L2U cache assoc.	2, 4, 8, 16
Space size	***7.7397x10^10***

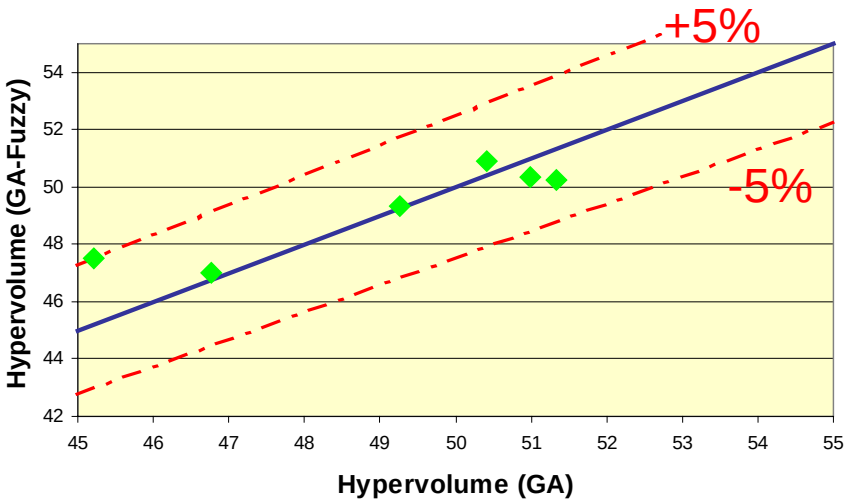
Application	Description	Avg. Simulation time (sec)
fir	Fir filter	9,1
ieee810	IEEE-1180 reference inverse DCT	37,5
adpcm-encode	Adaptive differential pulse code modulation speech encoding	22,6
adpcm-decode	Adaptive differential pulse code modulation speech decoding	20,2
mpeg2-decode	MPEG-2 video bitstream decoding	113,7
g721-encode	CCITT G.721 voice decompression	25,9

GA vs. GA-Fuzzy



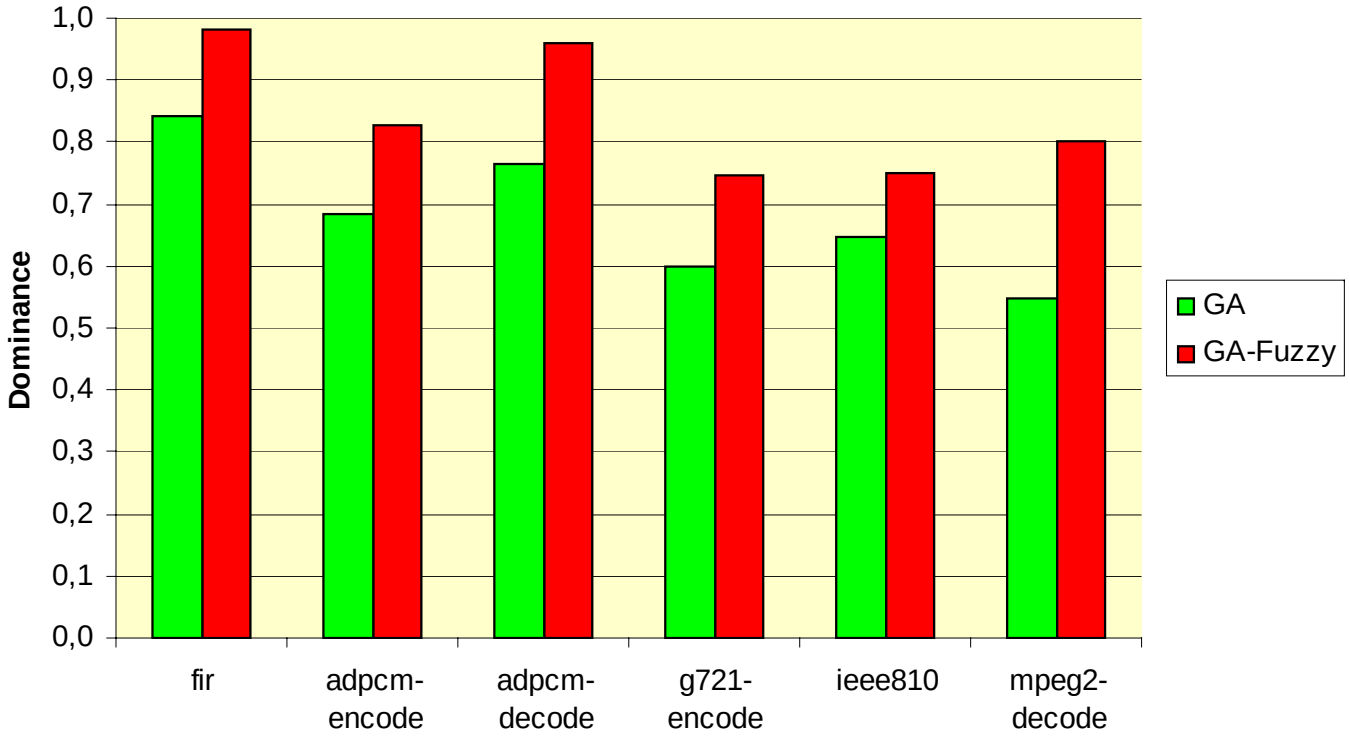
Application	Distance (%)	
	Avg.	Max
fir	0,57	12,21
adpcm-encode	1,52	4,31
adpcm-decode	0,59	4,82
g721-encode	1,10	13,46
ieee810	0,80	4,81
mpeg2-decode	0,88	12,26
Average	***0,91***	***8,65***





GA vs. GA-Fuzzy

- Equal number of simulations
  - ➔ Dominance: Let P_{GA} and P_{GAF} be the Pareto sets obtained by GA and GA-Fuzzy respectively
    - ✓ Dominance(P_{GA}) is the fraction of points of P_{GA} which belongs to Pareto(P_{GA} U P_{GAF})
    - ✓ Dominance(P_{GAF}) is the fraction of points of P_{GAF} which belongs to Pareto(P_{GA} U P_{GAF})

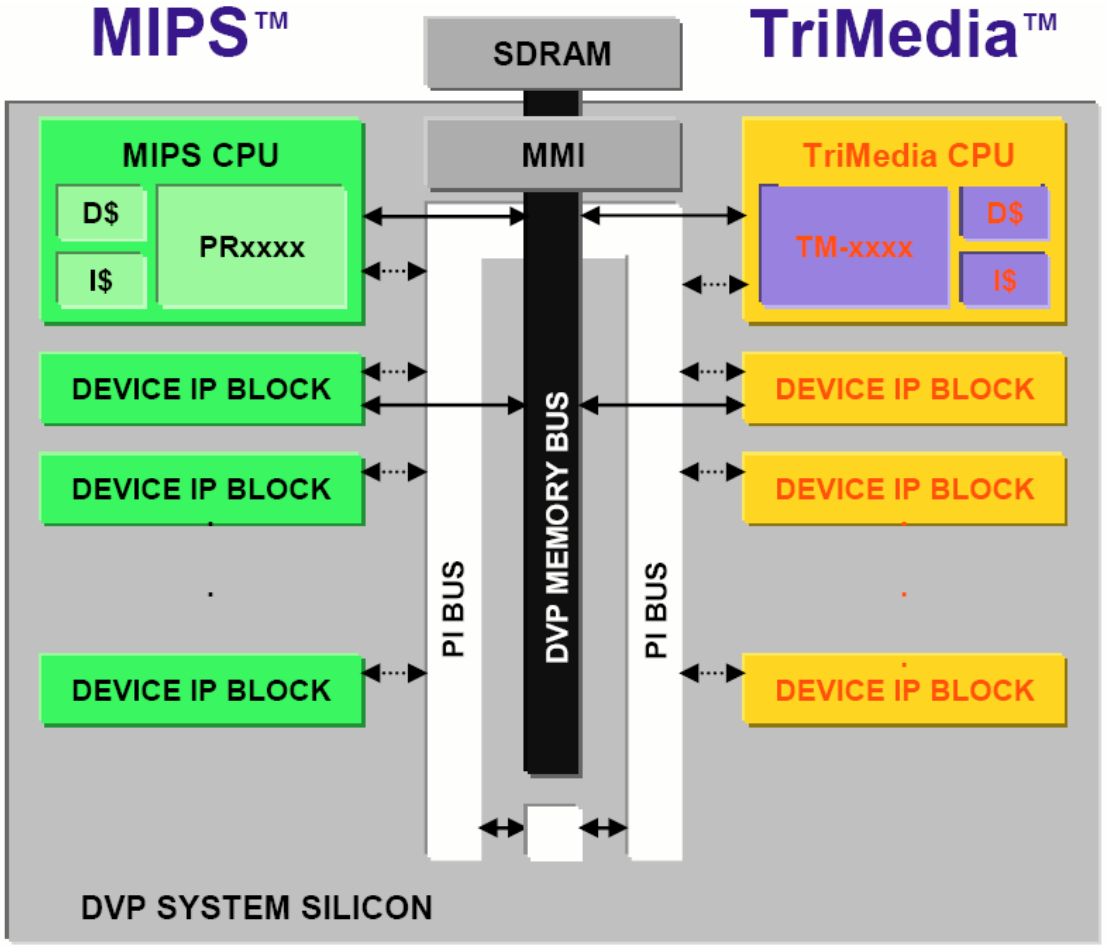


Networks on Chip

The Evolution of SoC Platforms

General-purpose Scalable RISC Processor
- **50 to 300+ MHz**
- **32-bit or 64-bit**

Library of Device IP Blocks
- **Image coprocessors**
- **DSPs**
- **UART**
- **1394**
- **USB**



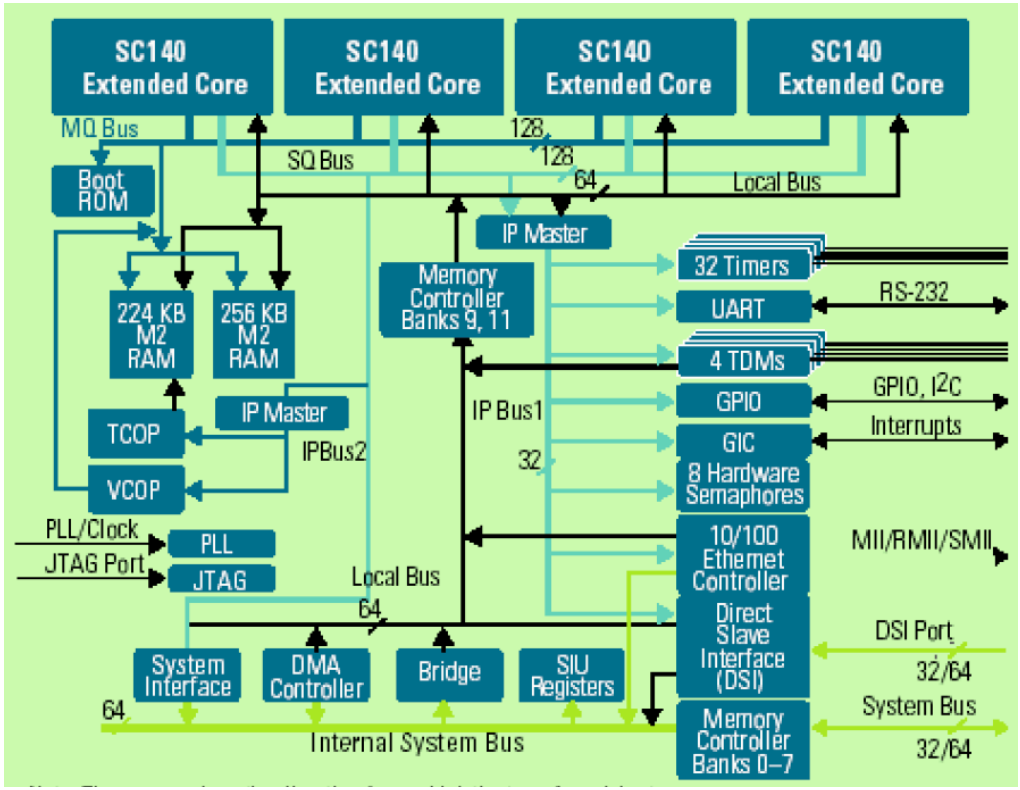
Scalable VLIW Media Processor:
- **100 to 300+ MHz**
- **32-bit or 64-bit**

Nexperia™ System Buses
- **32-128 bit**

2 Cores: Philips' Nexperia PNX8850 SoC platform for High-end digital video (2001)

Running Forward…

- Four 350/400 MHz StarCore SC140 DSP extended cores
- 16 ALUs: 5600/6400 MMACS
- 1436 KB of internal SRAM & multi-level memory hierarchy
- Internal DMA controller supports 16 TDM unidirectional channels,
- Two internal coprocesssors (TCOP and VCOP) to provide special-purpose processing capability in parallel with the core processors

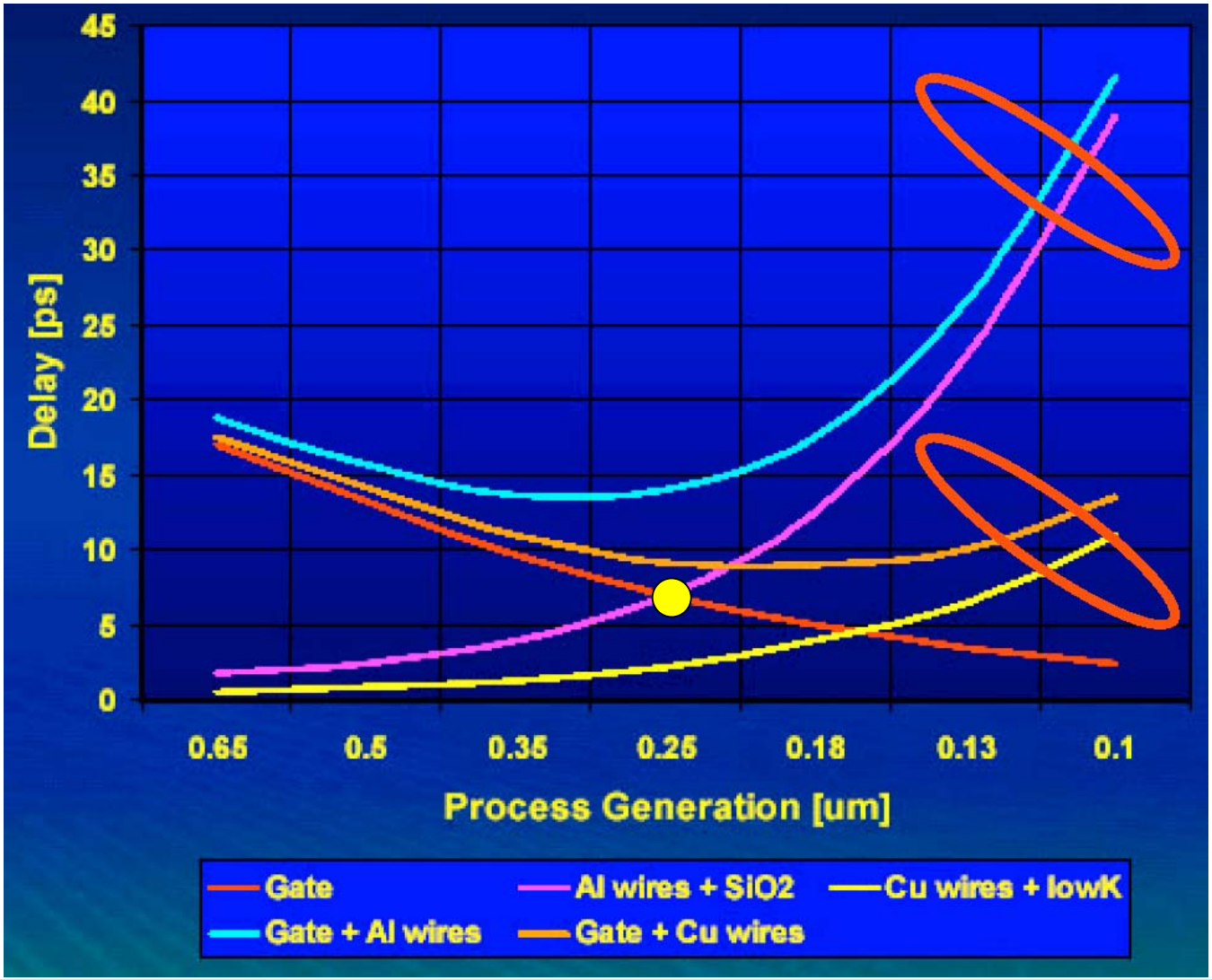


6 Cores: Motorola's MSC8126 SoC platform for 3G base stations (late 2003)

What's Happening in SoCs?

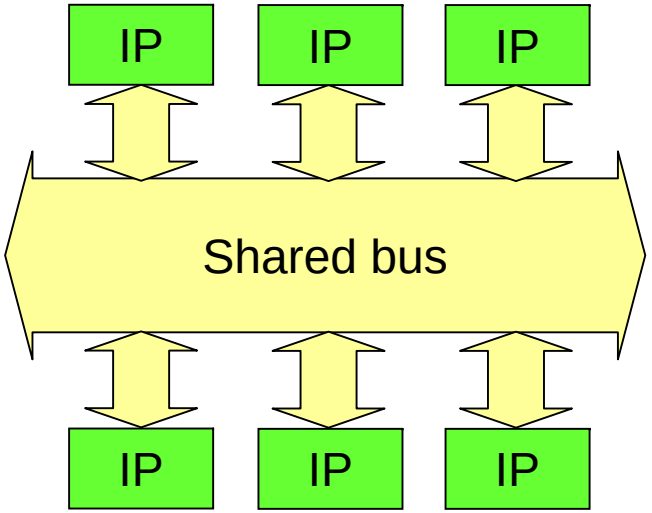
- Technology: no slow-down in sight!
  - ➔ Faster and smaller transistors: 90 → 65 → 45 nm
  - ➔ … but slower wires, lower voltage, more noise!
    - ✓ 80% or more of the delay of critical paths will be due to interconnects
- Design complexity: from 2 to 10 to 100 cores!
  - ➔ Design reuse is essential
  - ➔ …but differentiation/innovation is key for winning on the market!
- Performance and power: GOPS for MWs!
  - ➔ Performance requirements keep going up
  - ➔ …but power budgets don't!

The Deep Submicron Effects

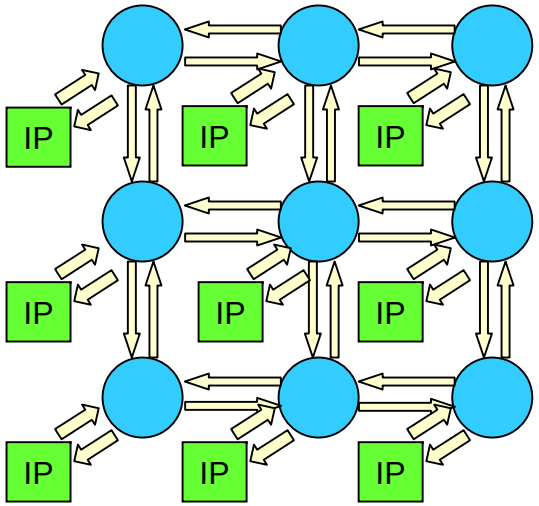


Communication Architectures

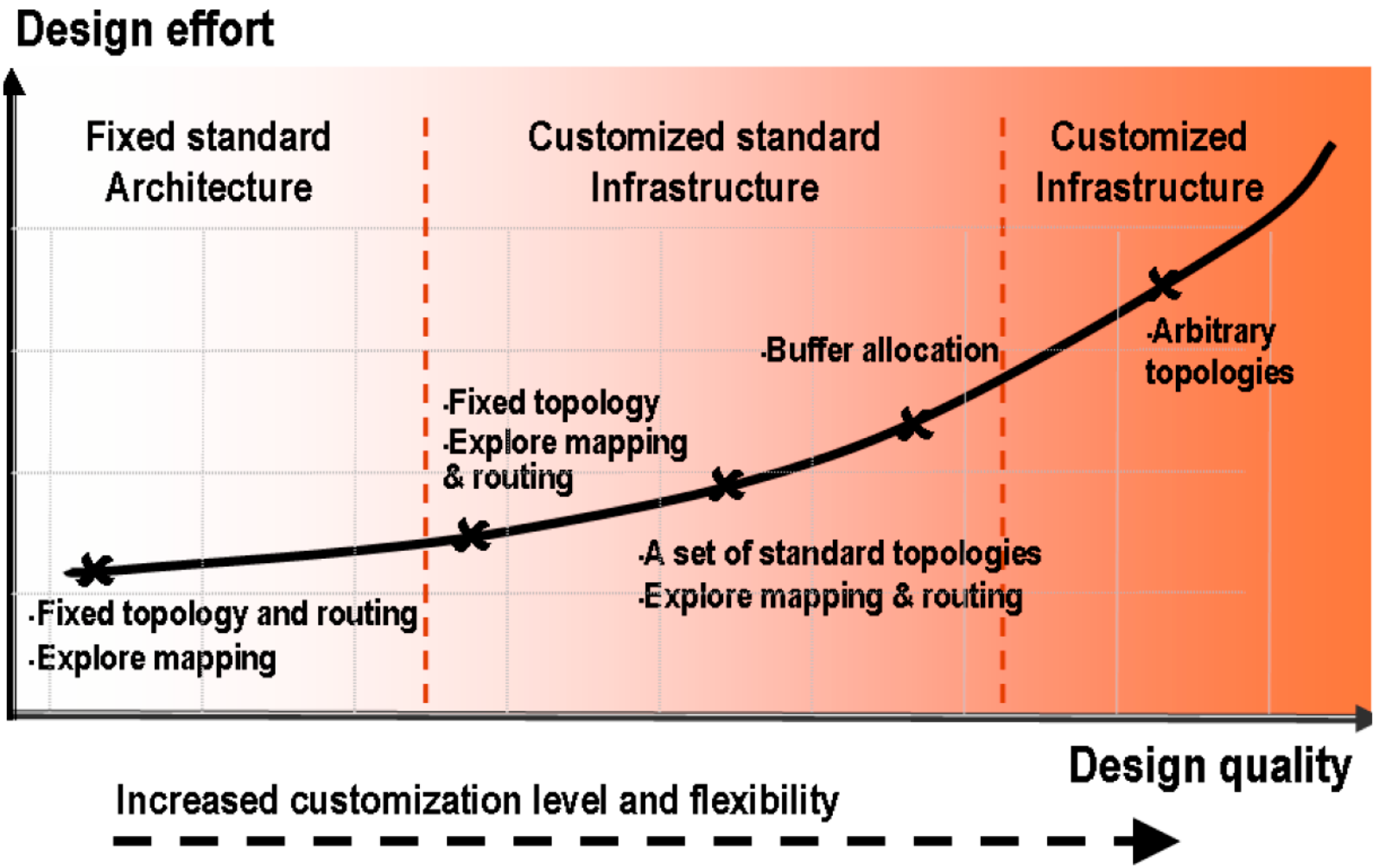
- Shared bus
  - Low area
  - Poor scalability
  - High energy consumption



- Network-on-Chip
  - Scalability and modularity
  - Low energy consumption
  - Increase of design complexity



Design Space Exploration for NoC



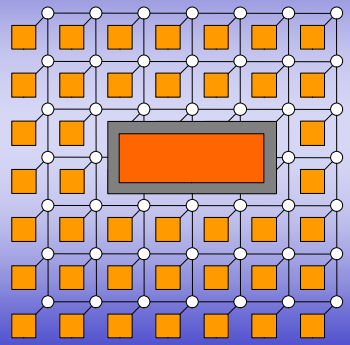
Ogras *et al.*, ASAP'05

Maurizio's Research Topics

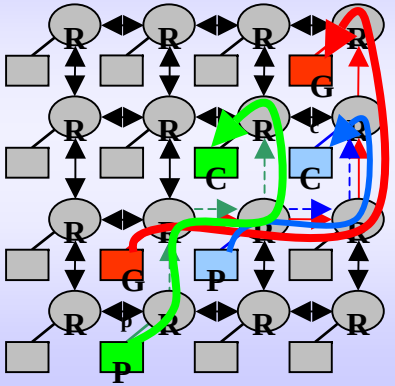
- Topological Mapping
- Routing Algorithms
  - ➔ Application Specific Routing Algorithms
  - ➔ Selection Strategies

NoC Research @ Jönköping Univ.

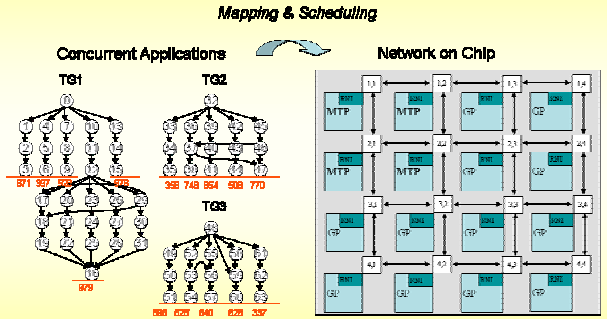
Region Concept and Deadlock free routing in Mesh NoC



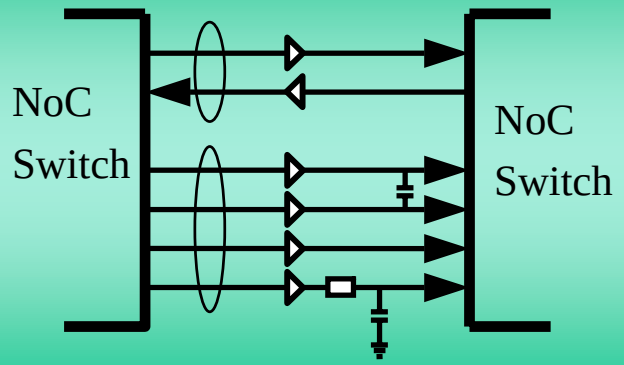
Routing Schemes in NoCs with Mixed QoS Traffic



Mapping Applications to NoC Systems



Delay Testing of NoC Interconnects



NoC Research Group

NoC Group @ Jonkoping (1 of 4)

- Region Concept and Deadlock free routing in Mesh NoC
  - ➔ Concept to handle cores larger than tile size like shared memory
  - ➔ Applications of Region Concept
    - ✓ Cores of different sizes
    - ✓ Reuse of application specific multiprocessor SoCs as resource
    - ✓ Sub-network with special properties with NoC
  - ➔ Issues
    - ✓ Deadlock free routing algorithm
    - ✓ Multiple access points
    - ✓ Multiple regions
    - ✓ Placement of regions

NoC Group @ Jonkoping (2 of 4)

- Routing Schemes in NoCs with Mixed QoS Traffic
  - ➔ Mixed QoS traffic = GT + BE
  - ➔ Concurrent applications running on NoC lead to mixed QoS traffic
  - ➔ Variation in GT traffic leads to underutilization of network resources
  - ➔ We model underutilization as SLACK
  - ➔ Slack can be used to improve performance of BE traffic
    - ✓ Improves latency, throughput and jitter of BE traffic
    - ✓ Reduce BE packet drop

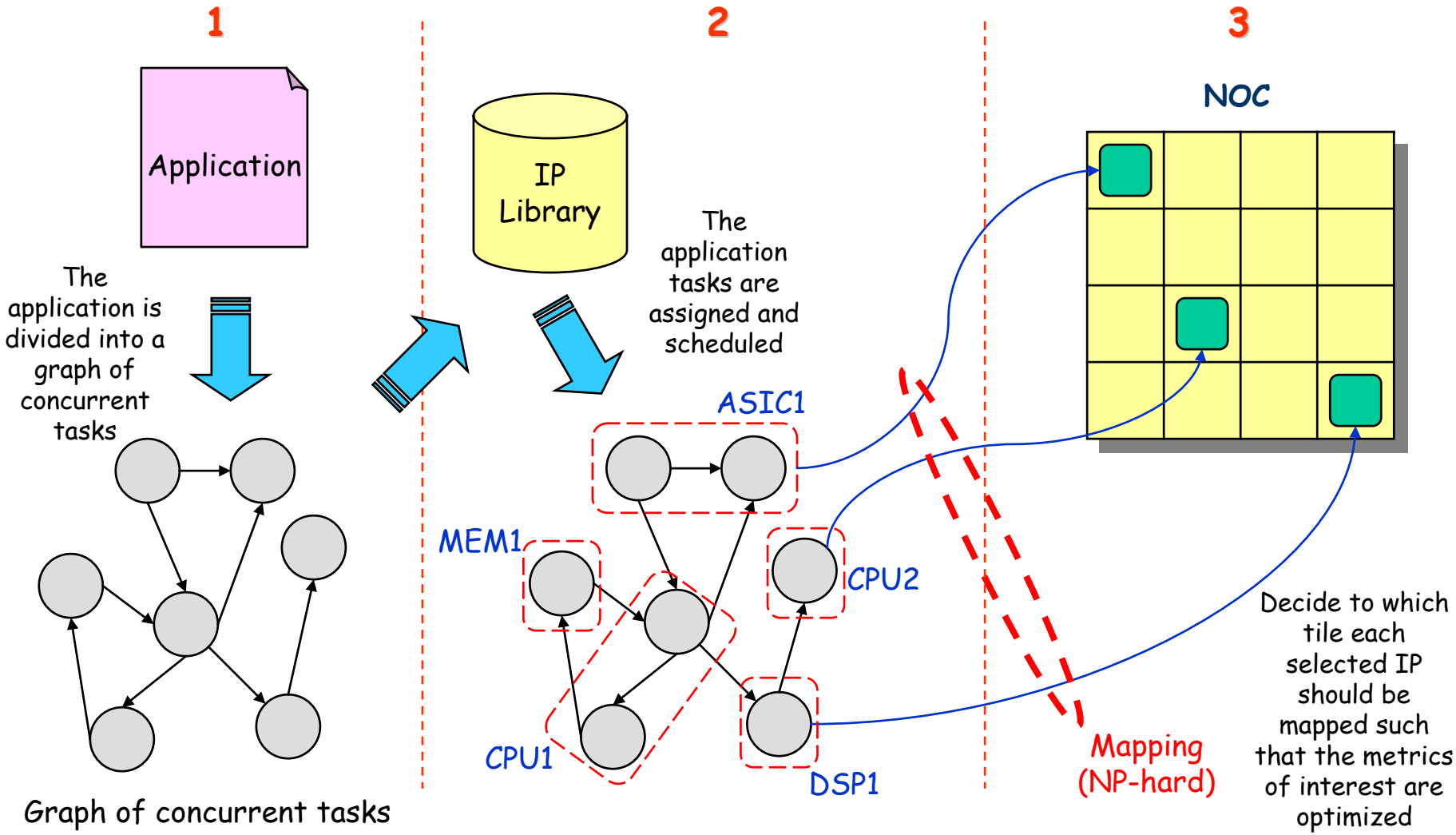
NoC Group @ Jonkoping (4 of 4)

- Mapping Applications to NoC Systems
  - ➔ Special emphasis is on using Multi-threaded processors (MTPs) as NoC resources
  - ➔ Reduces communication cost by running communicating tasks as threads on MTP
    - ✓ Assumption: Area of MTP with 4 threads much less than four processors
  - ➔ MTP can hide memory and I/O latencies

NoC Group @ Jonkoping (4 of 4)

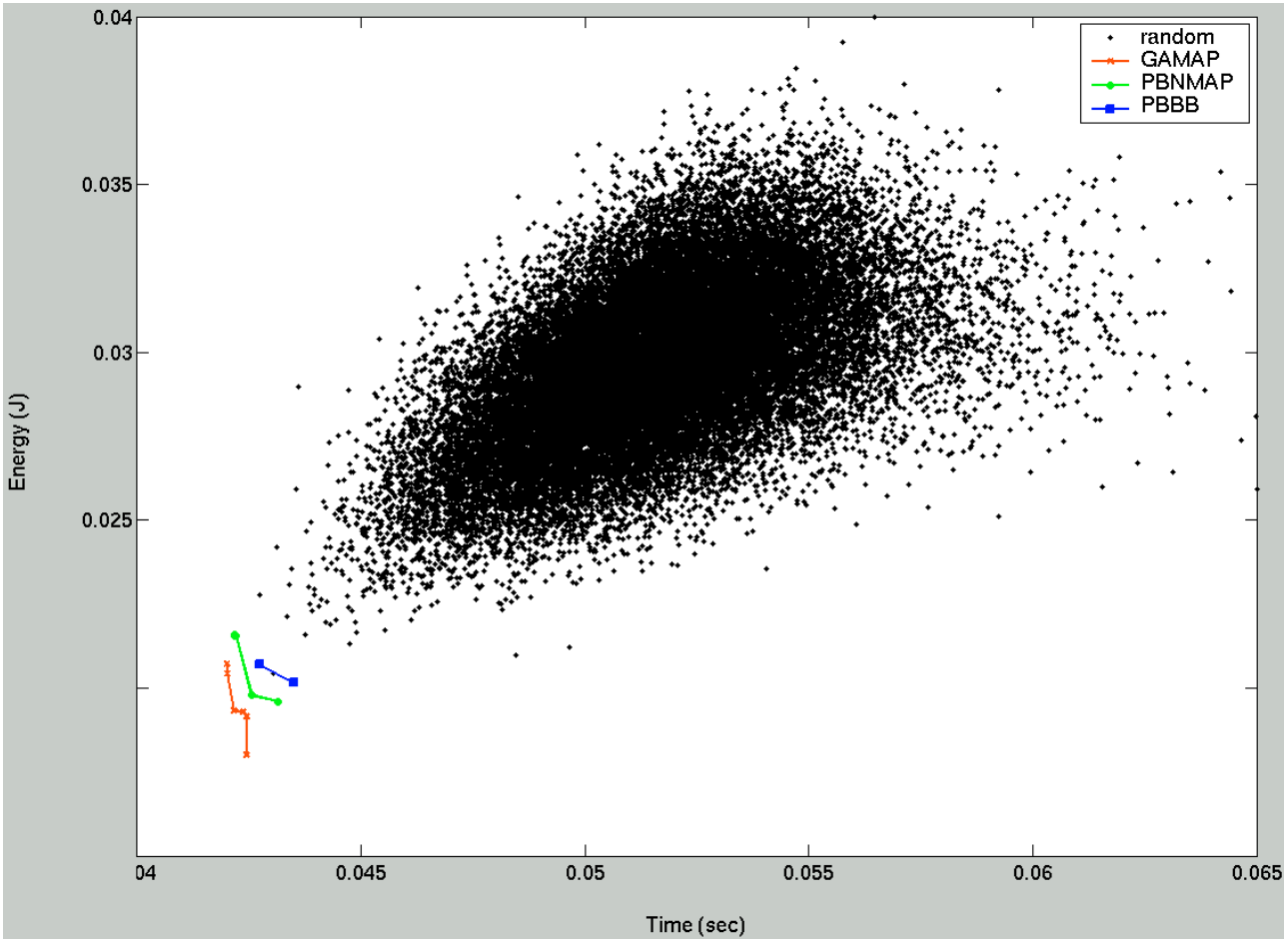
- Testing of NoC Interconnects
  - ➔ Focus on crosstalk induced delay faults and Glitch faults
  - ➔ Important for on-chip buses connecting GALS domains
  - ➔ Contributions
    - ✓ Efficient method to test delay faults in links connecting two NoC switches
    - ✓ Method to test delay faults in links connecting two NoC switches
    - ✓ BIST hardware for delay and glitch testing

The Mapping Problem



Experiments

Real Traffic (MPEG-2)



High performance mapping

	0	1	2	3
0	ASIC5	ASIC4	DSP2	CPU2
1	CPU1	ASIC3	MEM1	ASIC1
2	DSP1	ASIC2	IP3	IP1
3	MEM2	DSP3	IP2	IP4

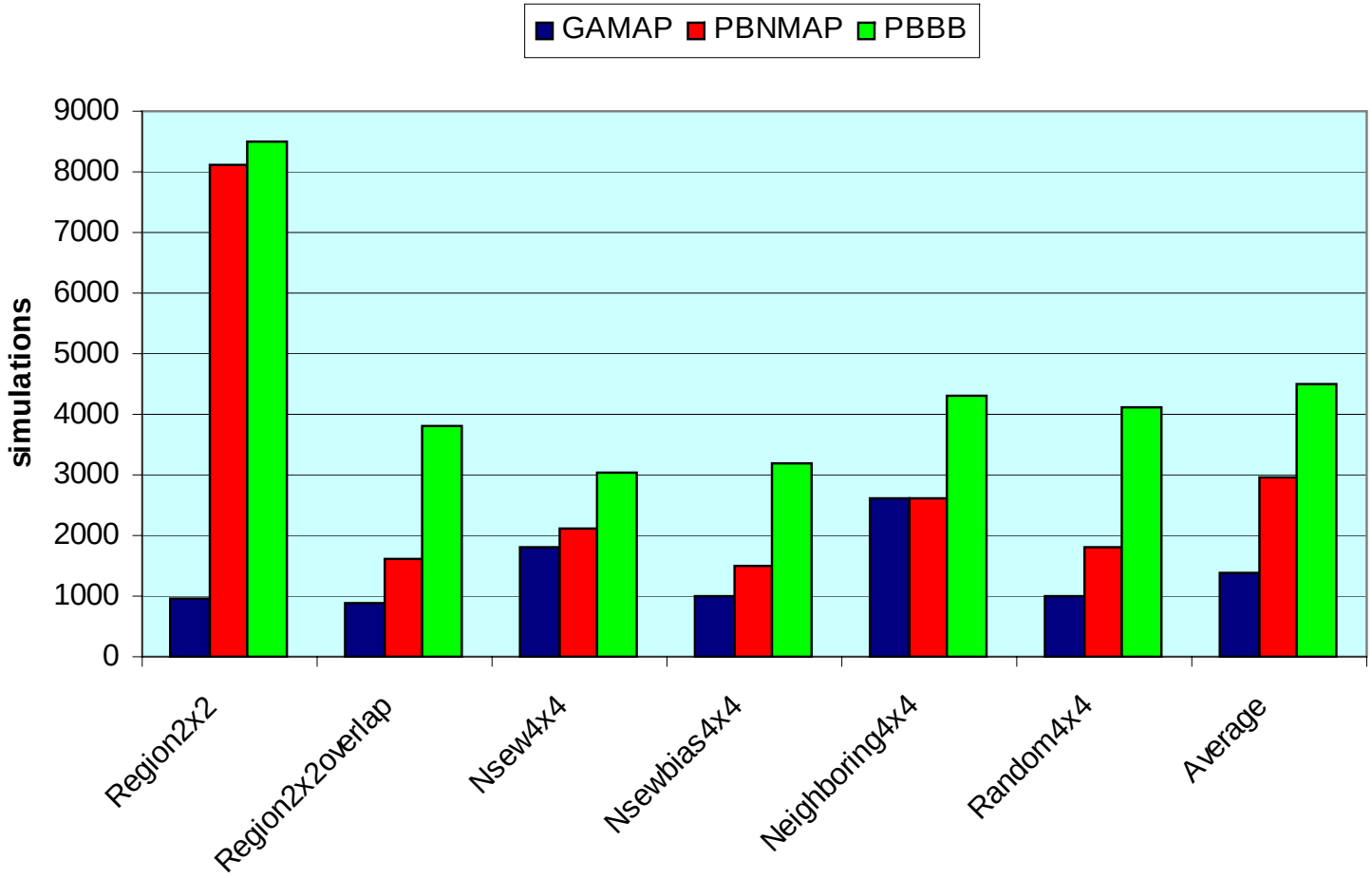
Execution time 41.0 ms
Energy: 20.8 mJ

Low energy mapping

	0	1	2	3
0	ASIC5	ASIC3	ASIC1	DSP2
1	DSP1	ASIC4	MEM1	CPU2
2	CPU1	ASIC2	IP3	IP1
3	MEM2	DSP3	IP2	IP4

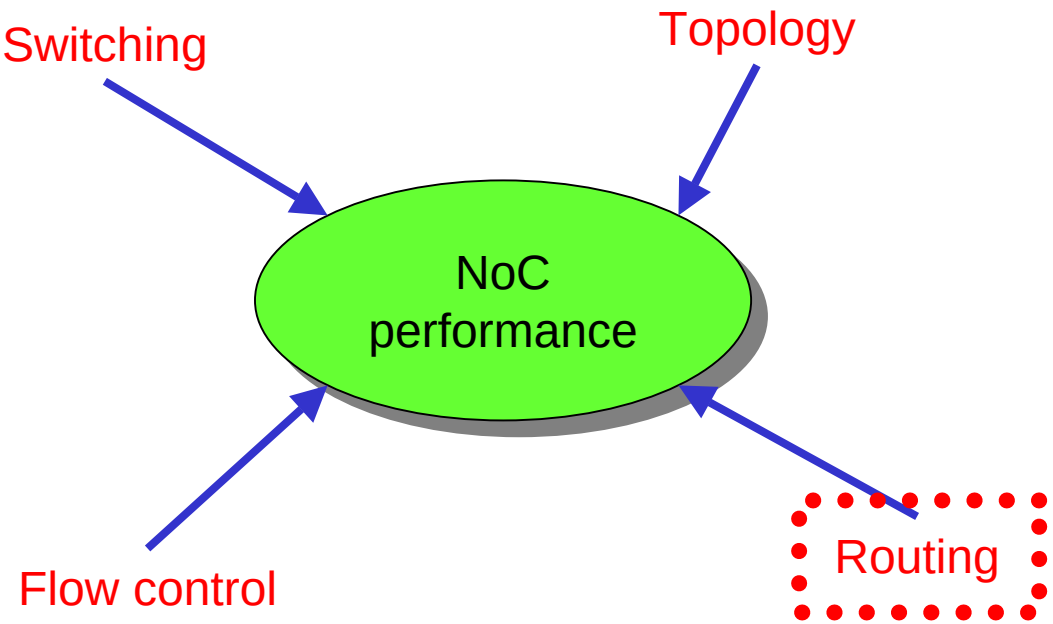
Execution time 42.4 ms
Energy: 17.8 mJ

Experiments
Efficiency



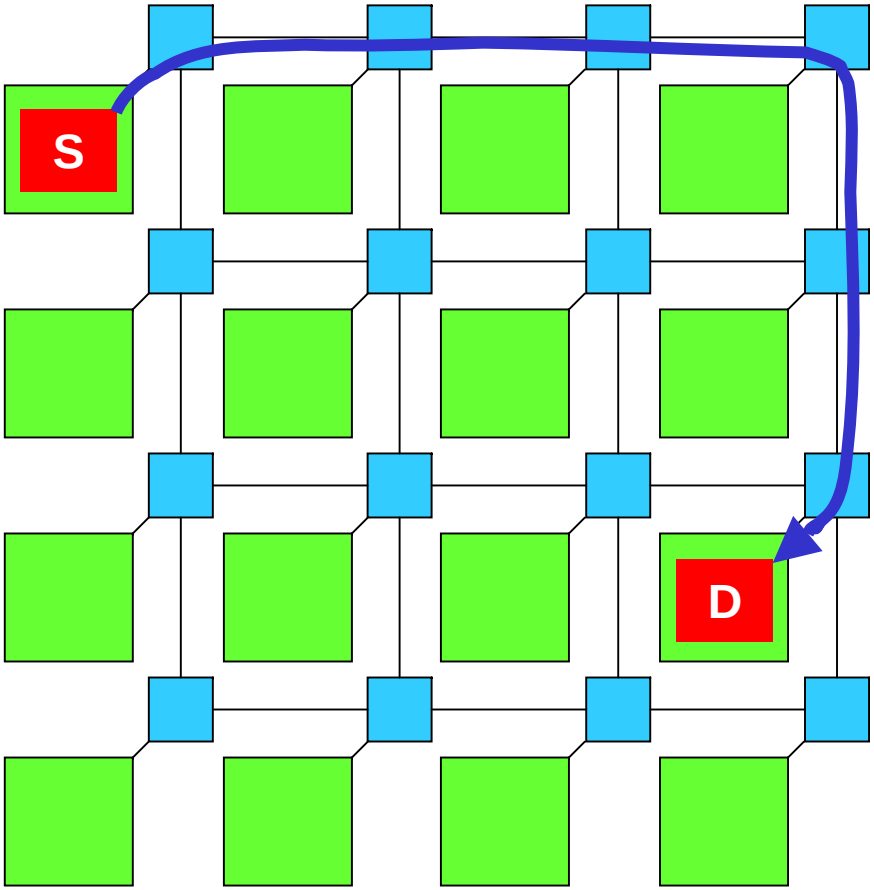
Routing Algoritms

NoC and Routing



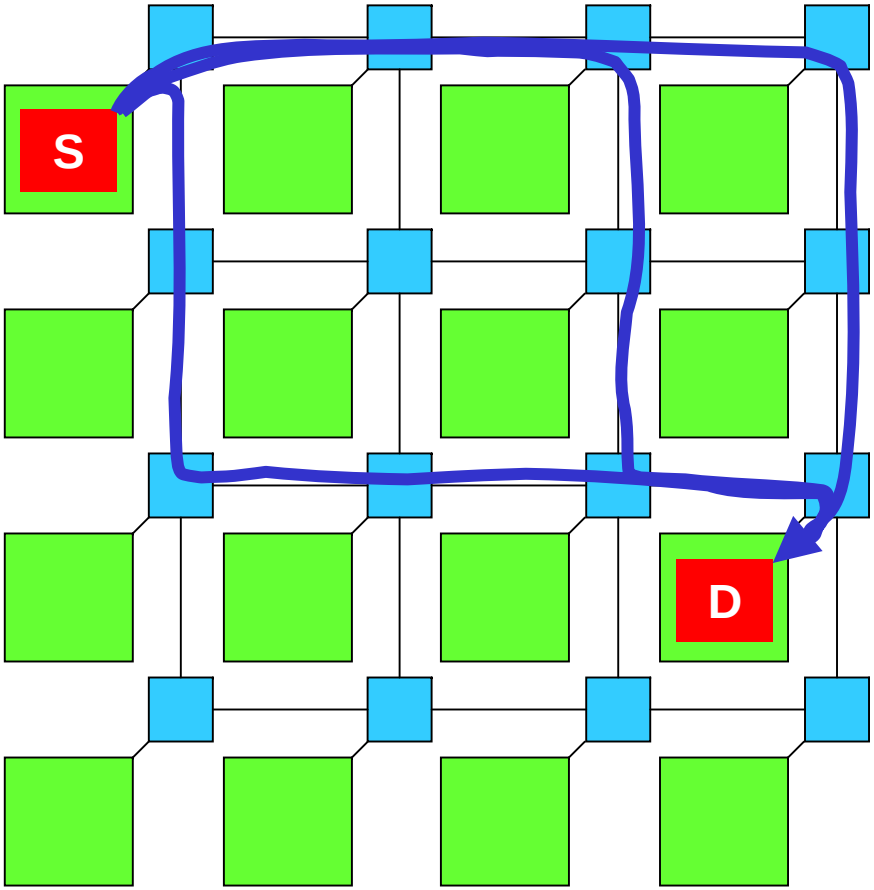
- Routing determines the path selected by a packet to reach its destination
  - Deterministic
  - Adaptive

Deterministic Routing



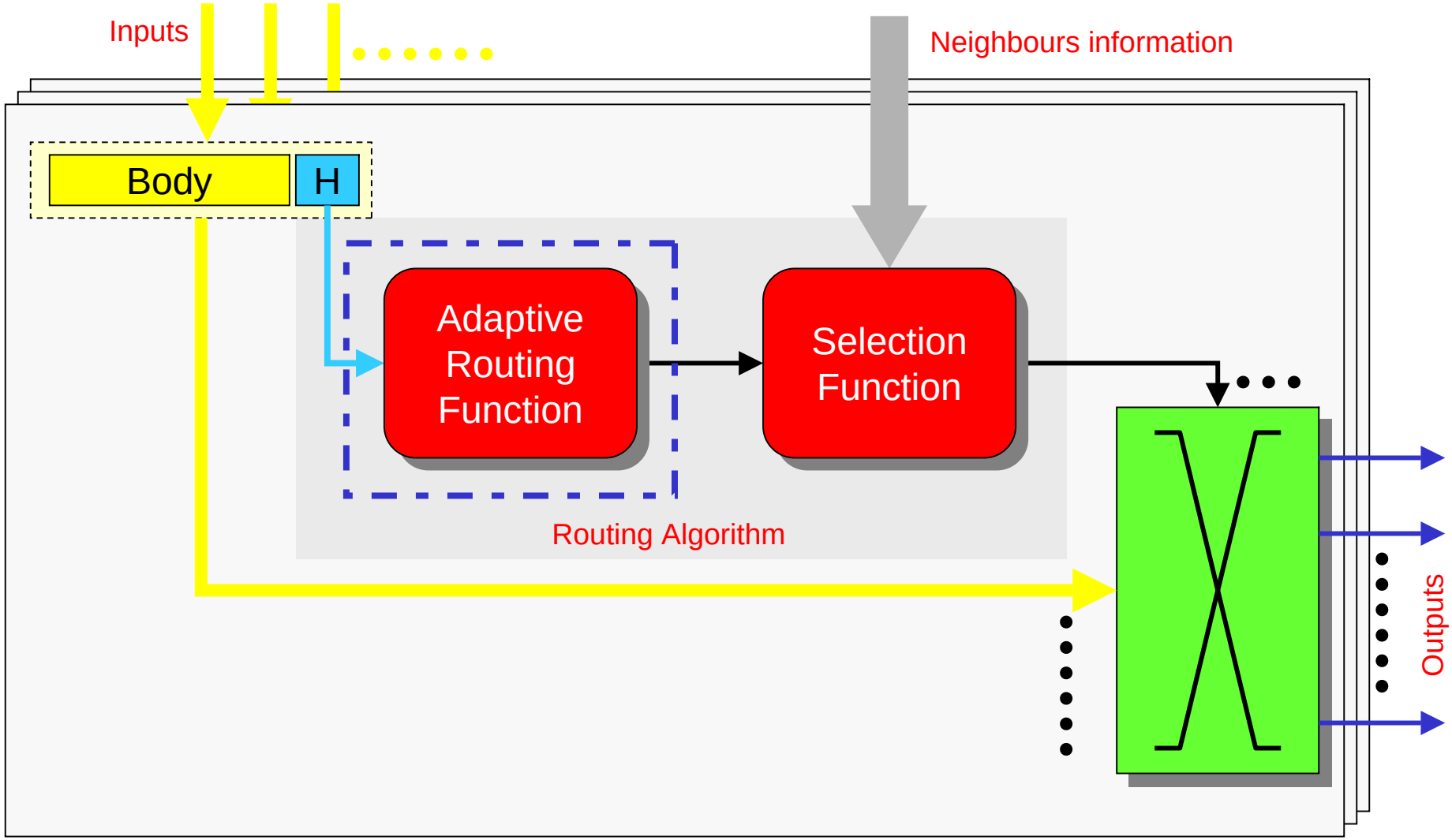
- *Deterministic algorithms* always choose the same path between two nodes
  - ➔ Easy to implement and to make deadlock free
  - ➔ In-order arrival of packets
  - ➔ Do not use path diversity and thus bad on load balancing
  - ➔ Inefficient use of network resources

Adaptive Routing

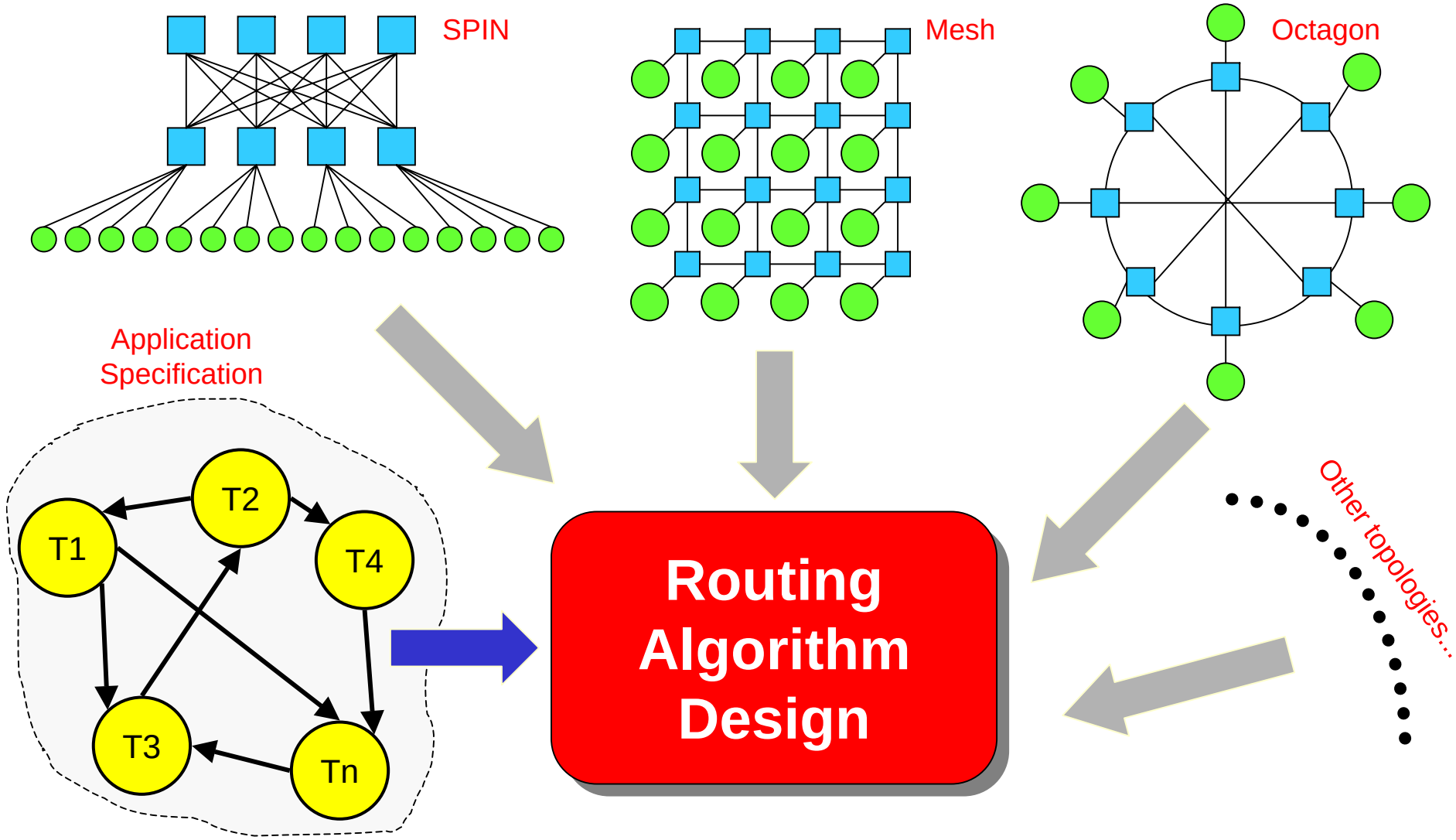


- *Adaptive algorithms* multiple paths from the source to the destination are possible
  - ➔ Length of queues
  - ➔ Historical channel load
  - ➔ Increase the chance that packets may avoid hot spots or faulty regions

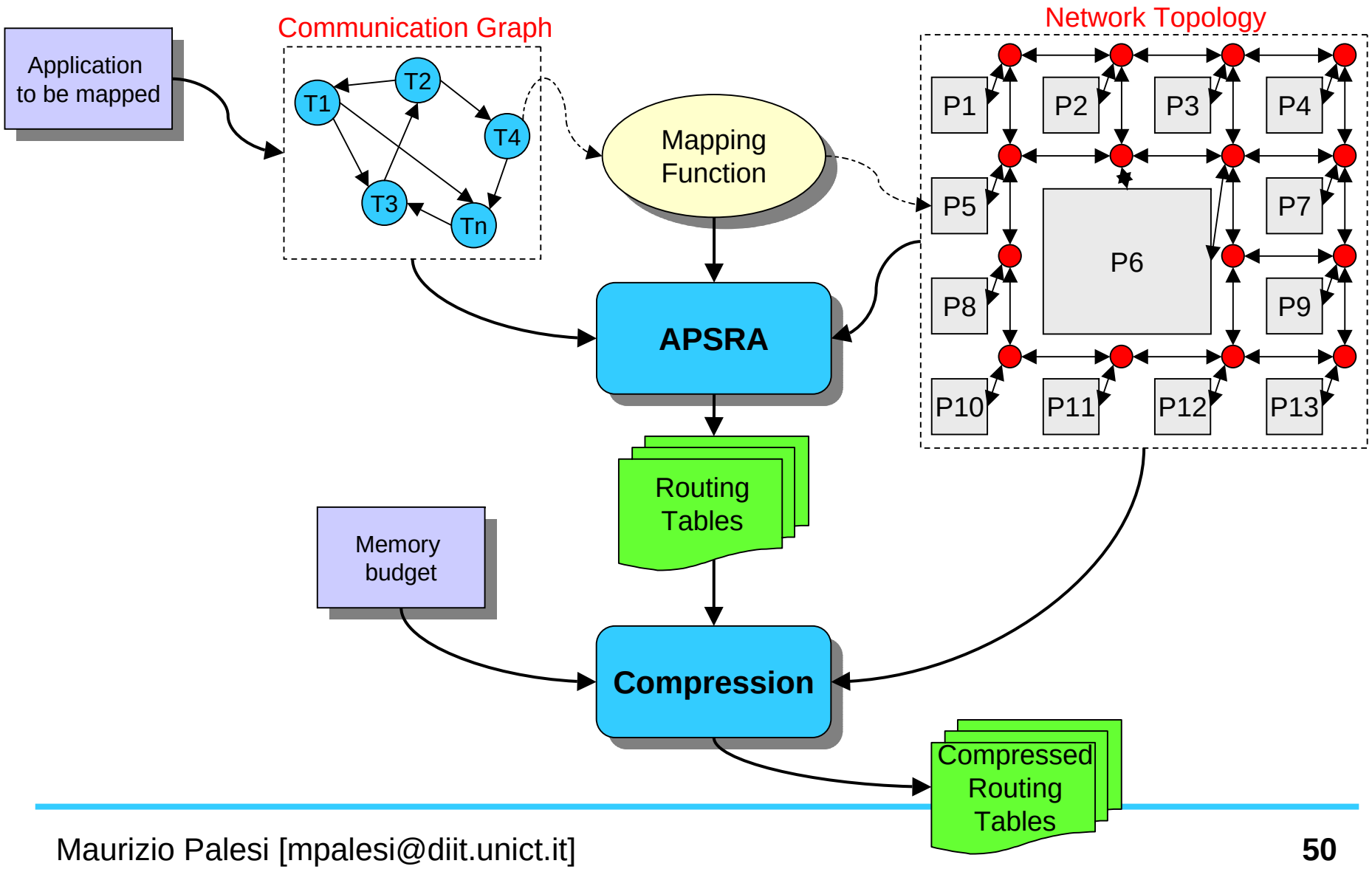
Routing and Selection



Routing Algorithm Design

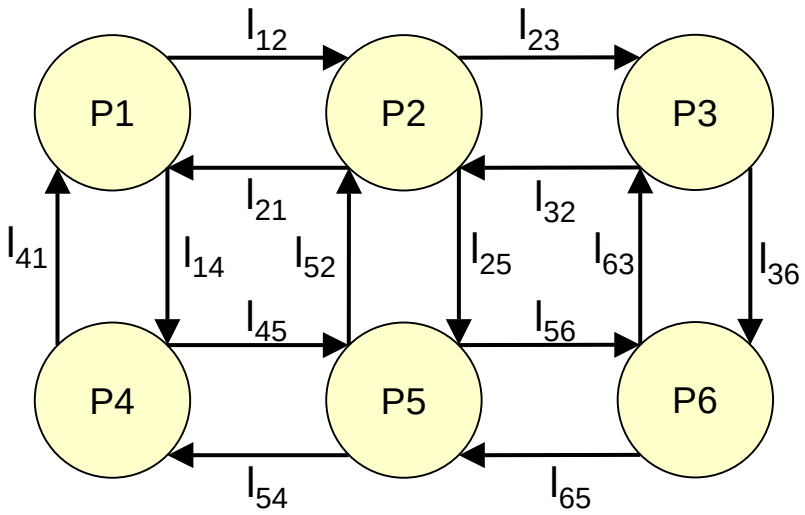


APSRA Design Methodology

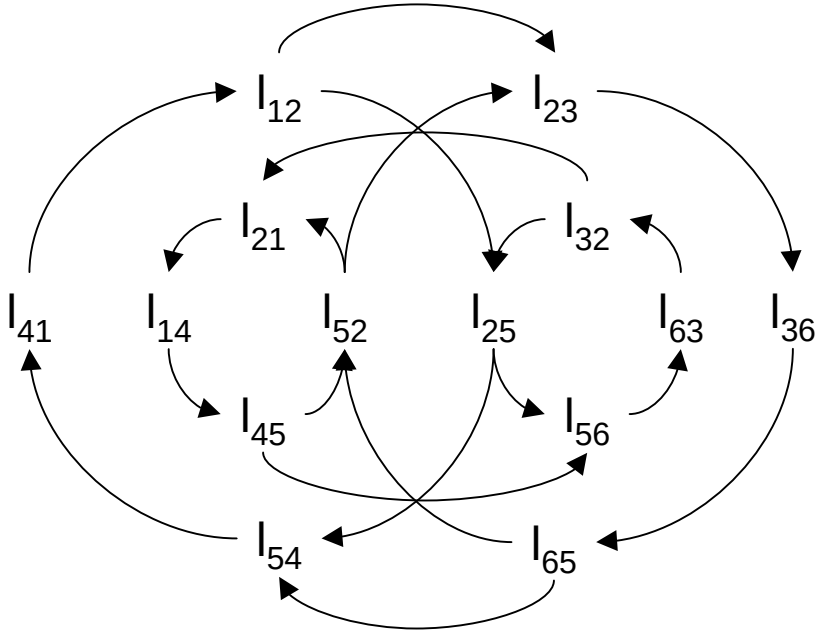


APSRA Example

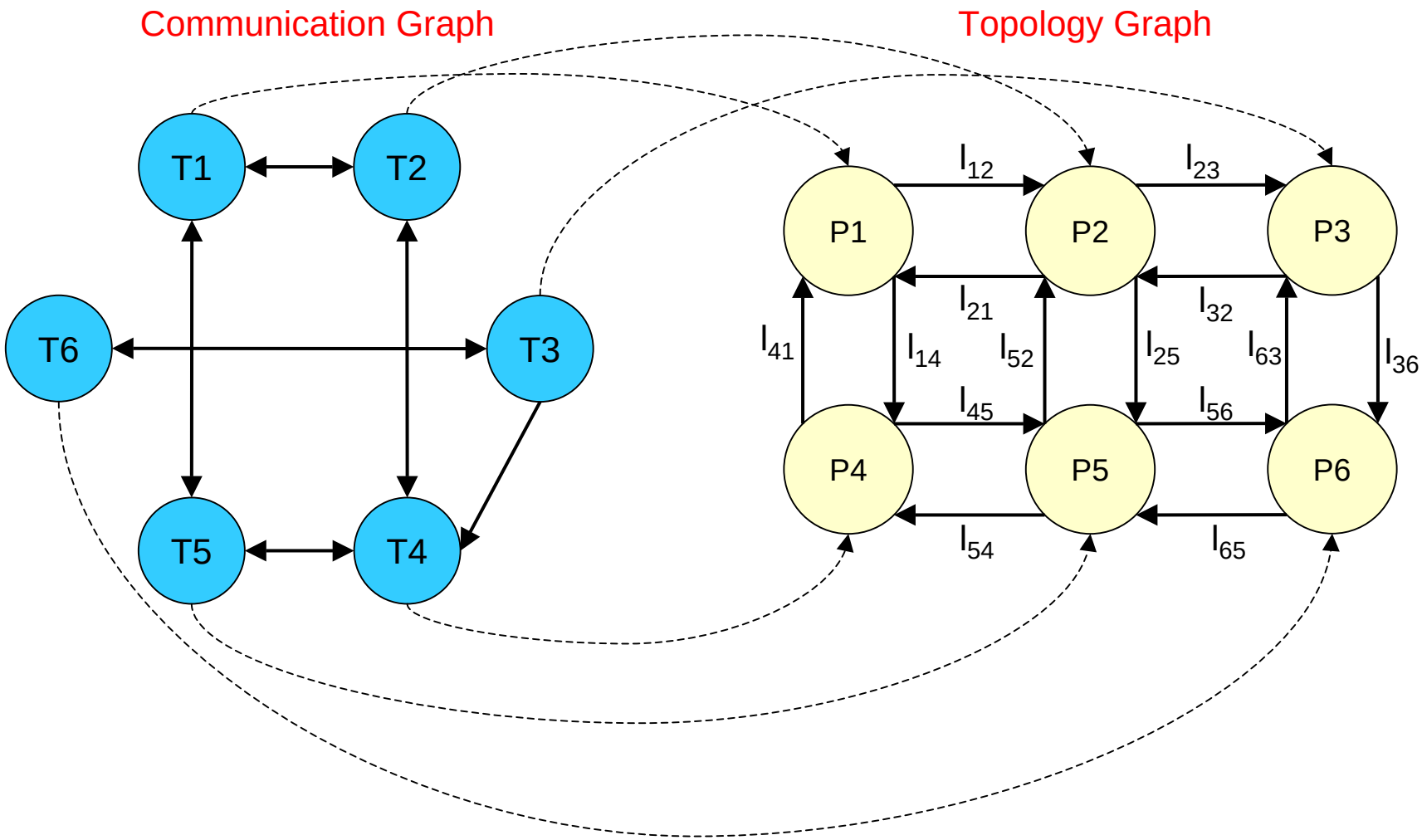
Topology Graph



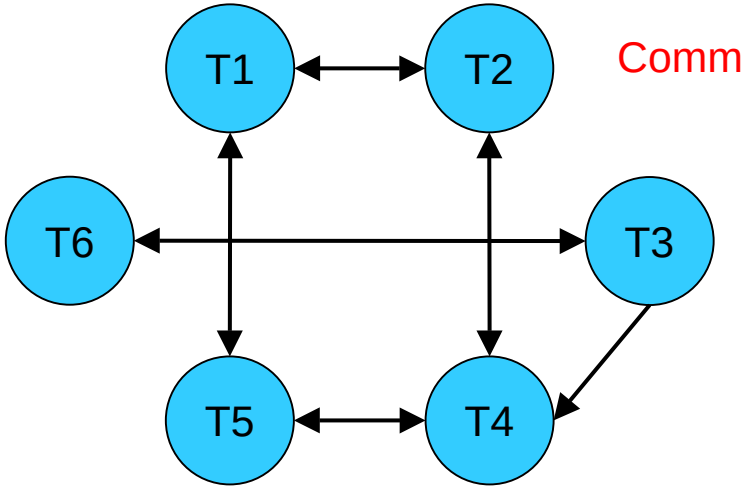
Channel Dependency Graph



APSRA Example (cnt'd)



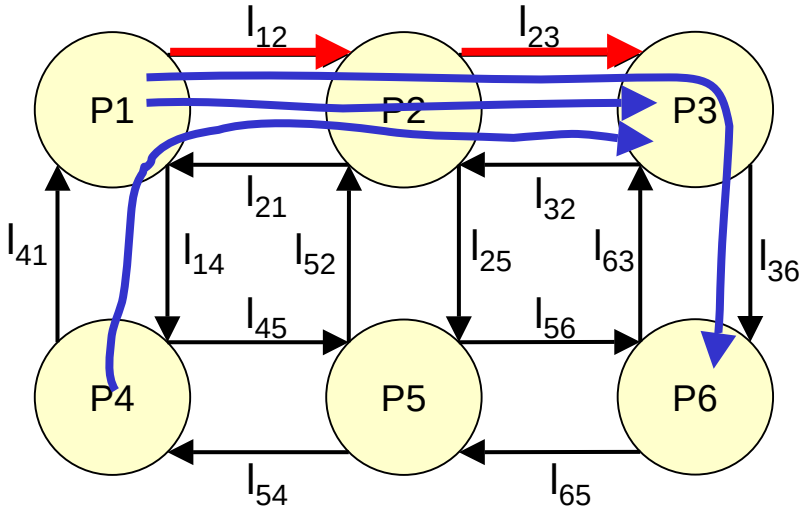
APSRA Example (cnt'd)

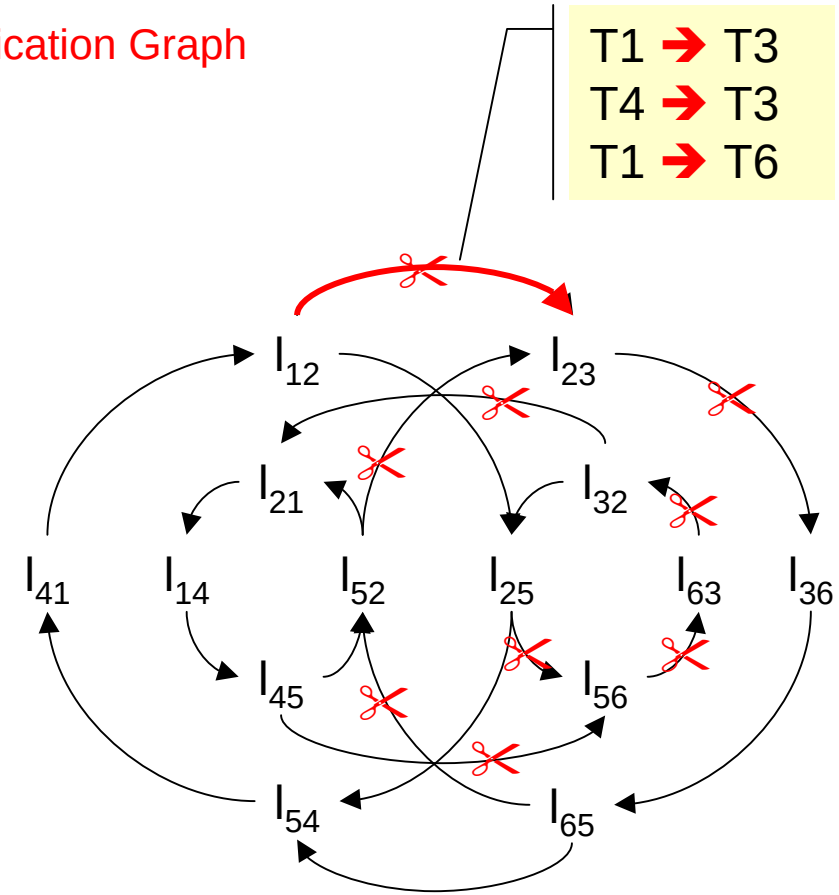


Communication Graph

T1 → T3
T4 → T3
T1 → T6

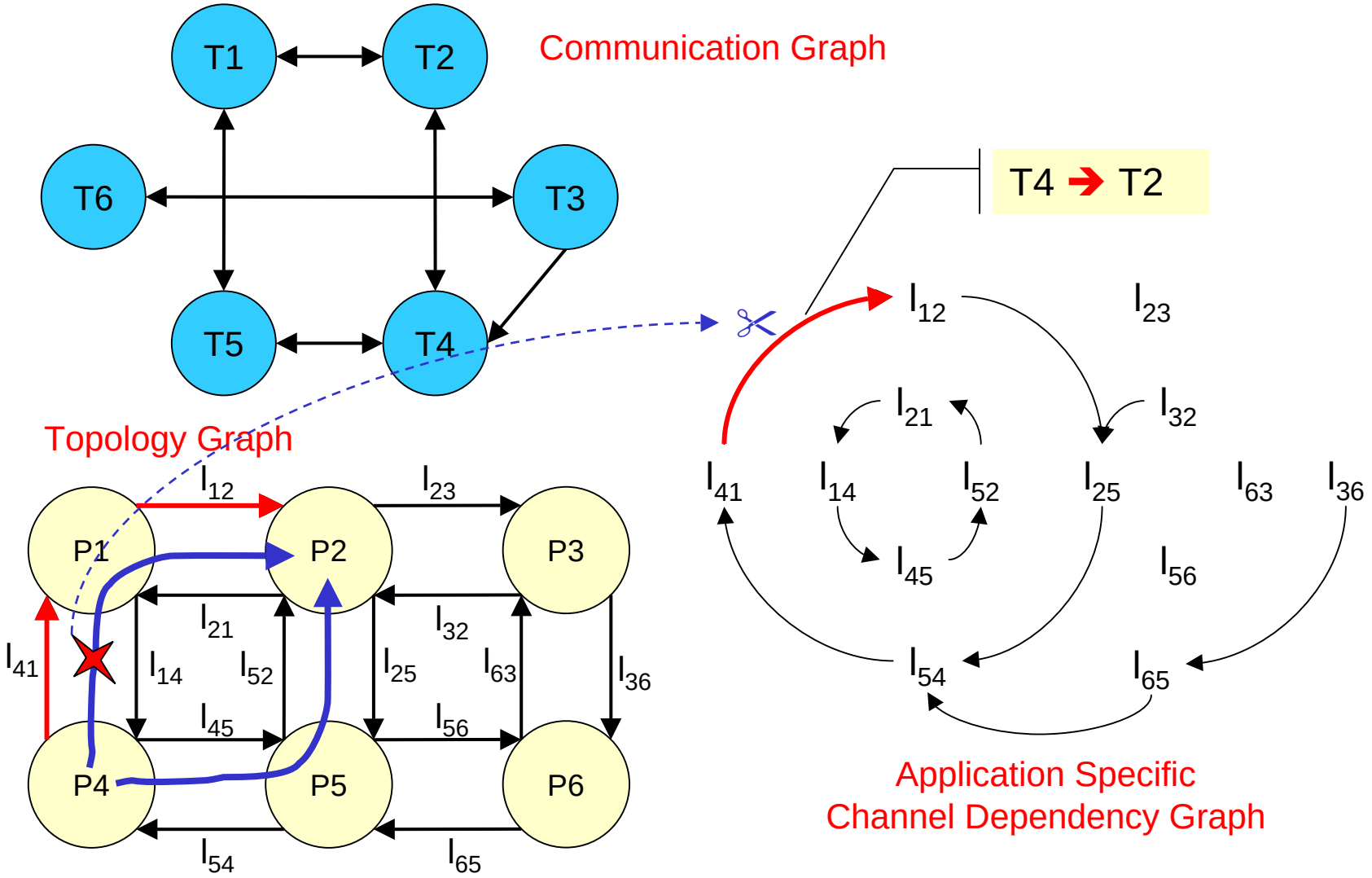
Topology Graph



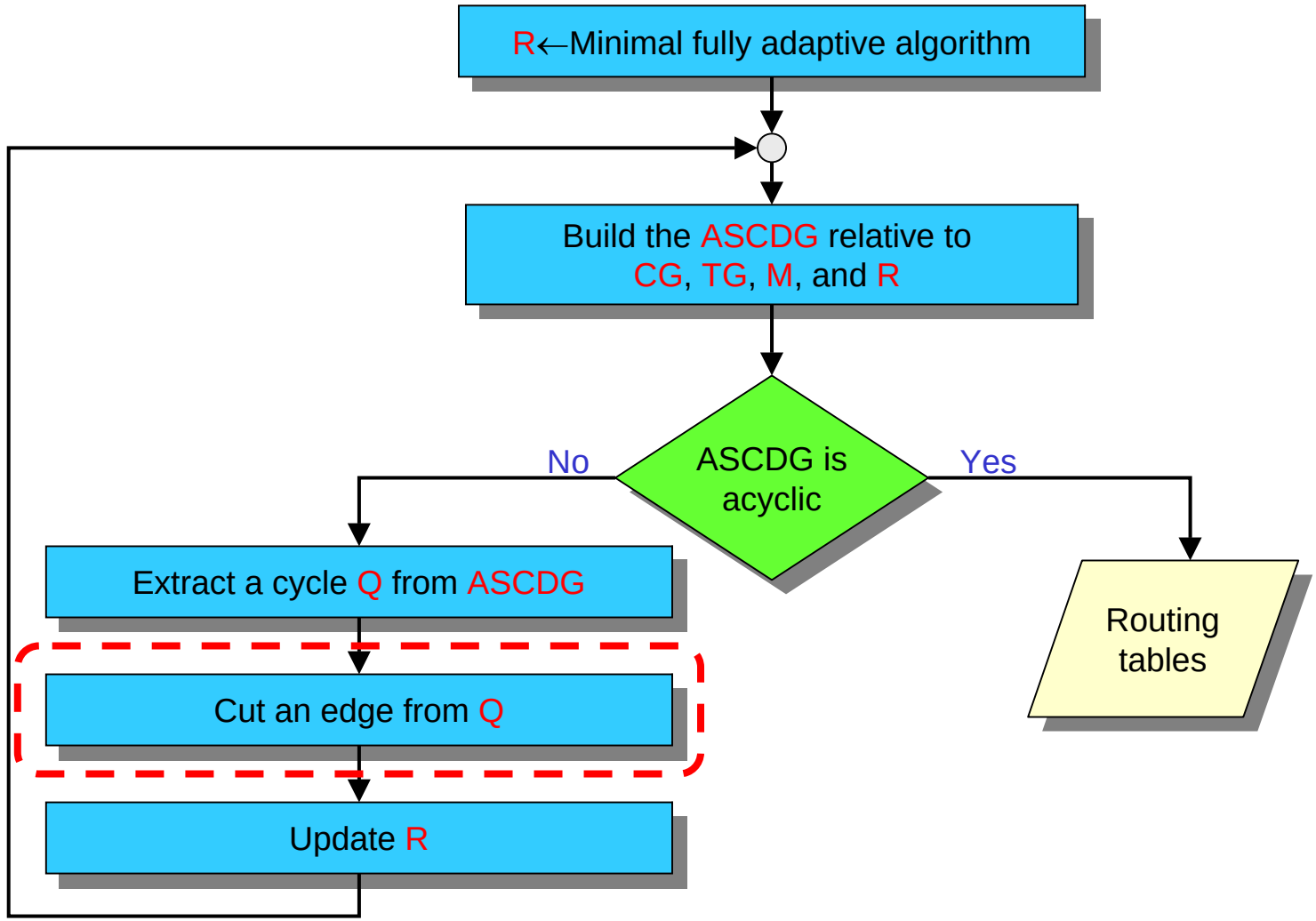


Channel Dependency Graph

APSRA Example (cnt'd)



APSRA Main Algorithm



Cutting Edge with Minimum Loss

- Cutting an edge ➔ Remove a dependency ➔ Remove one or more paths from one or more source-destination pairs
  - ➔ Reduction in adaptivity
  - ➔ Reachability issues
- Select a dependency *d* to be removed
  - ➔ Which satisfy the following reachability constraint

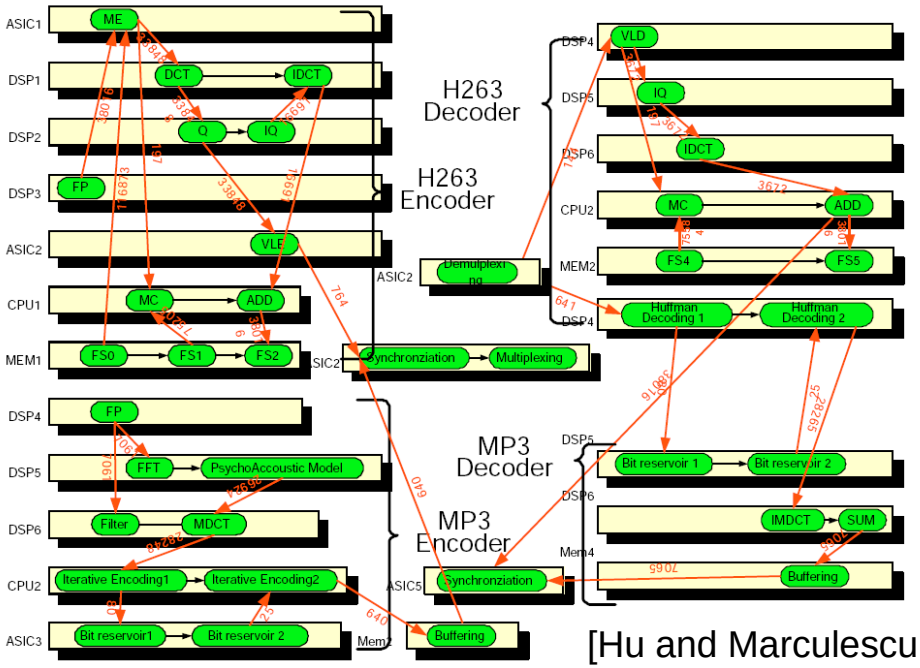
$$\bigwedge_{c \in A(d)} \left| \Phi(c) \right| > 1$$

  - ➔ And minimise

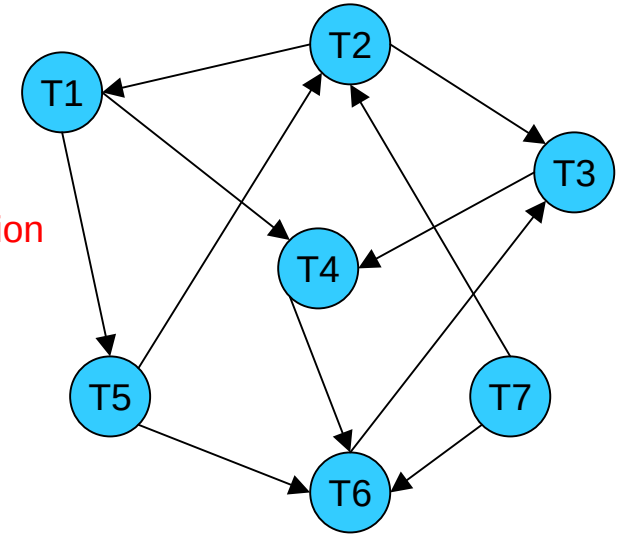
$$\min \sum_{c \in A(d)} \frac{1}{TMP(c)}$$

Experiments

- Three traffic scenarios
  - Uniform random
  - Locality random
  - Multimedia system



Communication Graph

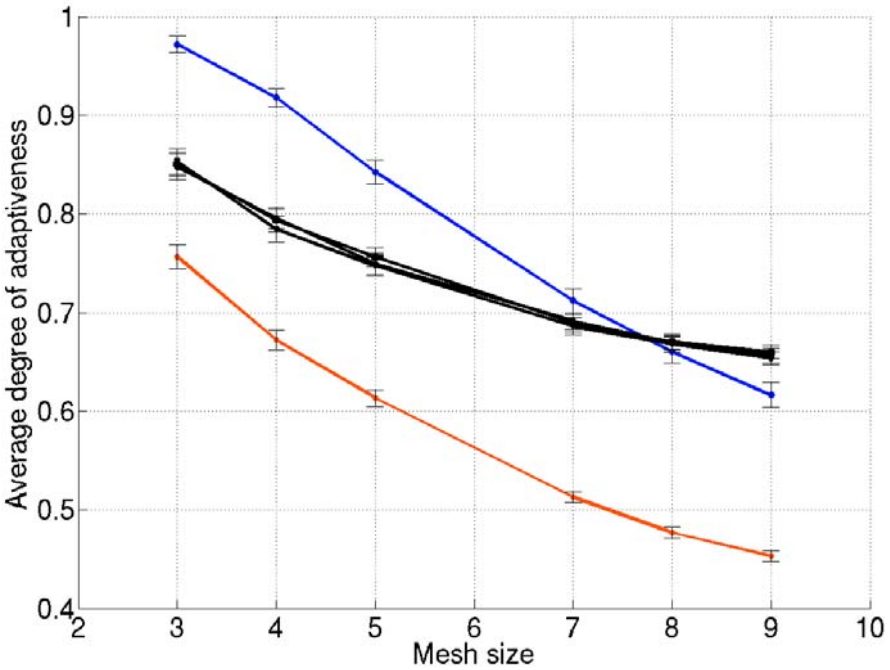


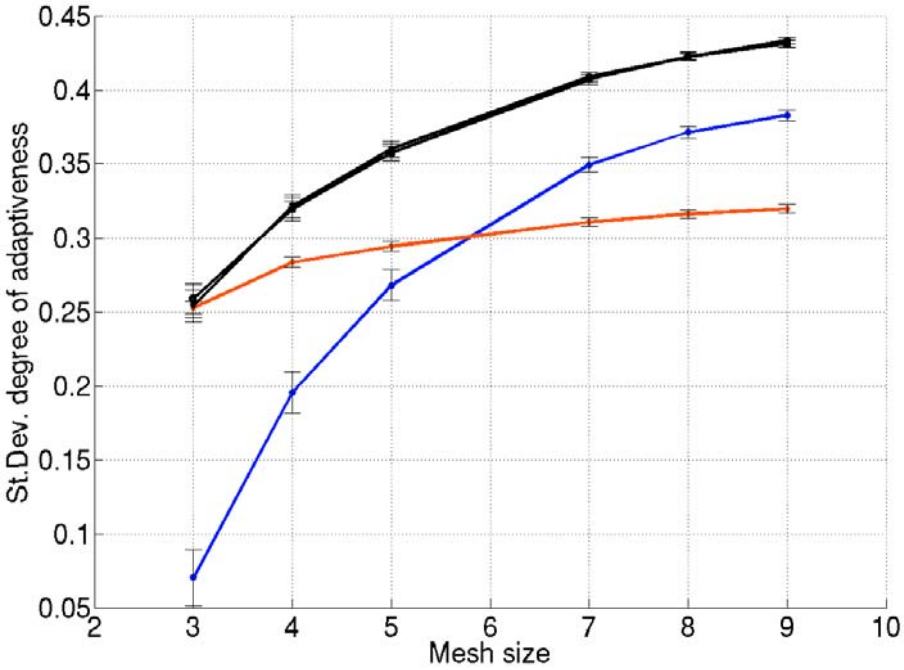
$$\rho = \frac{number\ of\ communications}{number\ of\ tasks}$$

[Hu and Marculescu, TCAD05]

Adaptivity Comparison

Uniform Random

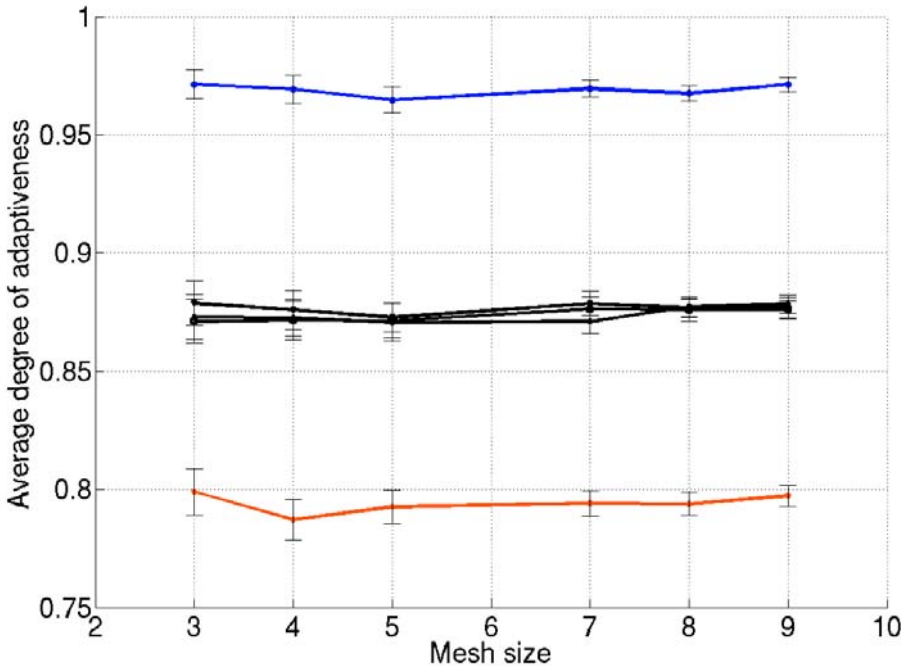


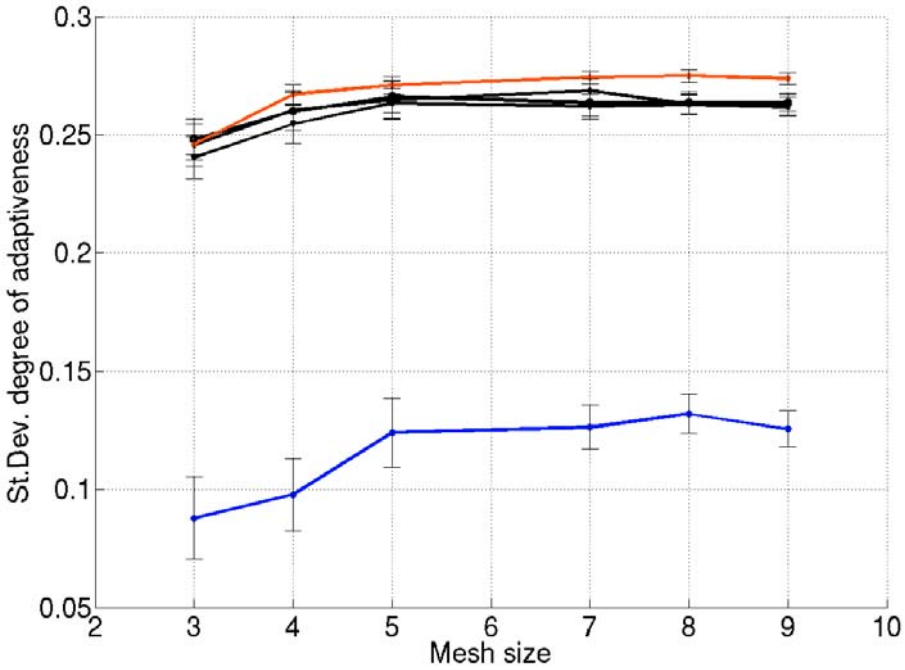


Turn model (West First, North Last, Negative First)
Odd-Even
APSRA

Adaptivity Comparison

Locality

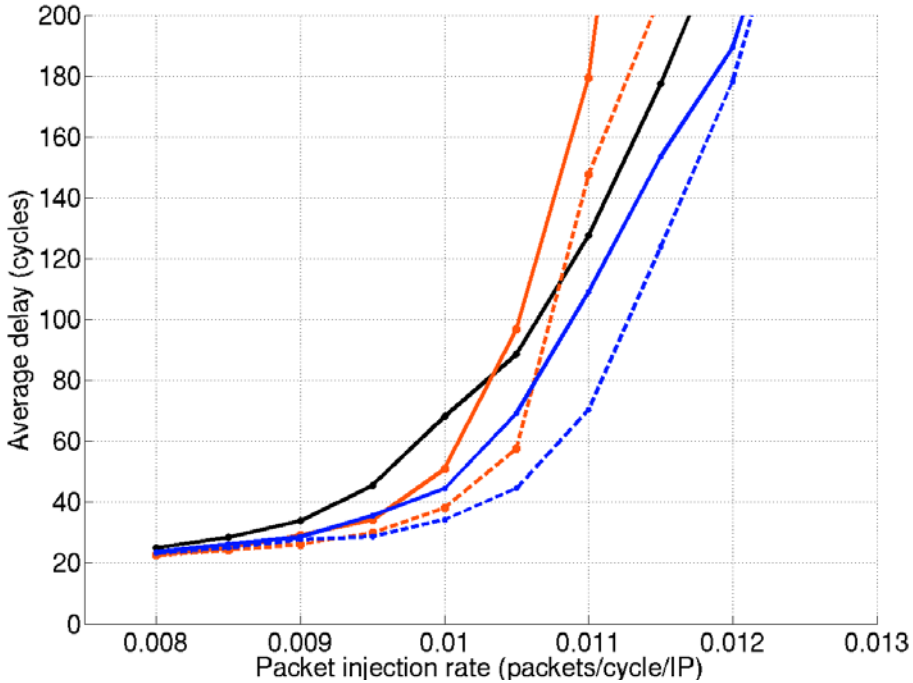


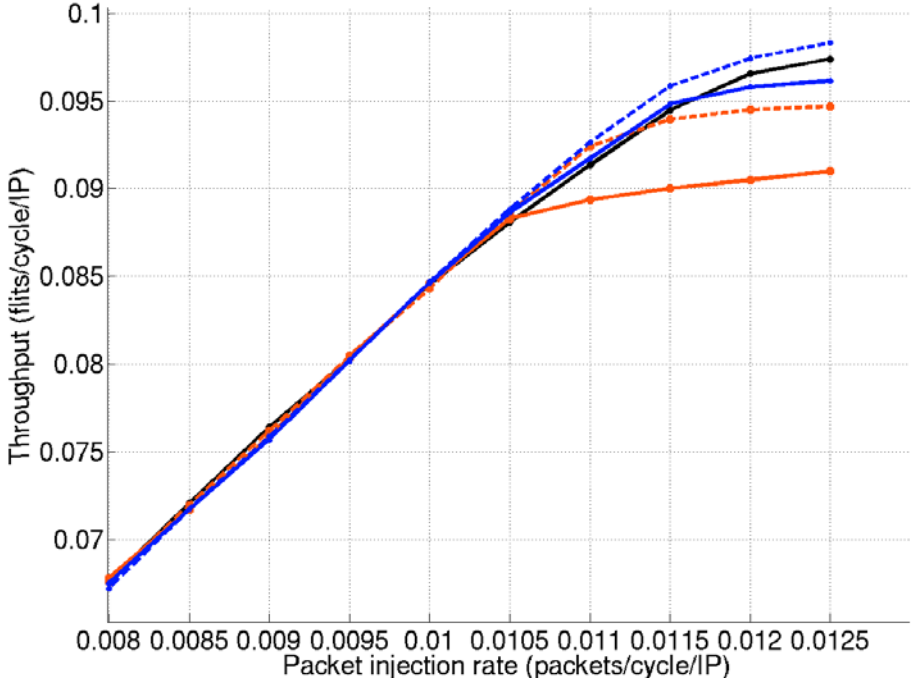


Turn model (West First, North Last, Negative First)
Odd-Even
APSRA

Performance Evaluation

Uniform random, ρ=2

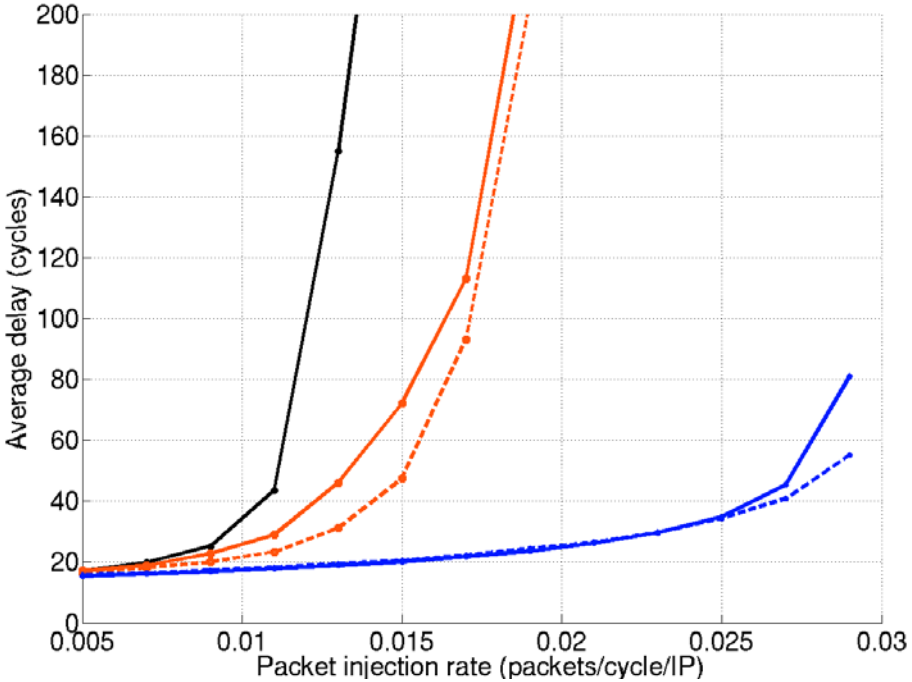


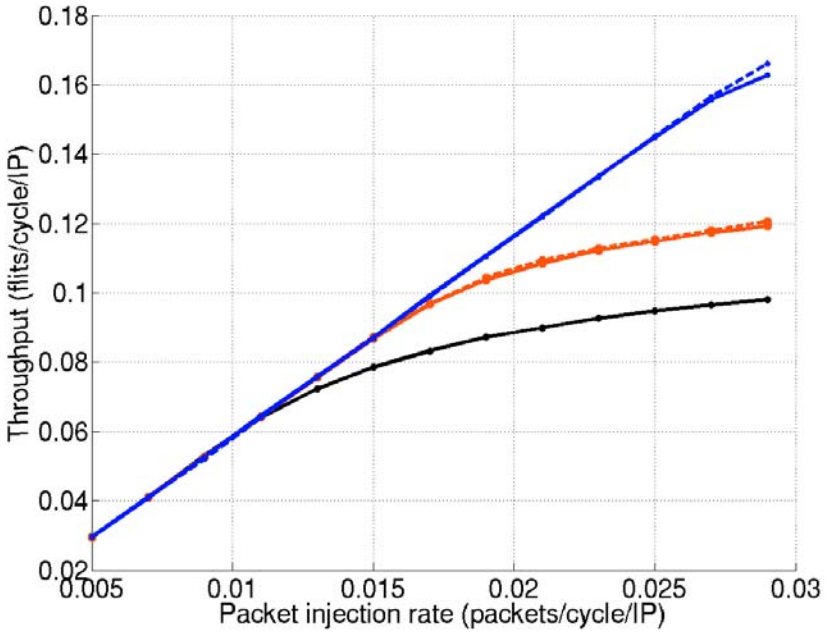


XY
Odd-Even (sel=random)
APSRA (sel=random)
Odd-Even (sel=buffer level)
APSRA (sel=buffer level)

Performance Evaluation

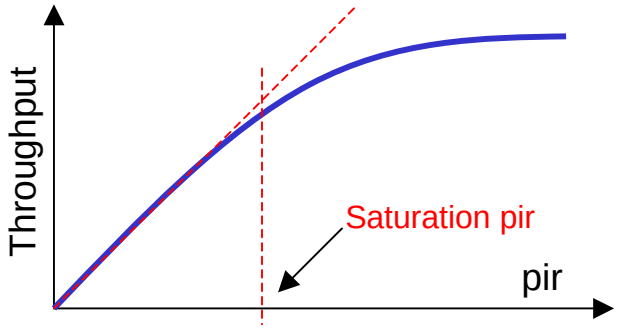
Transpose 1





XY
Odd-Even (sel=random)
APSRA (sel=random)
Odd-Even (sel=buffer level)
APSRA (sel=buffer level)

Results Summary

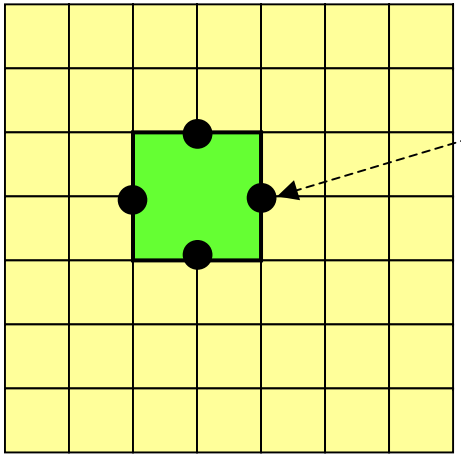


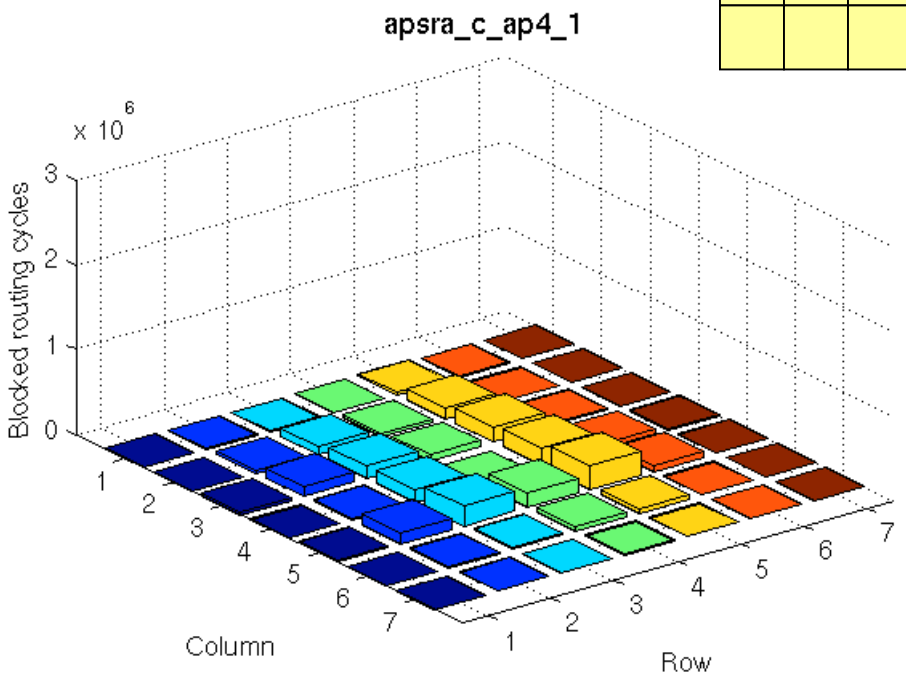
Traffic scenario	Max. pir (packets/cycle/IP) XY	OE	APSRA	APSRA improvement vs. XY	vs. OE
Random	0,012	0,0105	0,012	0,0%	14,3%
Locality	0,019	0,02	0,021	10,5%	5,0%
Transpose 1	0,011	0,015	0,027	145,5%	80,0%
Transpose 2	0,011	0,016	0,027	145,5%	68,8%
Hotspot-4c	0,0033	0,0035	0,0038	13,6%	7,1%
Hotspot-4tr	0,0027	0,0031	0,0035	29,6%	12,9%
Hotspot-8r	0,0039	0,0059	0,0067	71,8%	13,6%
Mms	0,0174	0,0174	0,0196	12,6%	12,6%
Average improvement				***53,6%***	***26,8%***

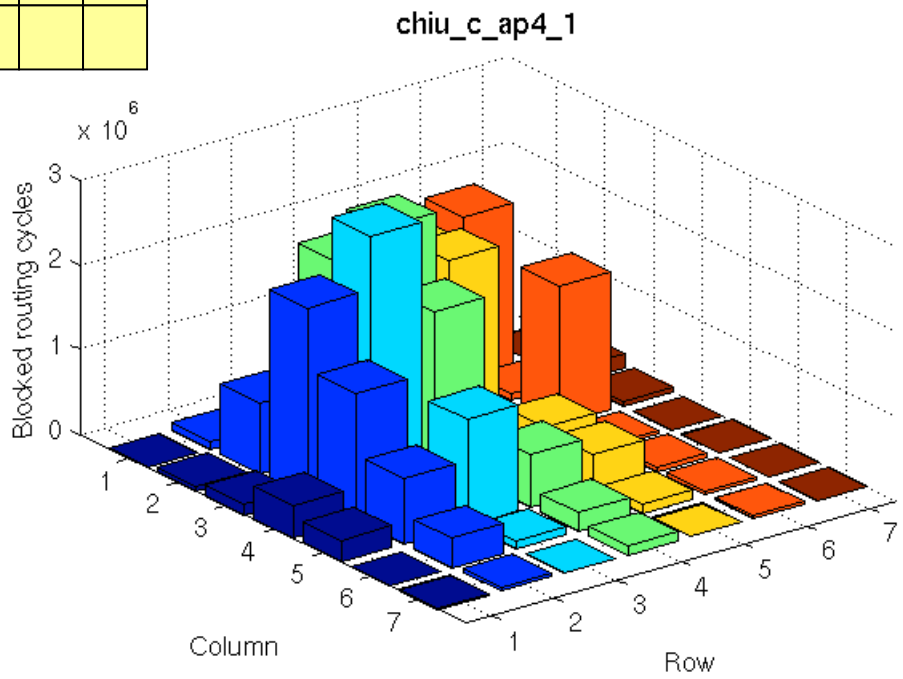
Results Summary (Delay)

Traffic scenario	pir (pkts/cycle/IP)	Average delay (cycles) XY	OE	APSRA	APSRA improvement vs. XY	vs. OE
Random	0,01	68	51	34	49,8%	32,8%
Locality	0,02	39	34	29	24,2%	13,2%
Transpose 1	0,011	91	39	19	79,2%	51,1%
Transpose 2	0,011	82	31	19	76,6%	38,4%
Hotspot-4c	0,003	46	50	34	26,7%	32,1%
Hotspot-4tr	0,003	52	37	30	42,2%	17,5%
Hotspot-8rs	0,003	34	25	20	41,8%	21,5%
Mms	0,02	36	30	23	36,1%	23,3%
				Average improvement	***47,1%***	***28,7%***

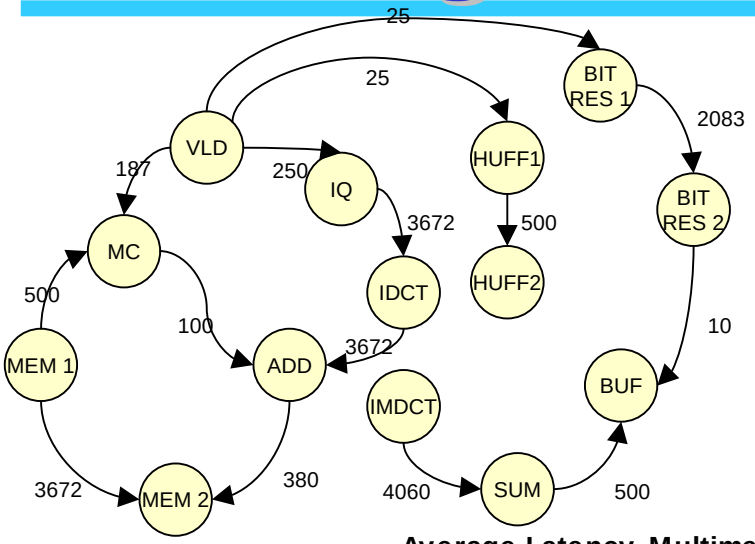
Heterogeneous 2D Mesh





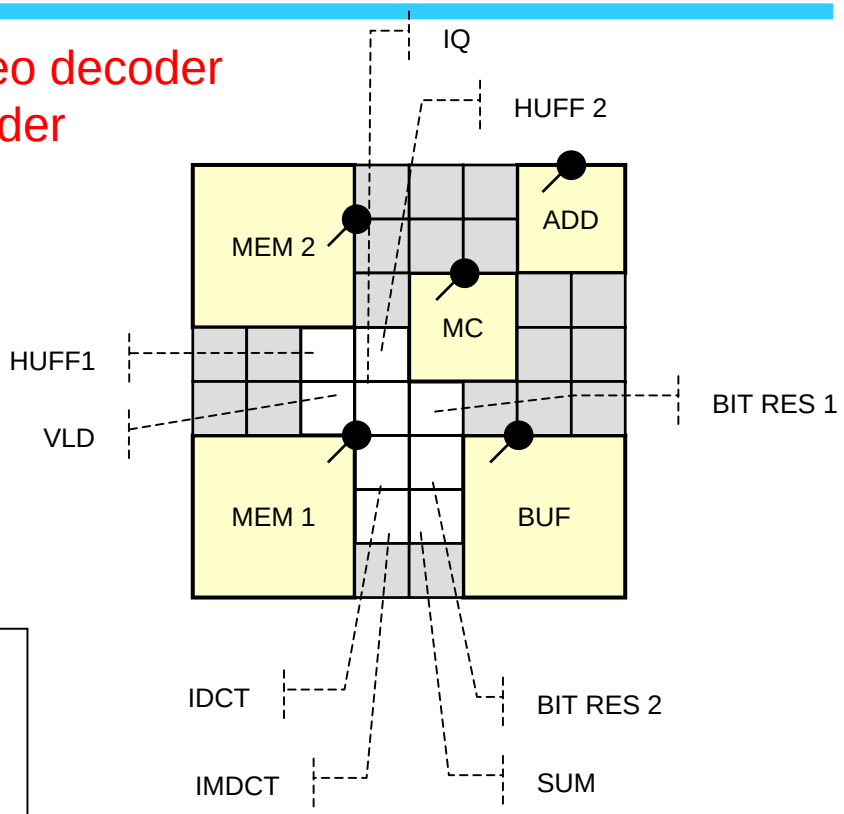


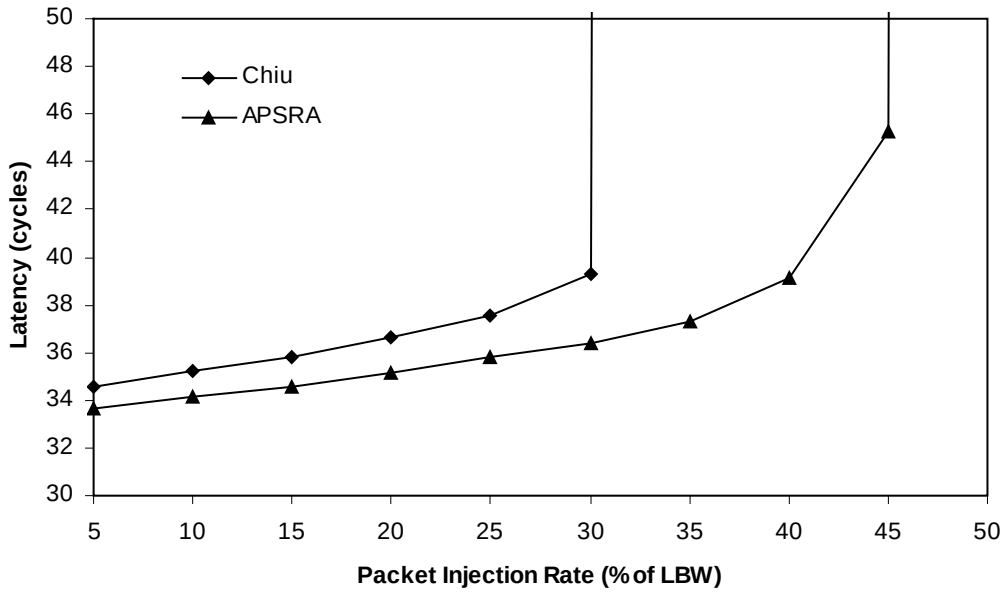
Heterogeneous 2D Mesh



Average Latency, Multimedia

H.263 video decoder
MP3 decoder





Strengths and Weakness

- APSRA properties
  - Exploting communication information
  - General applicability (topology independent)
  - High degree of adaptiveness
  - High performance
- Limitations
  - Table-based router
  - Area overhead (10%-15%)
    - Routing table compression [Palesi *et al.*, SAMOS VI]

Generic Mesh Topology Router

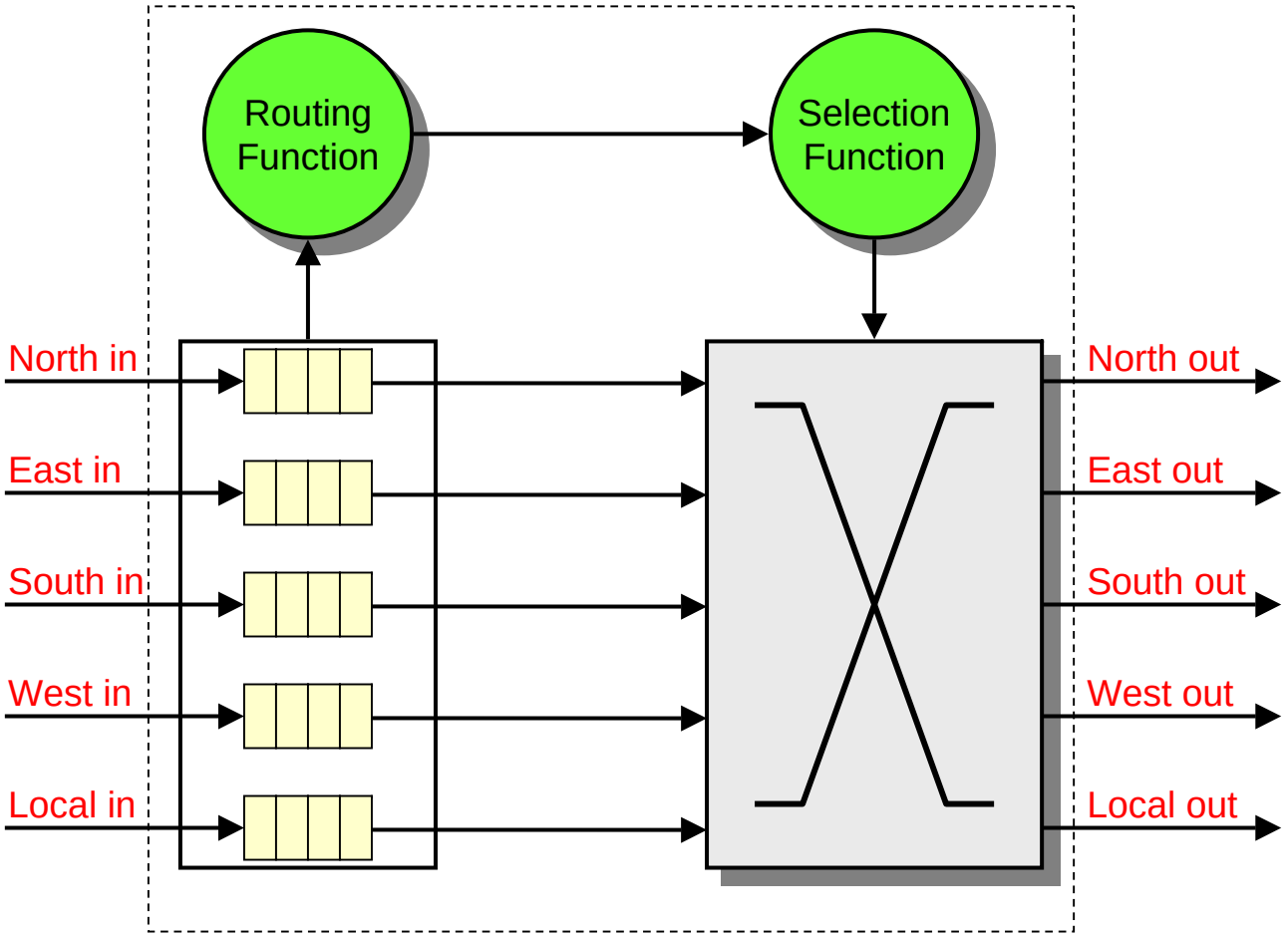
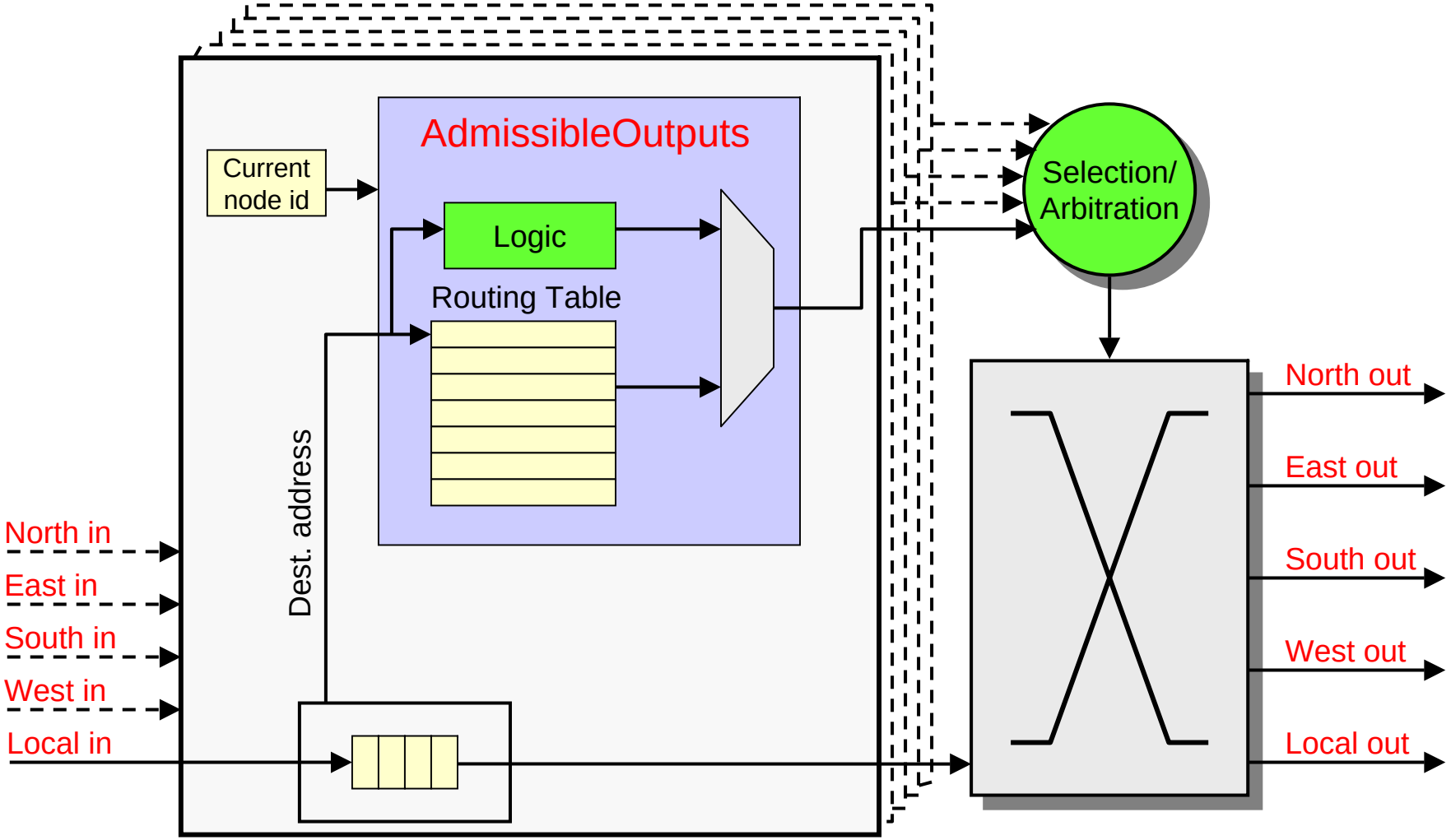
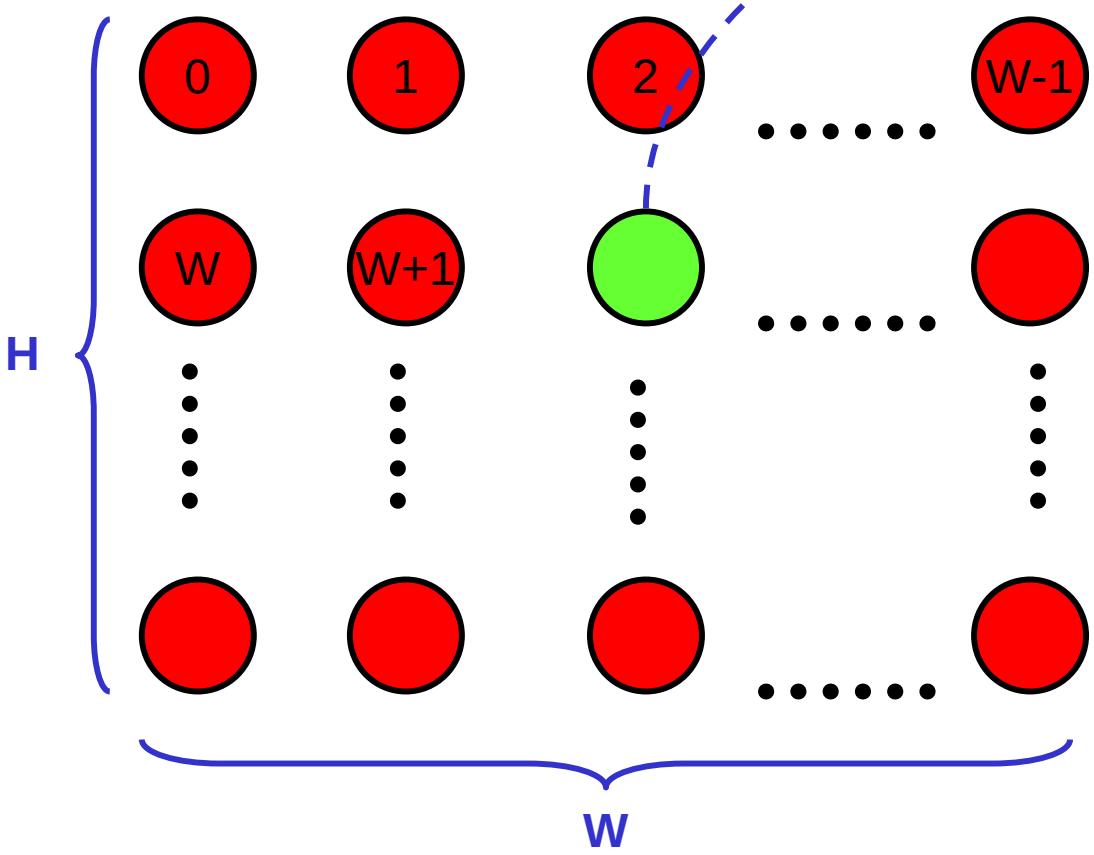


Table-Based NoC Router



Routing Table



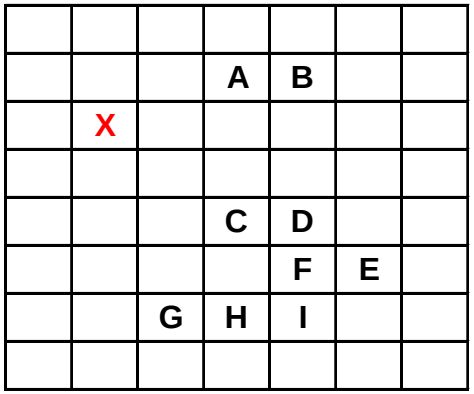
Destination id	Admissible outputs
1	South
2	North, East
3	West
4	South, West
5	South
…	…
W*H-1	South, East

N	S	E	W

4 bit

RT Size = (W*H-1)*4 [bits]

Compression Algorithm



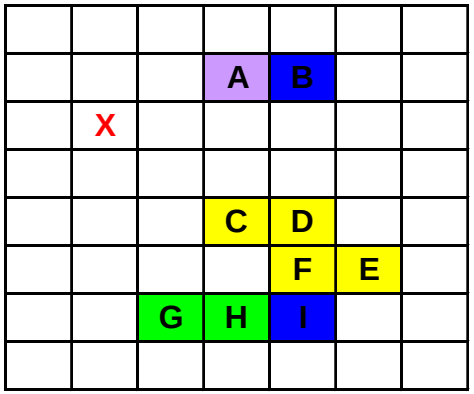
Dest	Admissible outputs
A	North, East
B	East
C	South, East
D	South, East
E	South, East
F	South, East
G	South
H	South
I	East

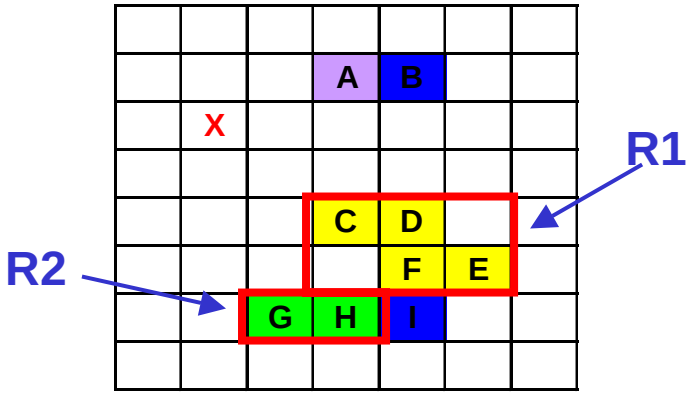
North, East

East

South, East

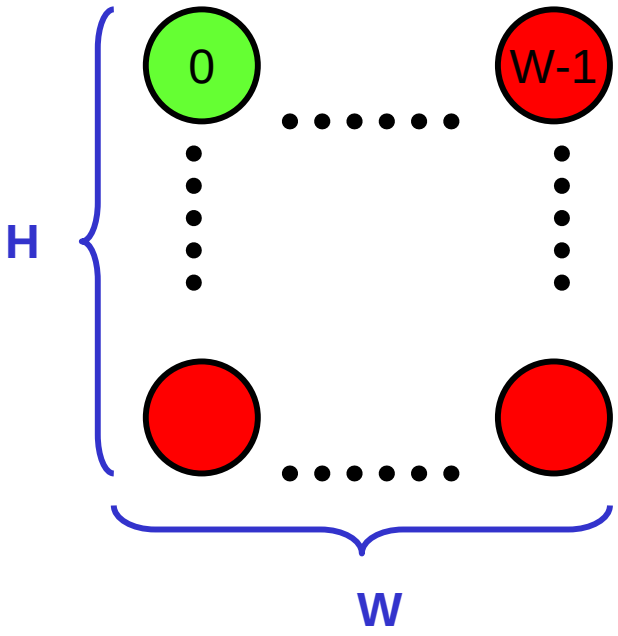
South





Dest	Admissible outputs
A	North, East
B	East
R1	South, East
R2	South
I	East

Compressed Routing Table Size



North

Region ID	Color
TL1, BR1	Green
TL2, BR2	Red
…	
TLm, BRm	Blue

South

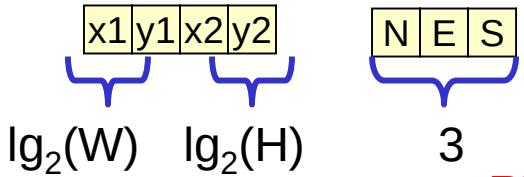
Region ID	Color
TL1, BR1	Green
TL2, BR2	Red
…	
TLm, BRm	Blue

West

Region ID	Color
TL1, BR1	Green
TL2, BR2	Red
…	
TLm, BRm	Blue

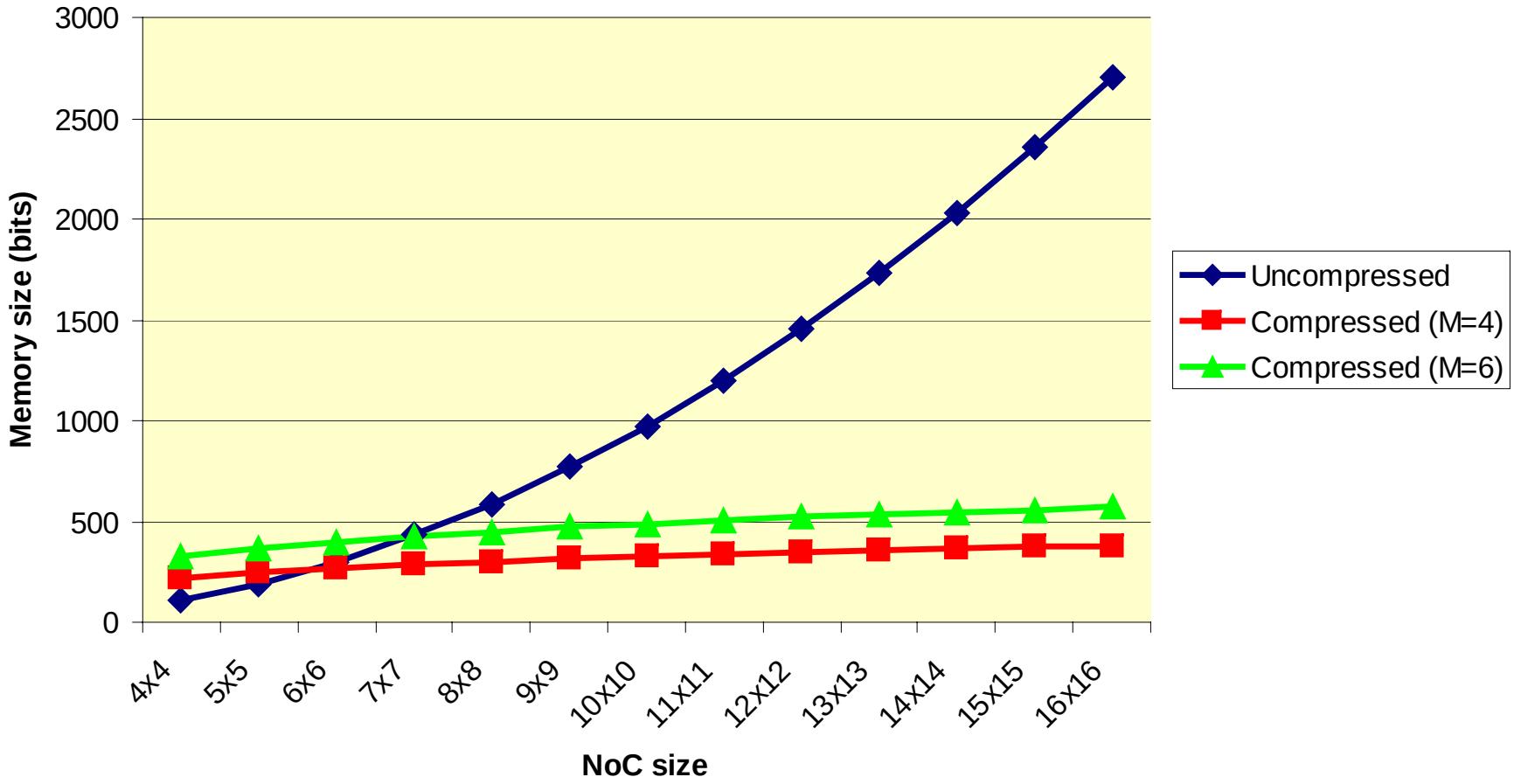
East

Region ID	Color
TL1, BR1	Green
TL2, BR2	Red
…	
TLm, BRm	Blue

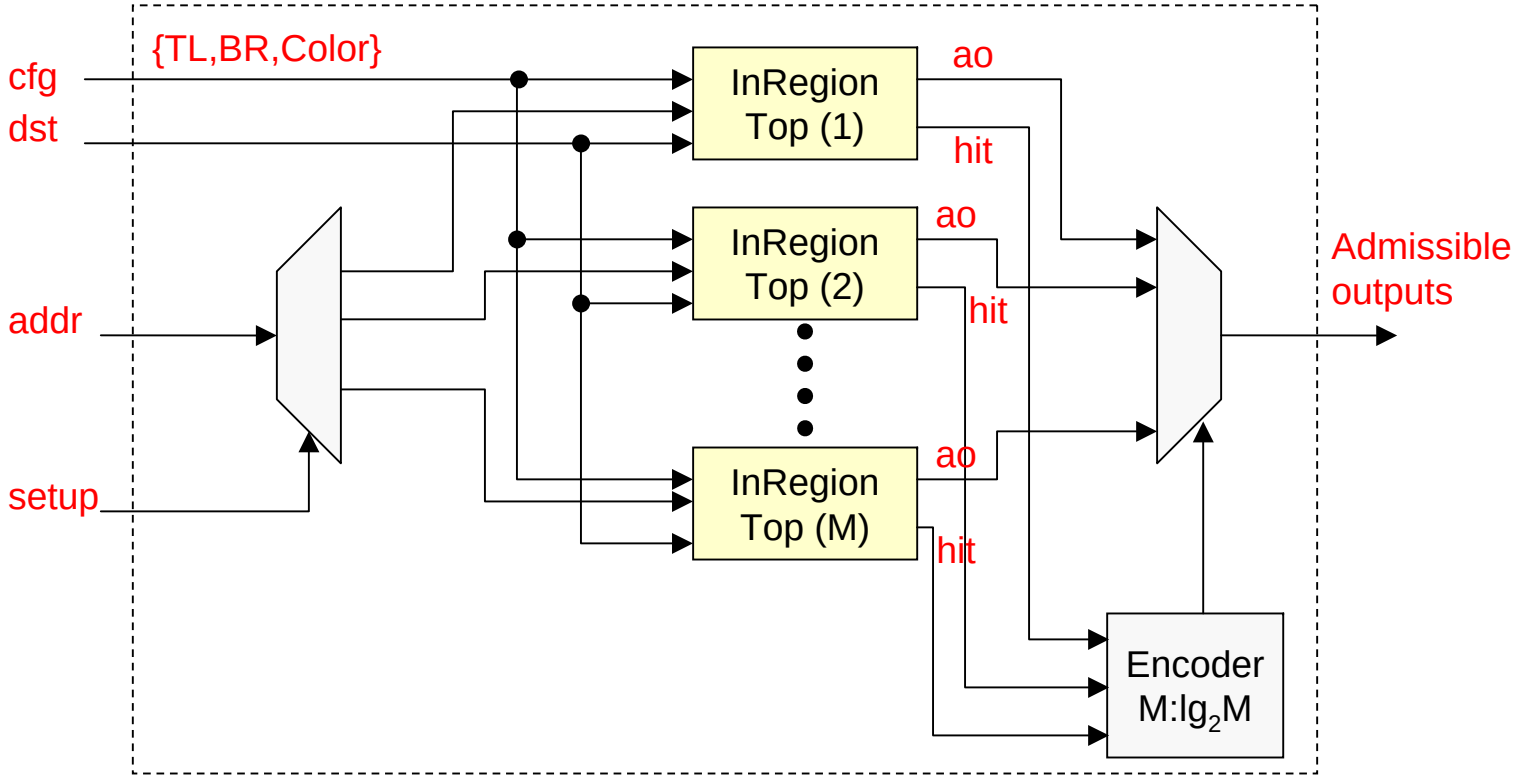


RT Size = $5*M*[3+\lg_2(W*H)^2]$ [bits]

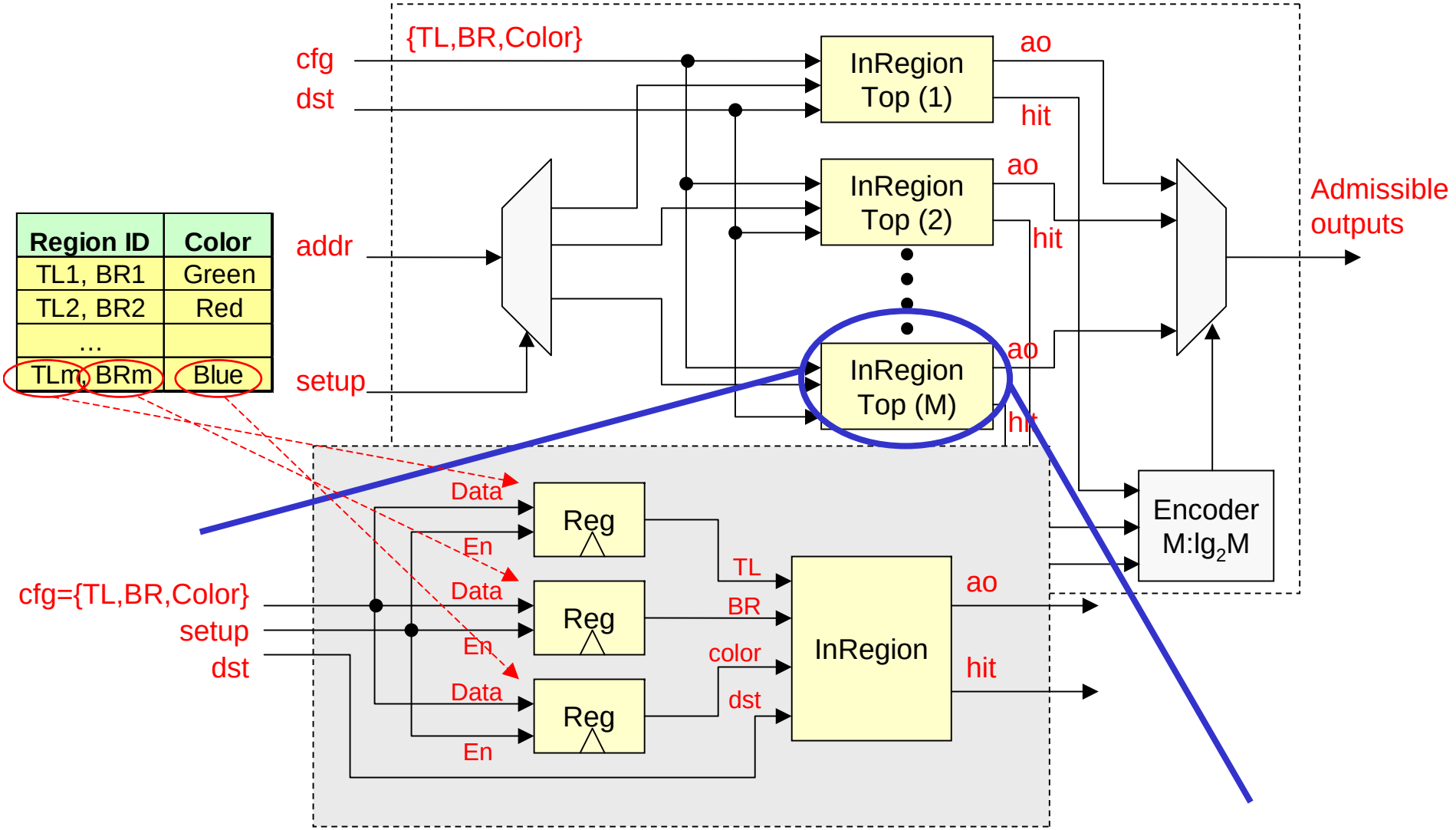
Router Table Size



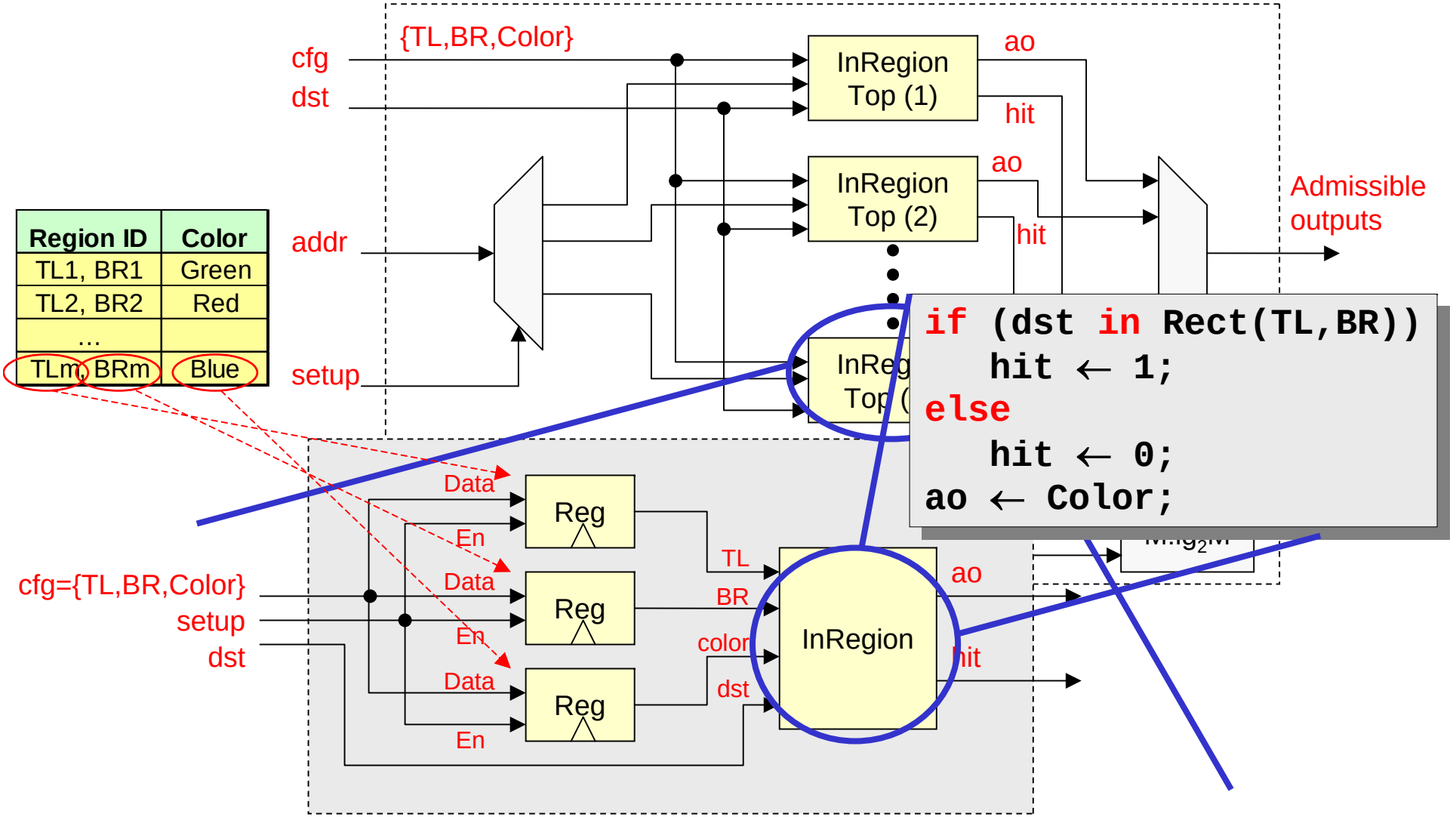
Block Diagram



Block Diagram

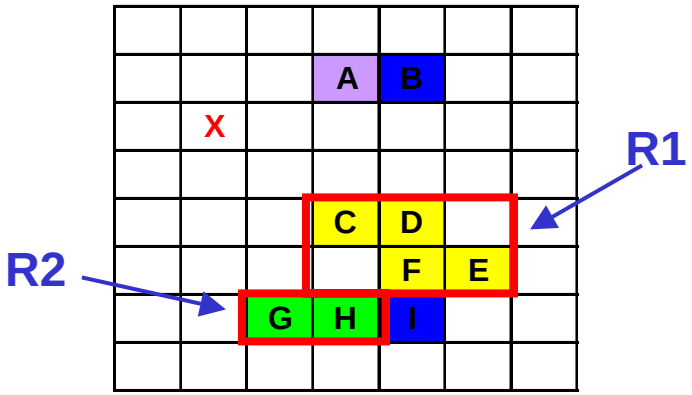


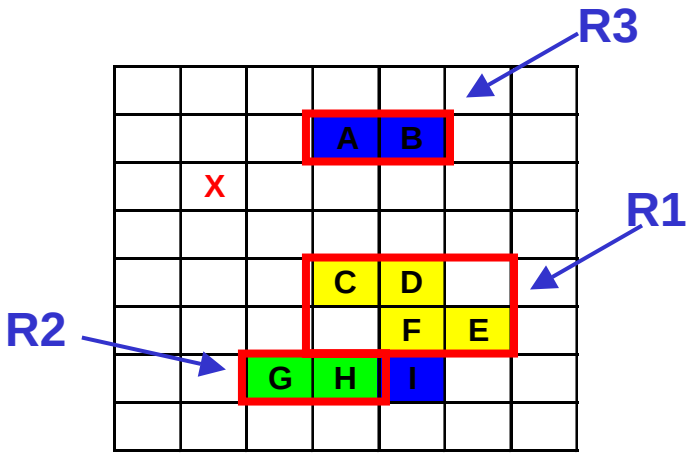
Block Diagram



Compression Algorithm (Lossy)

Dest	Admissible outputs
A	North, East
B	East
R1	South, East
R2	South
I	East

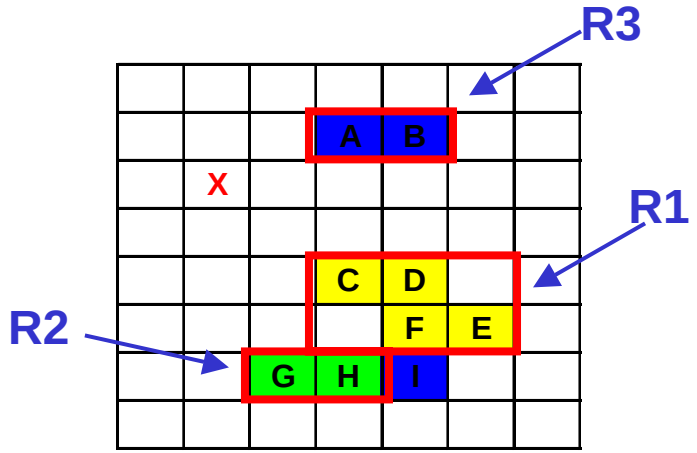


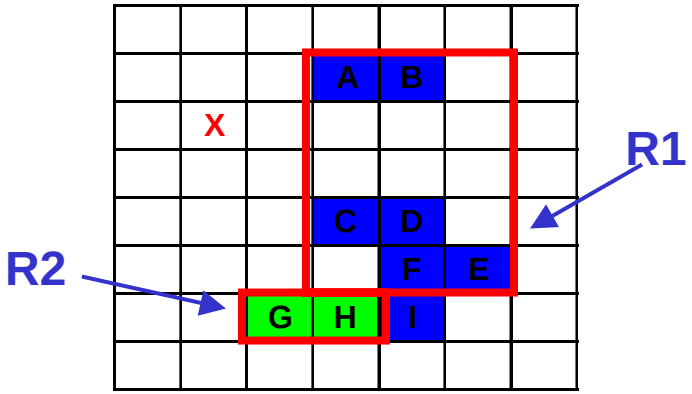


Dest	Admissible outputs
R3	East
R1	South, East
R2	South
I	East

Compression Algorithm (Lossy)

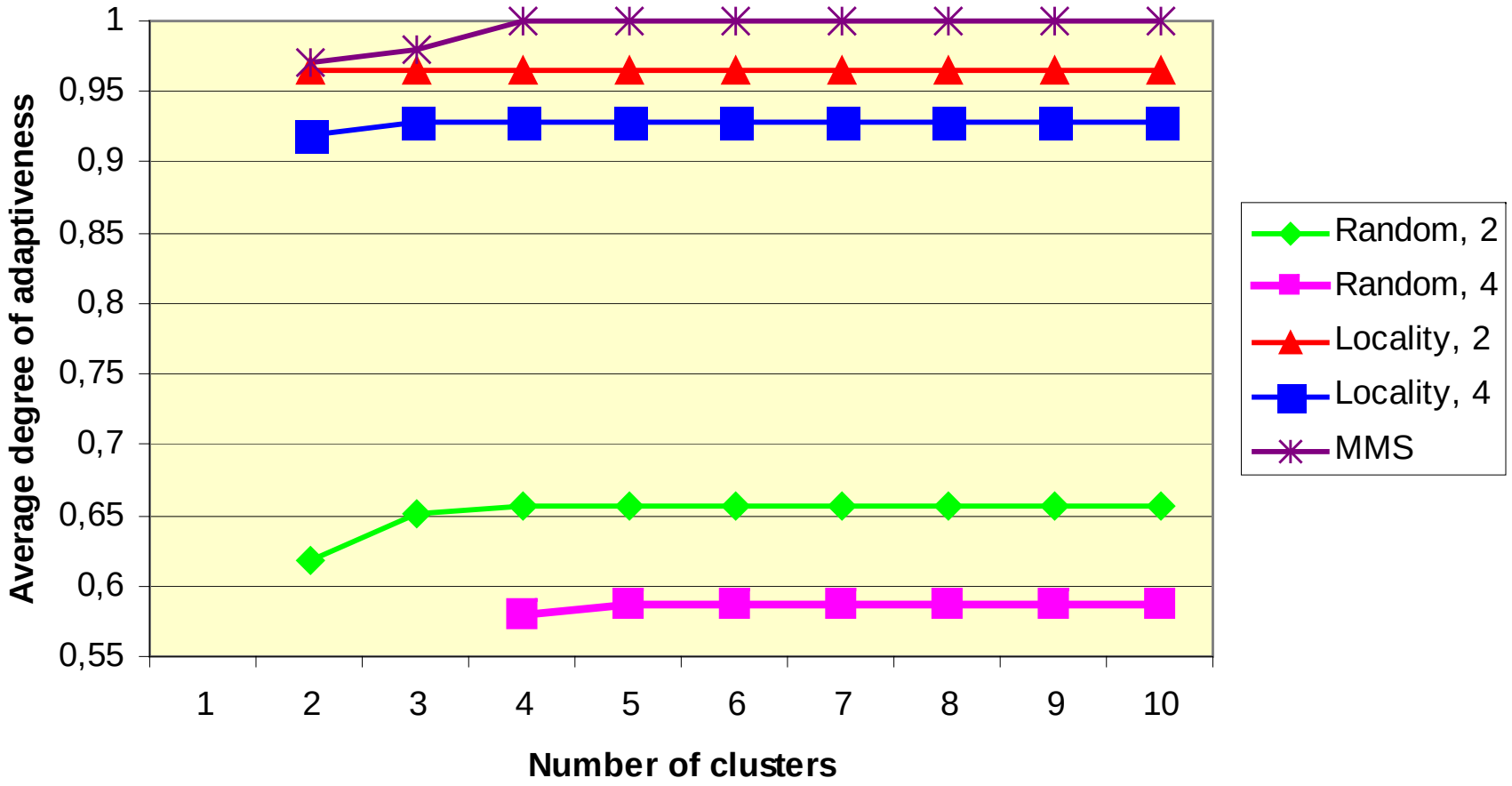
Dest	Admissible outputs
R3	East
R1	South, East
R2	South
I	East



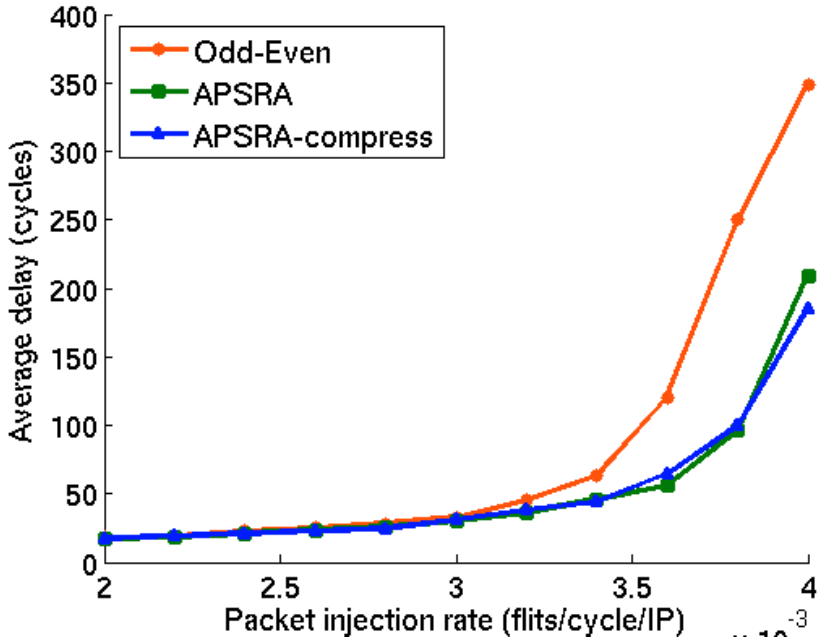


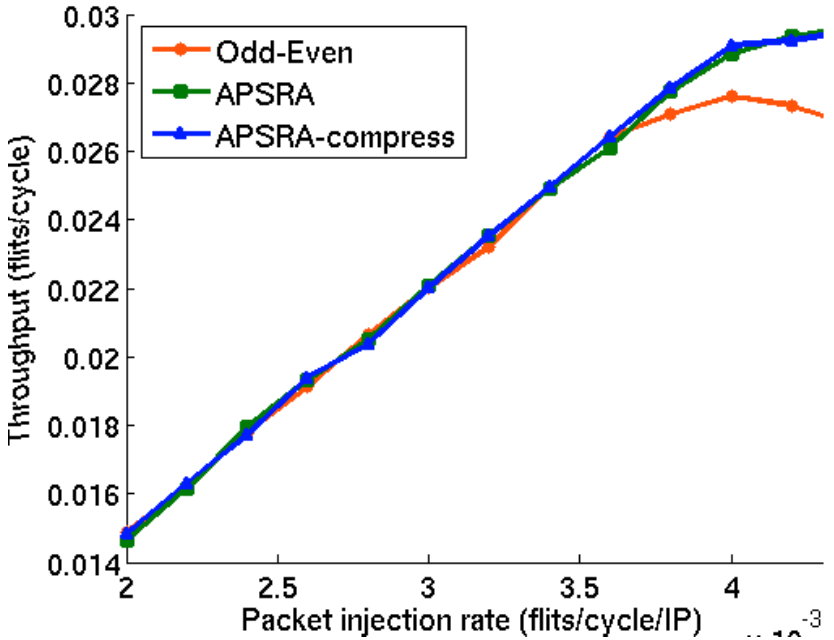
Dest	Admissible outputs
R1	East
R2	South
I	East

Degree of Adaptiveness



Performance Evaluation



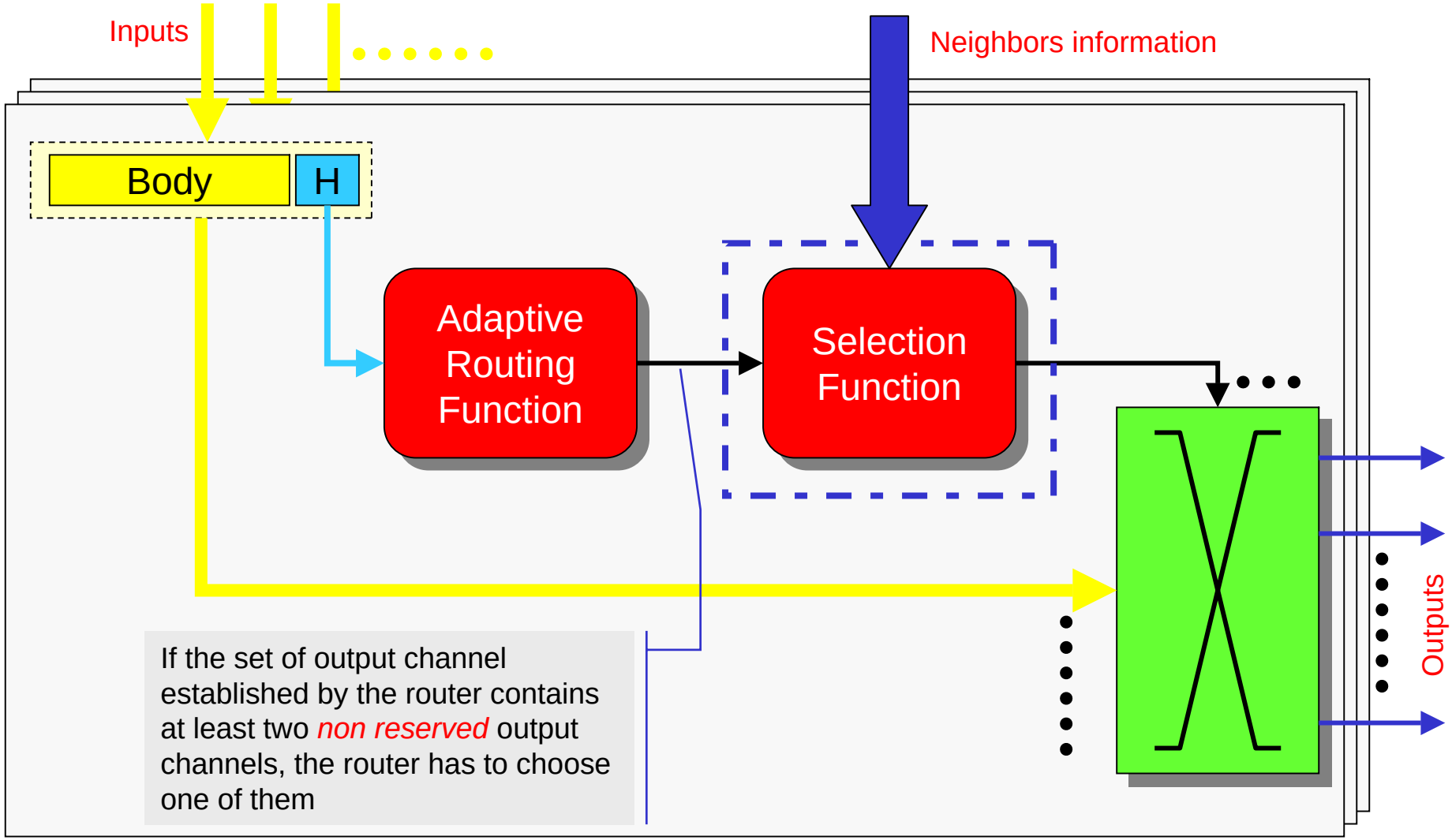


Design Issues

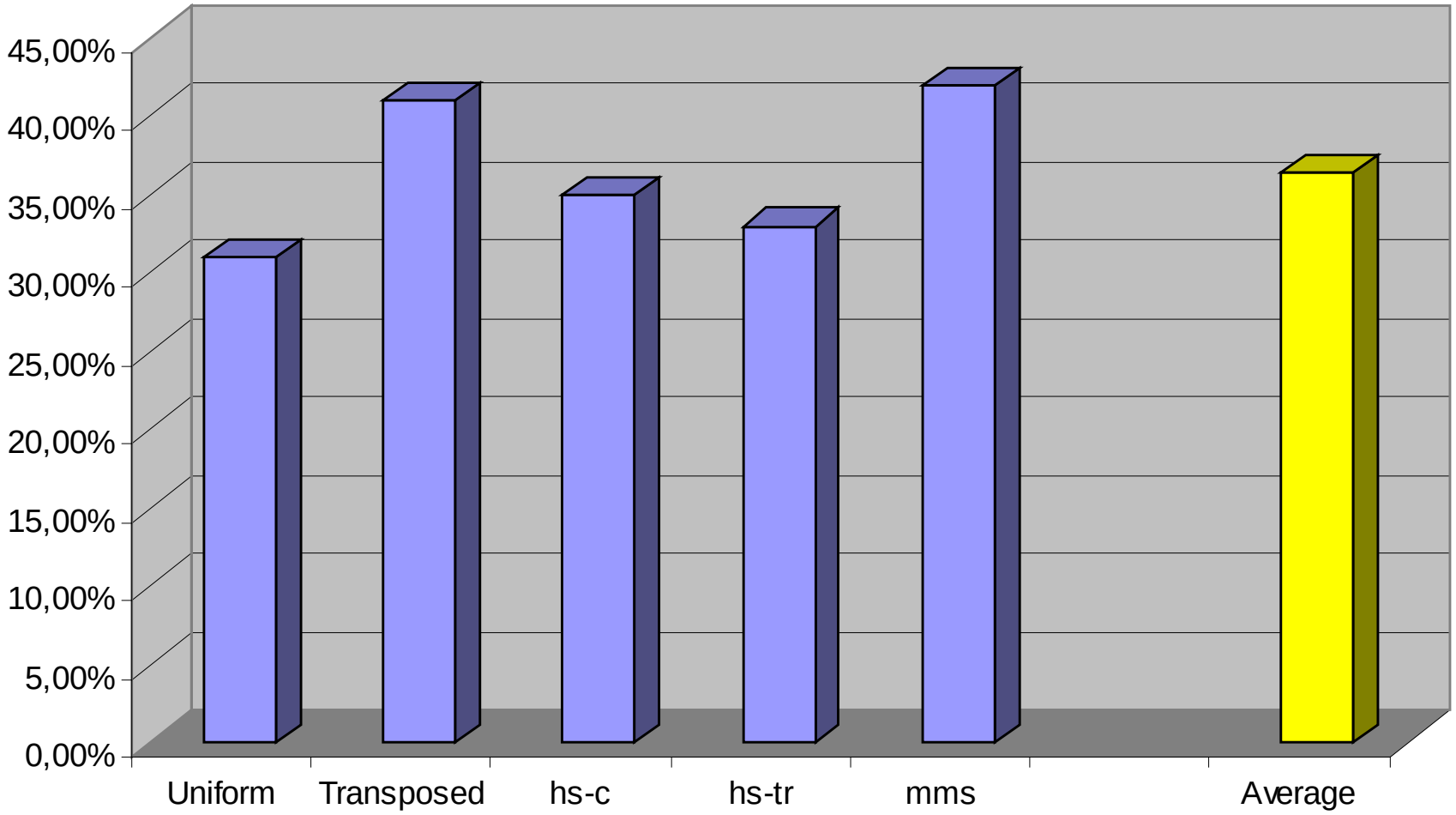
	XY	OddEven	RT4	RT8
Routing function	160,70	273,02	4262,98	9096,19
Input FIFO x 5	132891,84	132891,84	132891,84	132891,84
Crossbar	14041,73	14041,73	14041,73	14041,73
Arbiter	2007,94	2008,94	2009,94	2010,94
Total	149102,20	149215,52	153206,48	158040,69
RT4 Overhead	2,8%	2,7%		
RT8 Overhead	6,0%	5,9%		

- Values obtained after synthesis with Synopsys Design Compiler for a UMC 0.13μm technology library

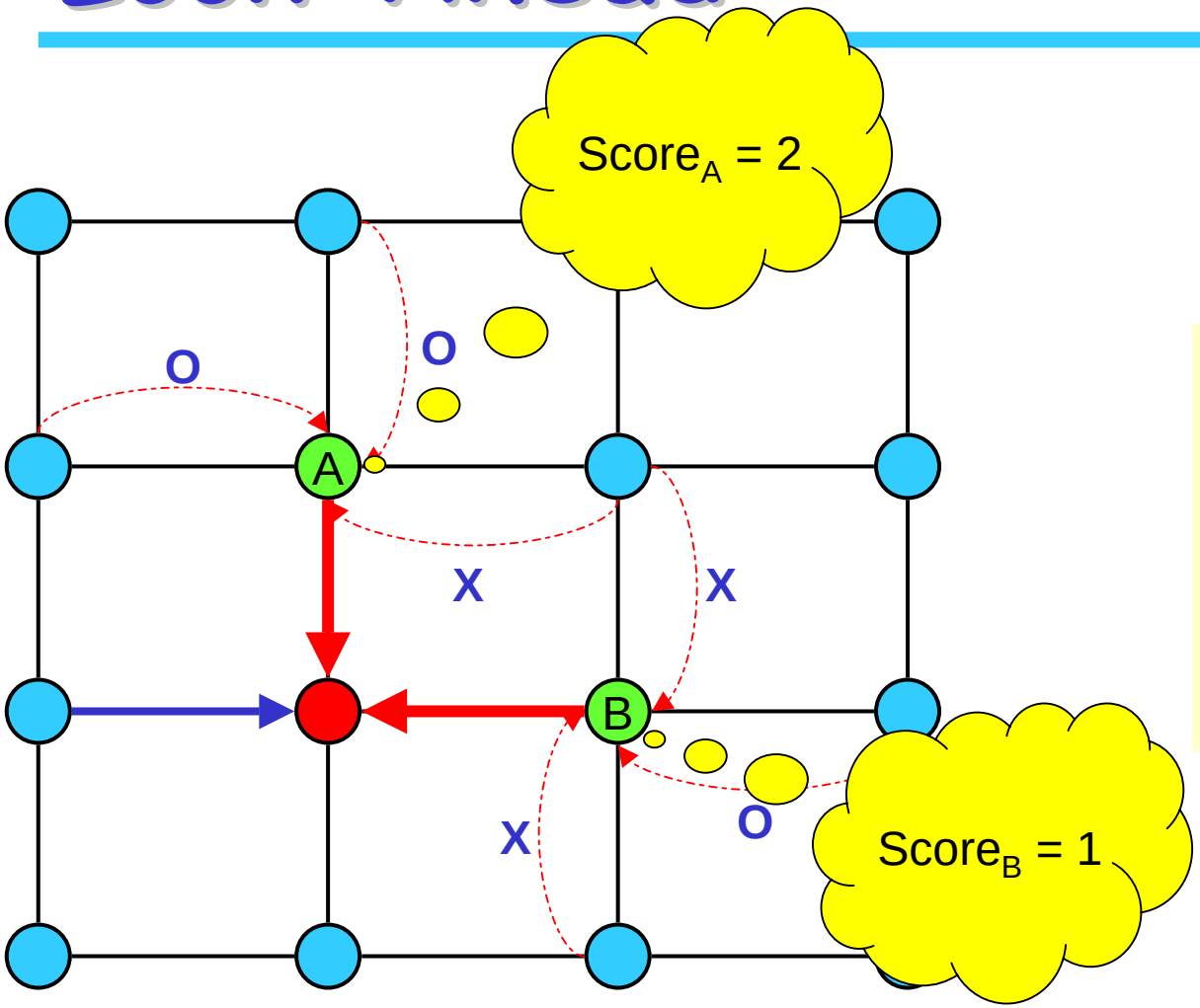
Routing and Selection

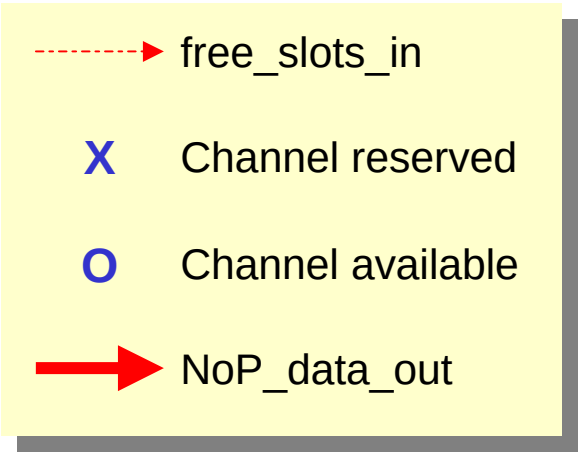


Situations of Indecision

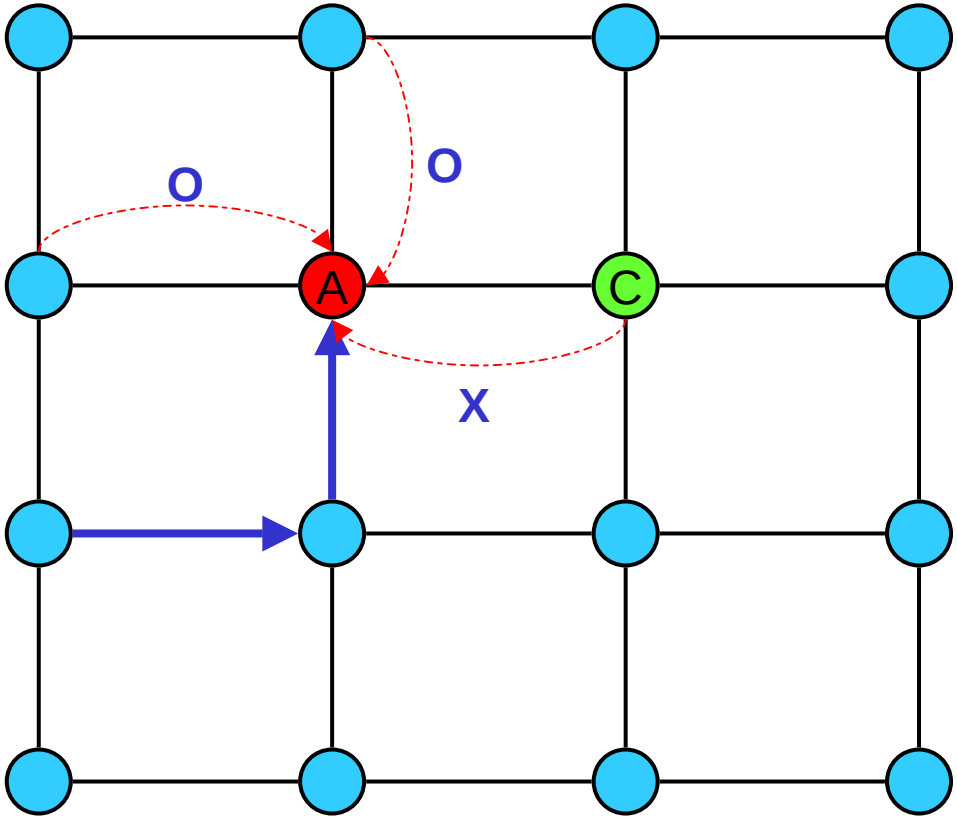


Look-Ahead


Score$_A$ = 2
O
O
A
X
X
B
X
O
Score$_B$ = 1


free_slots_in
X
Channel reserved
O
Channel available
NoP_data_out

Look-Ahead Problem



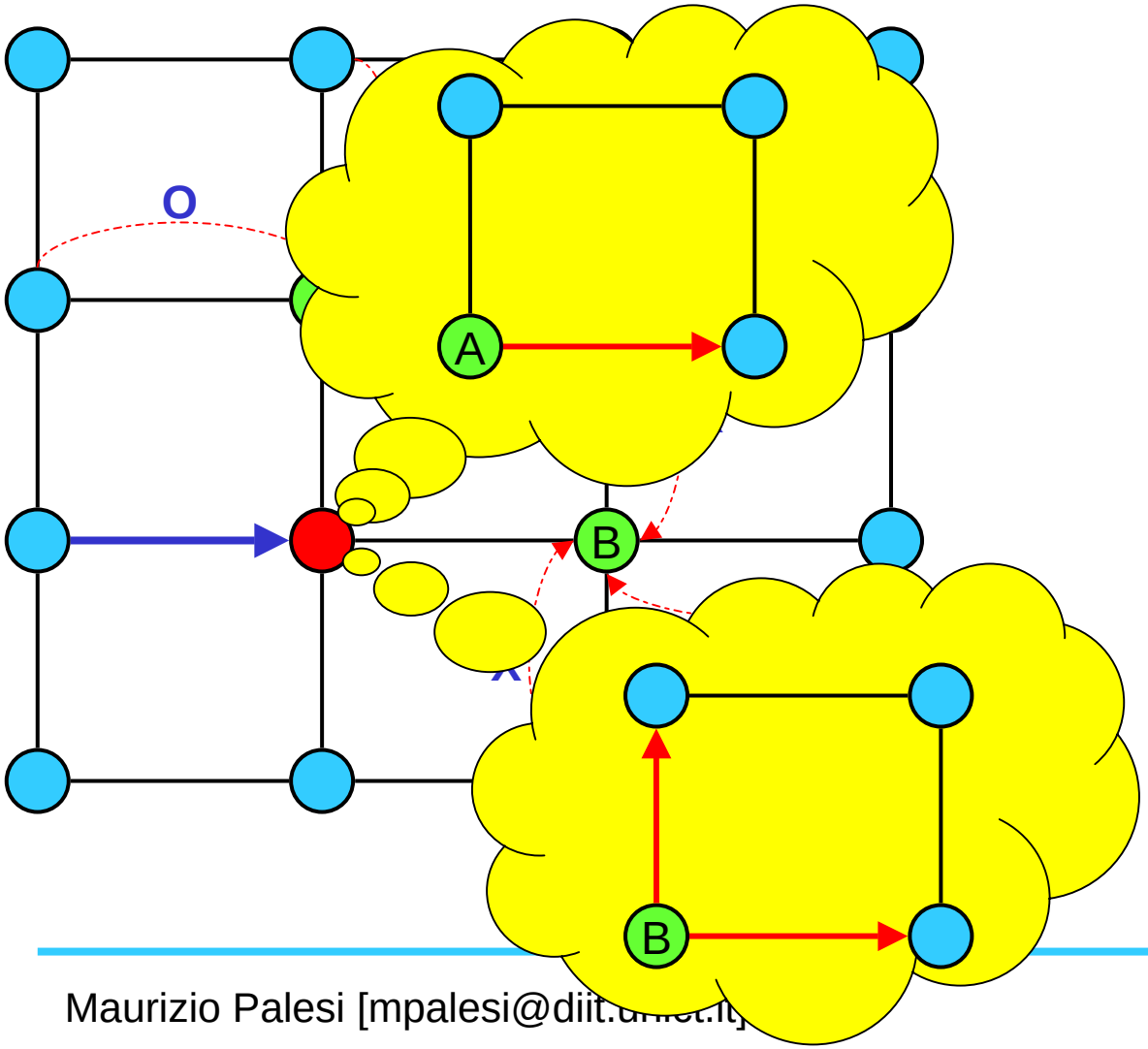
free_slots_in

X Channel reserved

O Channel available

NoP_data_out

NoP Strategy



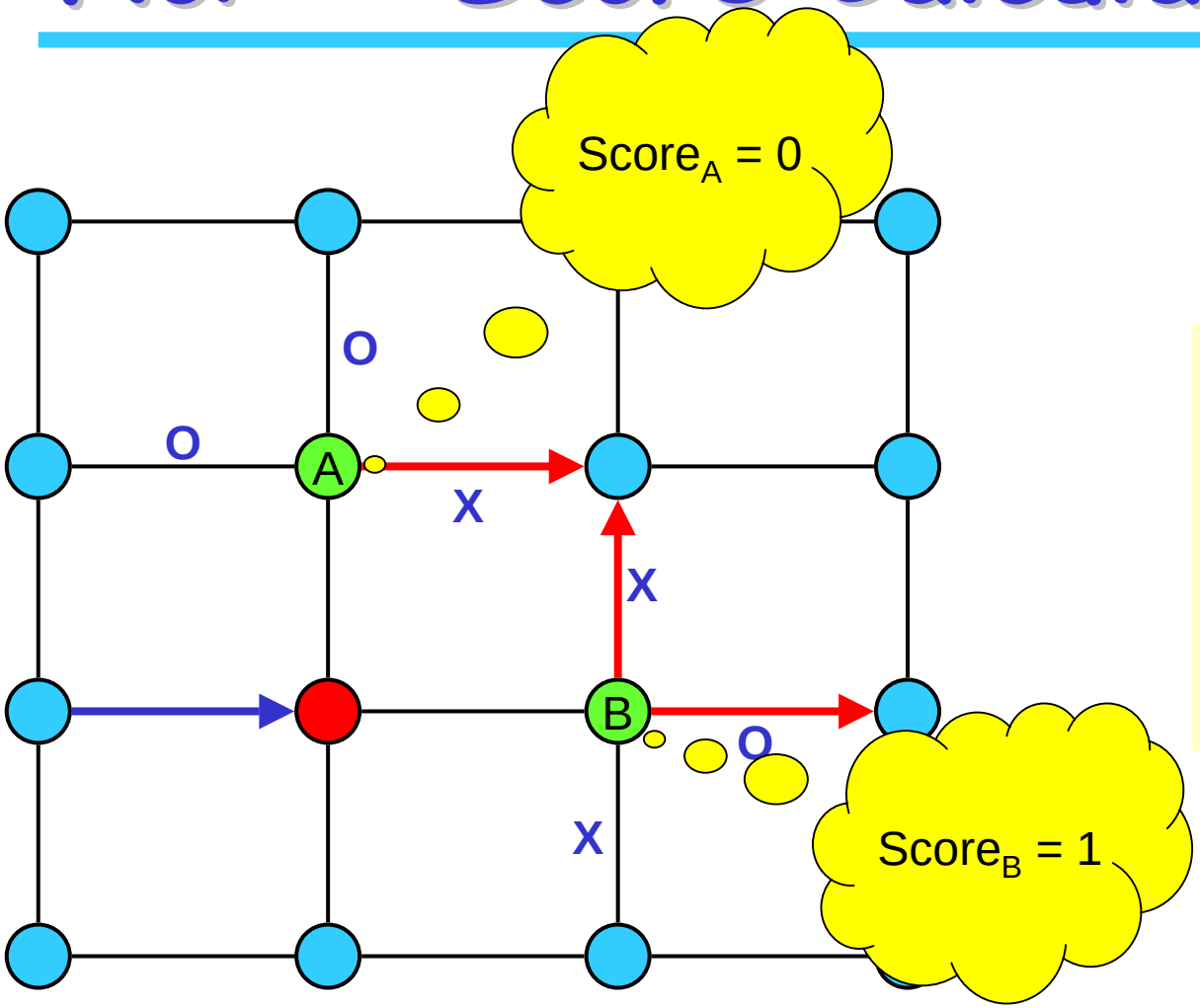
free_slots_in

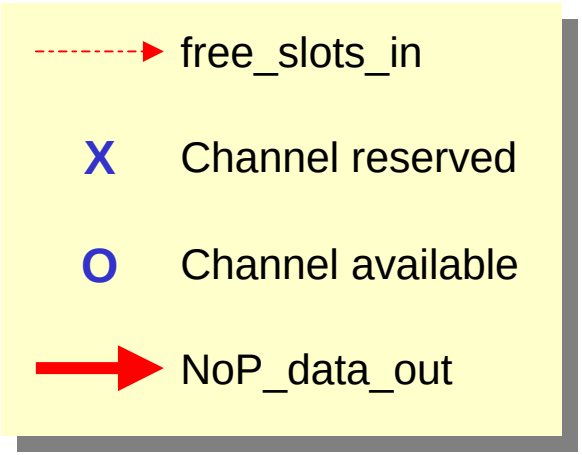
X Channel reserved

O Channel available

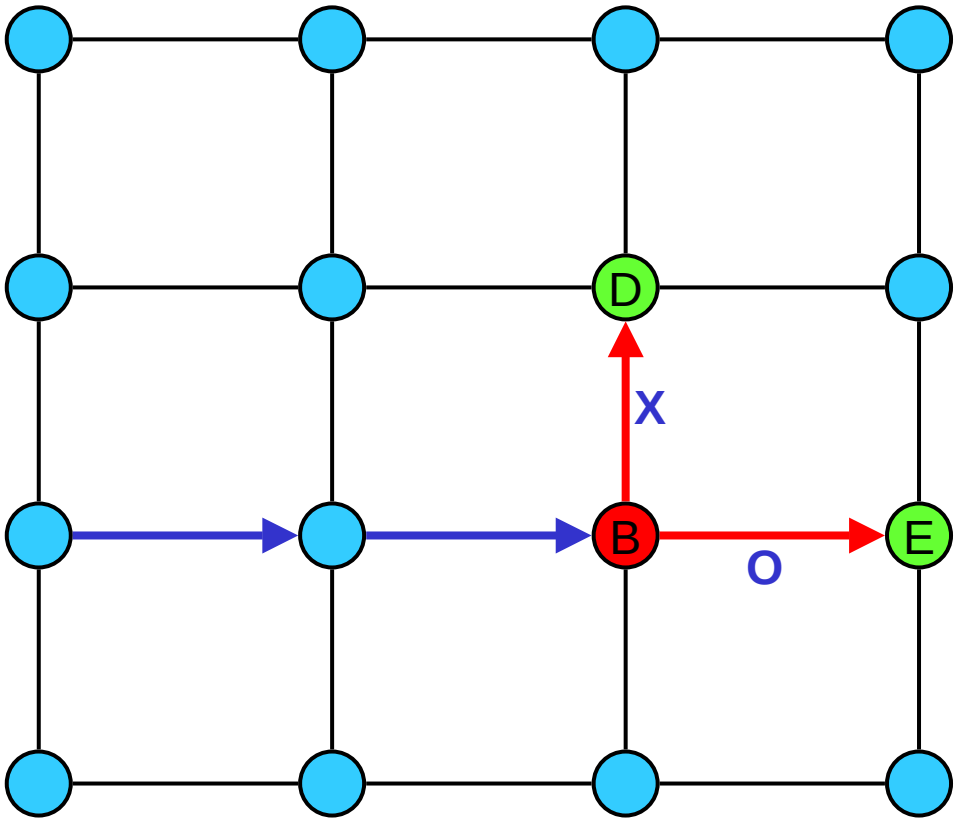
NoP_data_out

NoP - Score Calculation





NoP Strategy



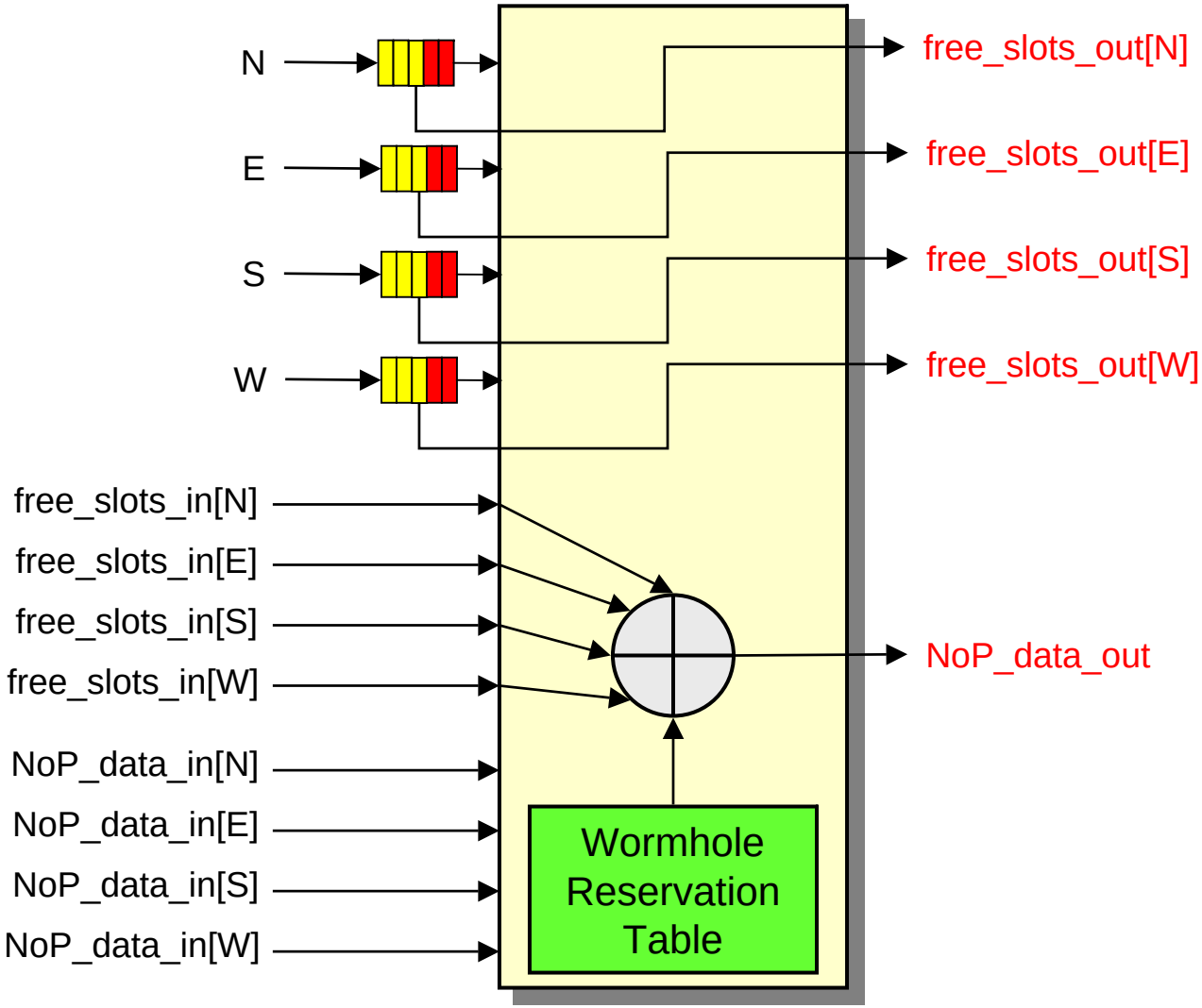
free_slots_in

X Channel reserved

O Channel available

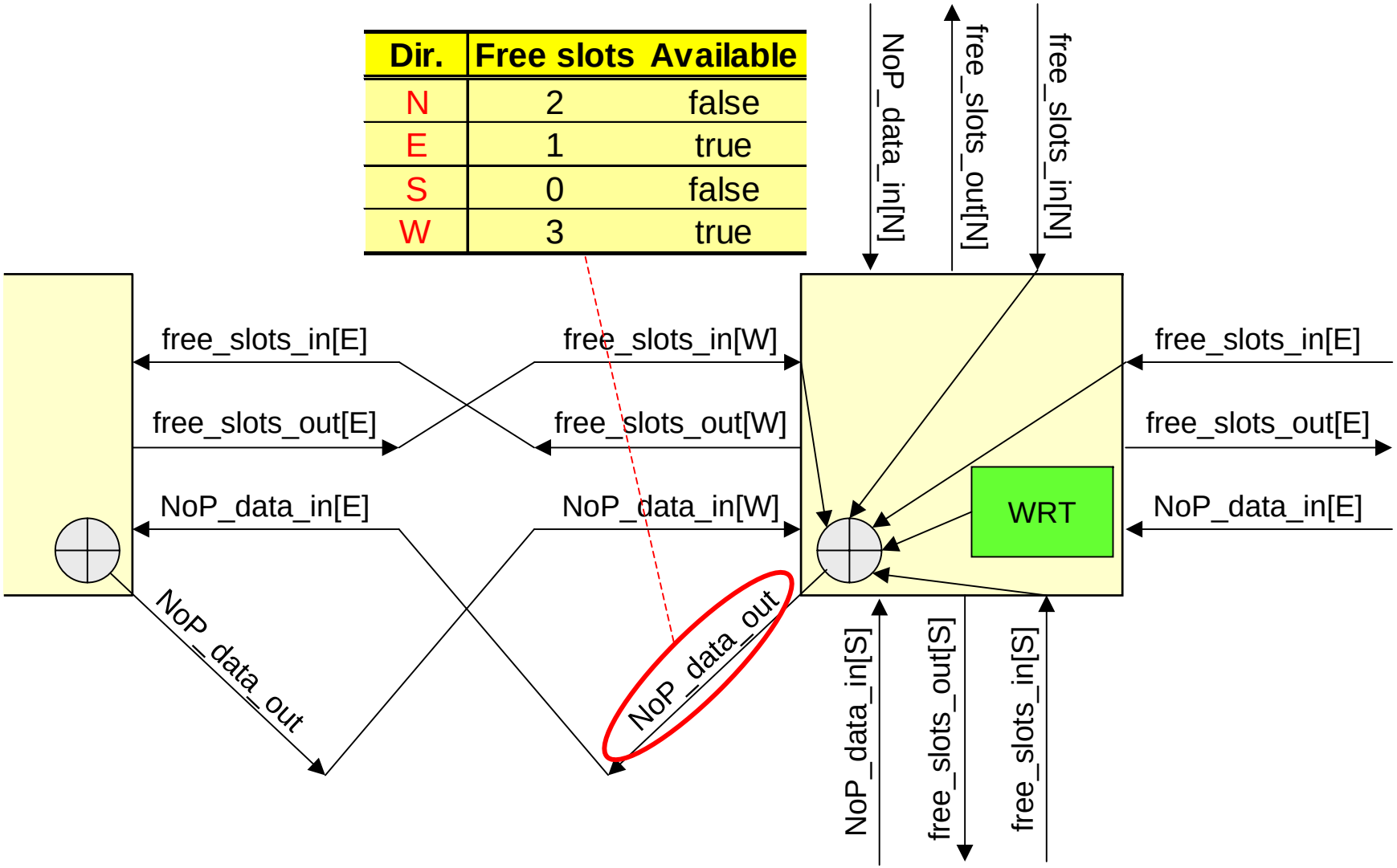
NoP_data_out

NoP Signals


N
E
S
W
free_slots_out[N]
free_slots_out[E]
free_slots_out[S]
free_slots_out[W]
free_slots_in[N]
free_slots_in[E]
free_slots_in[S]
free_slots_in[W]
NoP_data_out
NoP_data_in[N]
NoP_data_in[E]
NoP_data_in[S]
NoP_data_in[W]
Wormhole Reservation Table

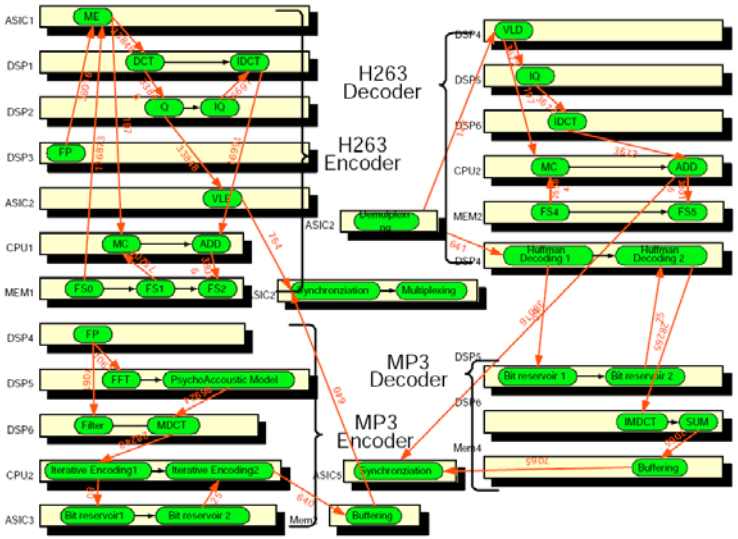
NoP Interfacing

Dir.	Free slots	Available
N	2	false
E	1	true
S	0	false
W	3	true



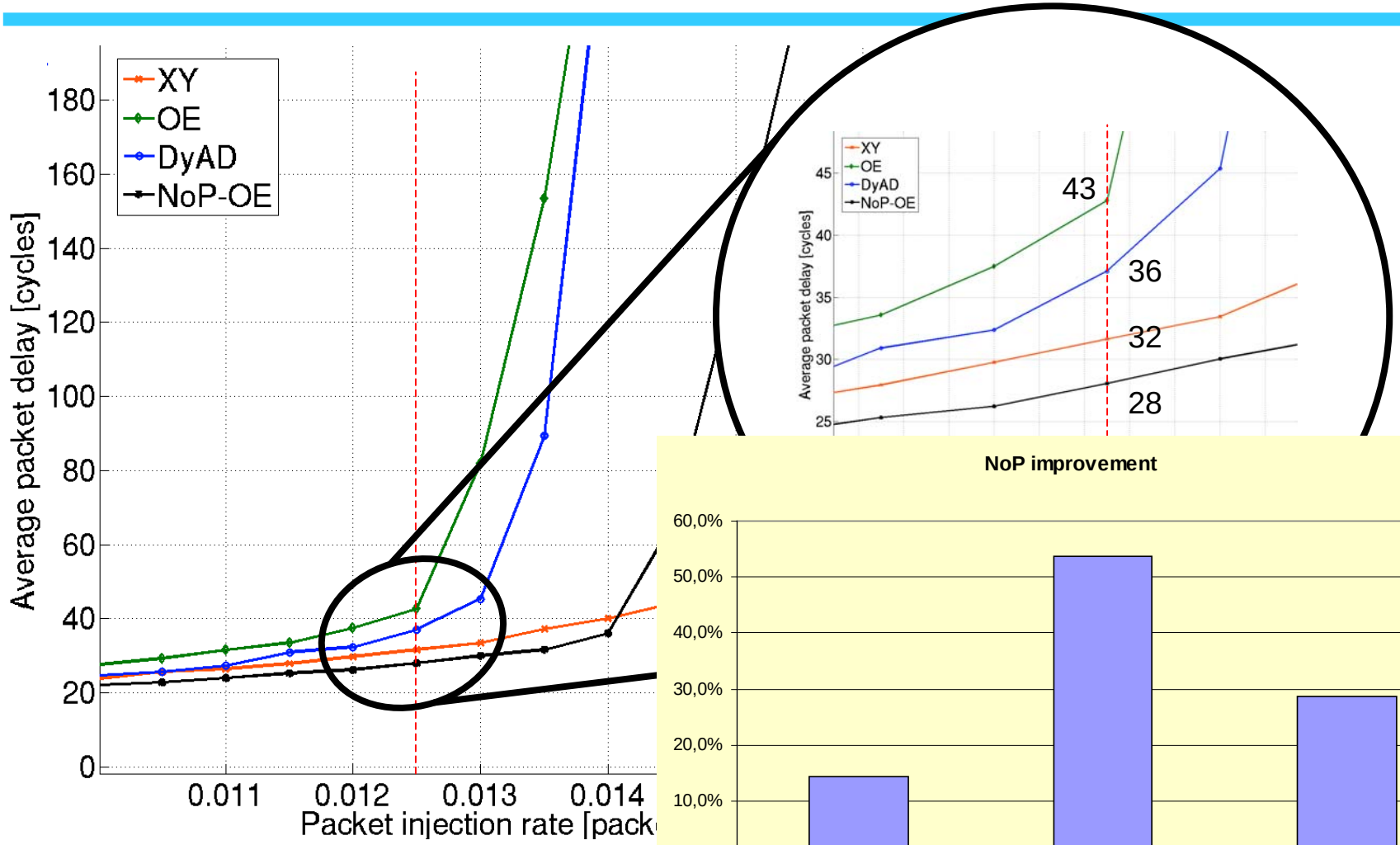
Experiments

- Synthetic and real traffic scenarios
  - Random
  - Transposed
  - Hot-spot
  - Multimedia application
- NoC modeling
  - SystemC simulator, flit-accurate
  - 8x8 mesh-based NoC
    - ✓ FIFO size: 4 flits
    - ✓ Values averaged over 60,000 packet arrivals (warm-up of 10,000 packets)
    - ✓ 95% confidence interval mostly within 2% of the mean



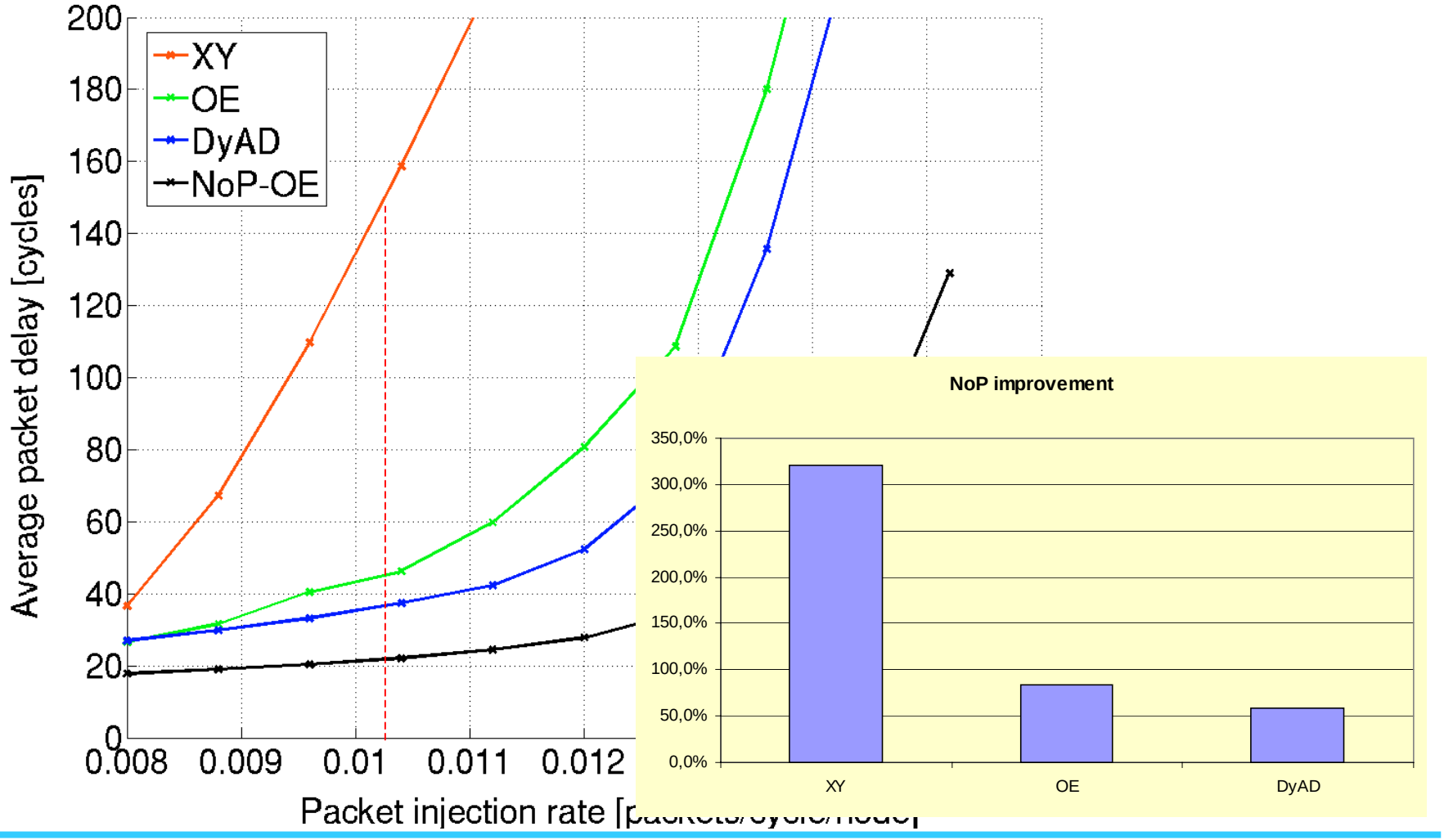
[Hu and Marculescu, TCAD05]

Uniform

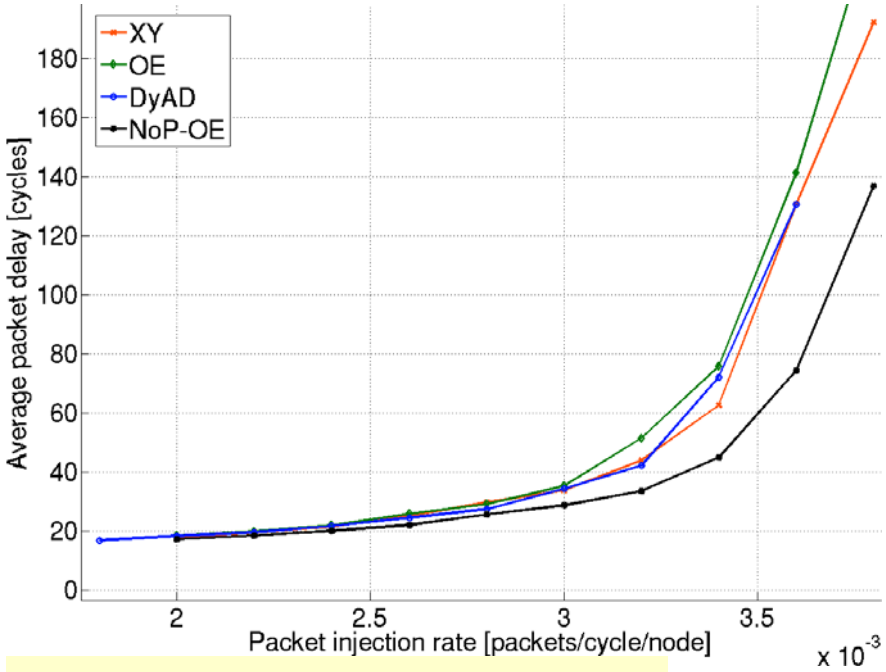

XY
OE
DyAD
NoP-OE
Average packet delay [cycles]
180
160
140
120
100
80
60
40
20
0
0.011
0.012
0.013
0.014
Packet injection rate [packe
43
36
32
28
25
30
35
40
45

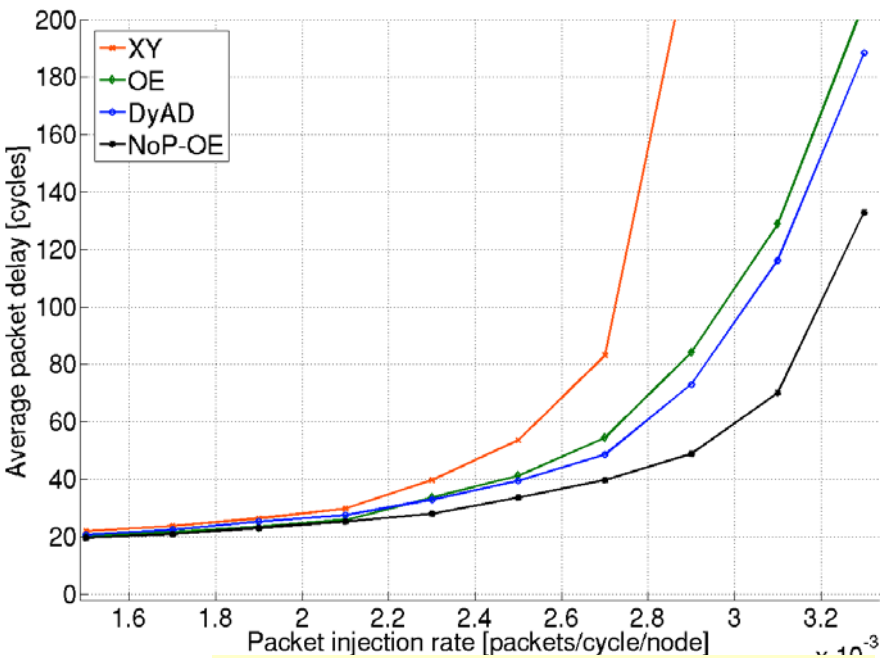
NoP improvement
60,0%
50,0%
40,0%
30,0%
20,0%
10,0%
0,0%
XY
OE
DyAD

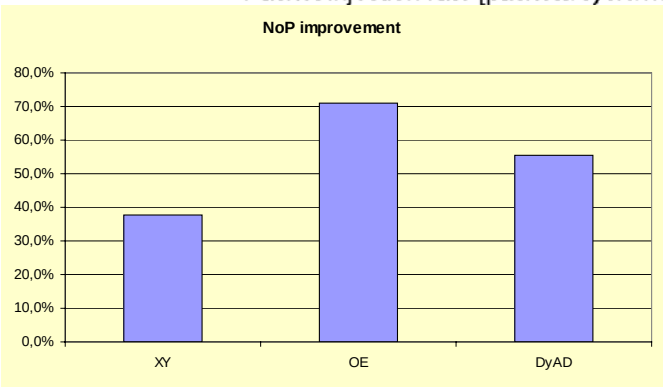
Transposed


XY
OE
DyAD
NoP-OE
Average packet delay [cycles]
200
180
160
140
120
100
80
60
40
20
0
0.008
0.009
0.01
0.011
0.012
Packet injection rate [
NoP improvement
350,0%
300,0%
250,0%
200,0%
150,0%
100,0%
50,0%
0,0%
XY
OE
DyAD

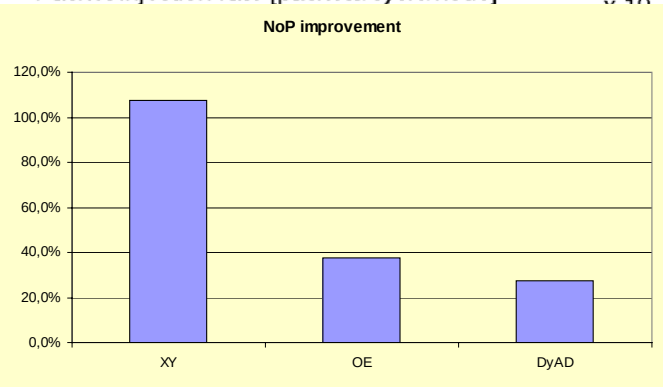
Hot-Spot



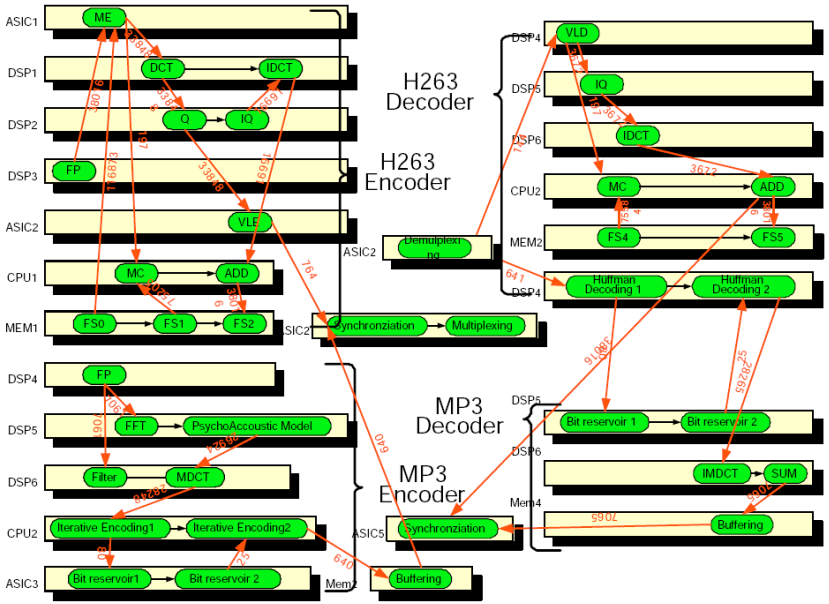


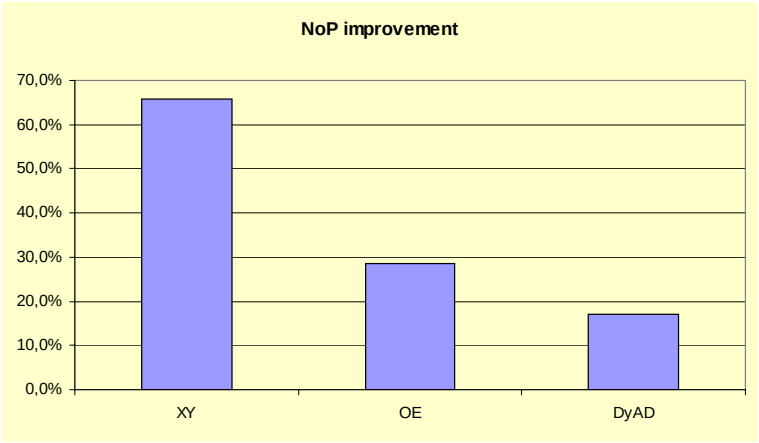


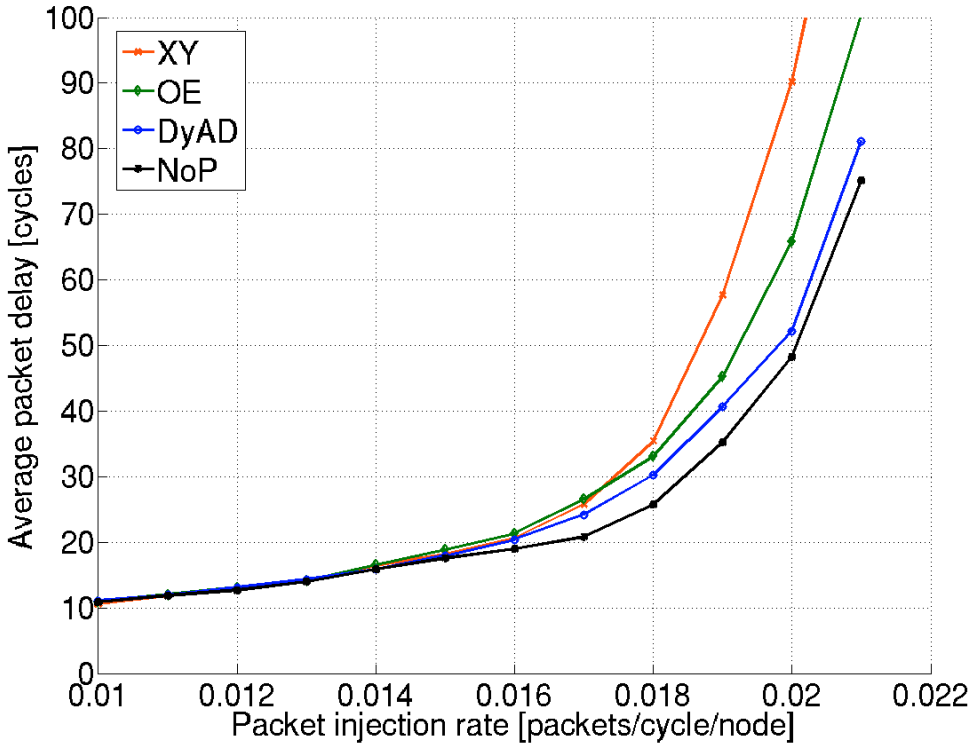
hs percentage 20%



MMS


ASIC1
DSP1
DSP2
DSP3
ASIC2
CPU1
MEM1
DSP4
DSP5
DSP6
CPU2
ASIC3
ME
DCT
IDCT
Q
IQ
FP
VLE
MC
ADD
FS0
FS1
FS2
FFT
PsychoAccoustic Model
Filter
MDCT
Iterative Encoding1
Iterative Encoding2
Bit reservoir1
Bit reservoir 2
H263 Decoder
H263 Encoder
MP3 Decoder
MP3 Encoder
Demultiplexing
Synchronization
Multiplexing
VLD
IQ
IDCT
MC
ADD
FS4
FS5
Huffman Decoding 1
Huffman Decoding 2
Bit reservoir 1
Bit reservoir 2
IMDCT
SUM
Synchronization
Buffering

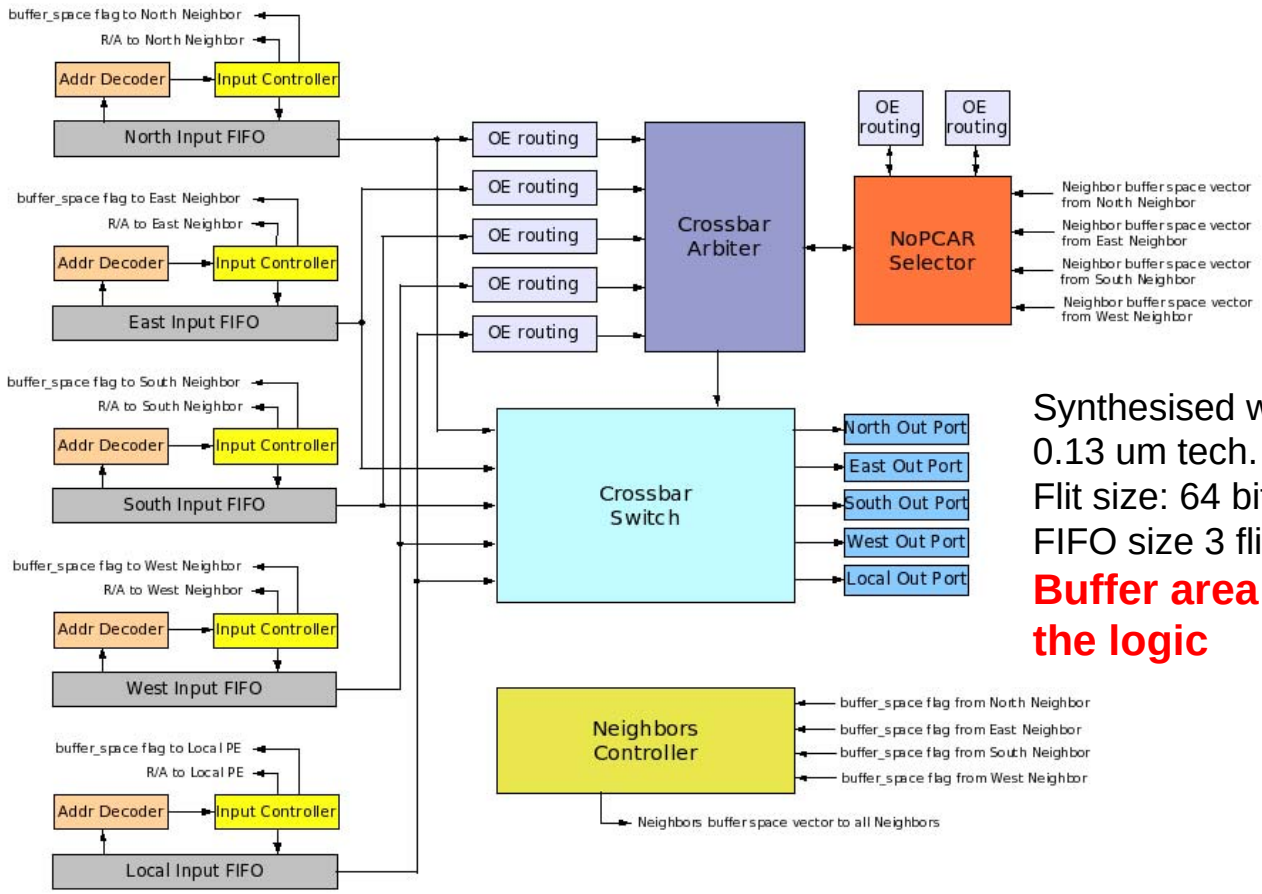

NoP improvement
70,0%
60,0%
50,0%
40,0%
30,0%
20,0%
10,0%
0,0%
XY
OE
DyAD


Average packet delay [cycles]
Packet injection rate [packets/cycle/node]
XY
OE
DyAD
NoP
0
10
20
30
40
50
60
70
80
90
100
0.01
0.012
0.014
0.016
0.018
0.02
0.022

Average Performance Improvement

- Average delay improvement of NoP-OE over
  - XY **109%**
  - Odd-Even **55%**
  - DyAD **37%**

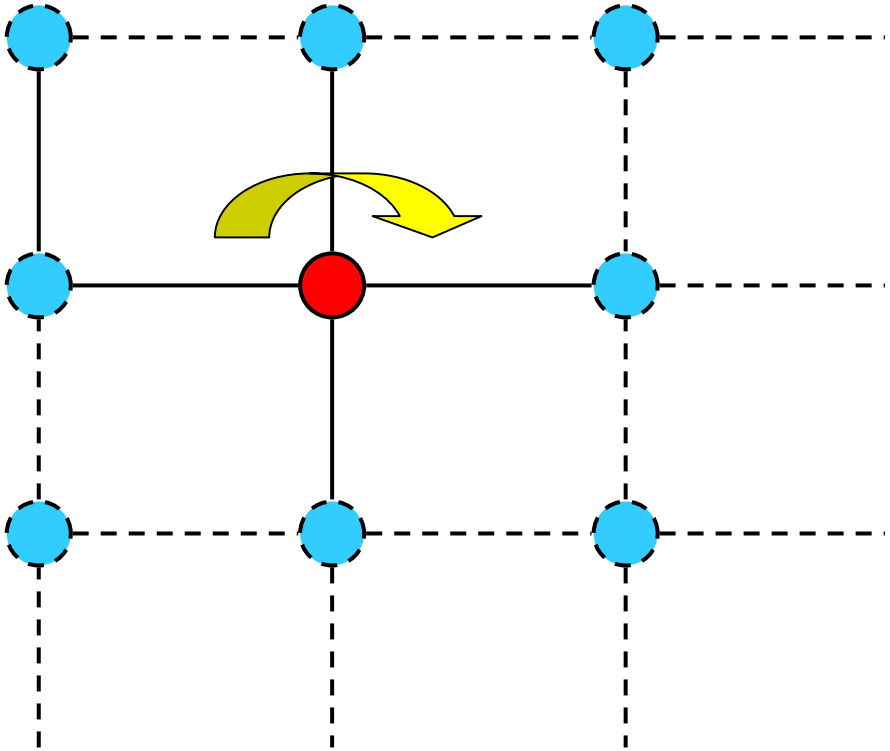
Block Diagram of the Router



Synthesised with Synopsys Design Compiler
0.13 um tech. Library from Virtual Silicon
Flit size: 64 bit
FIFO size 3 flits
Buffer area significantly dominates the logic

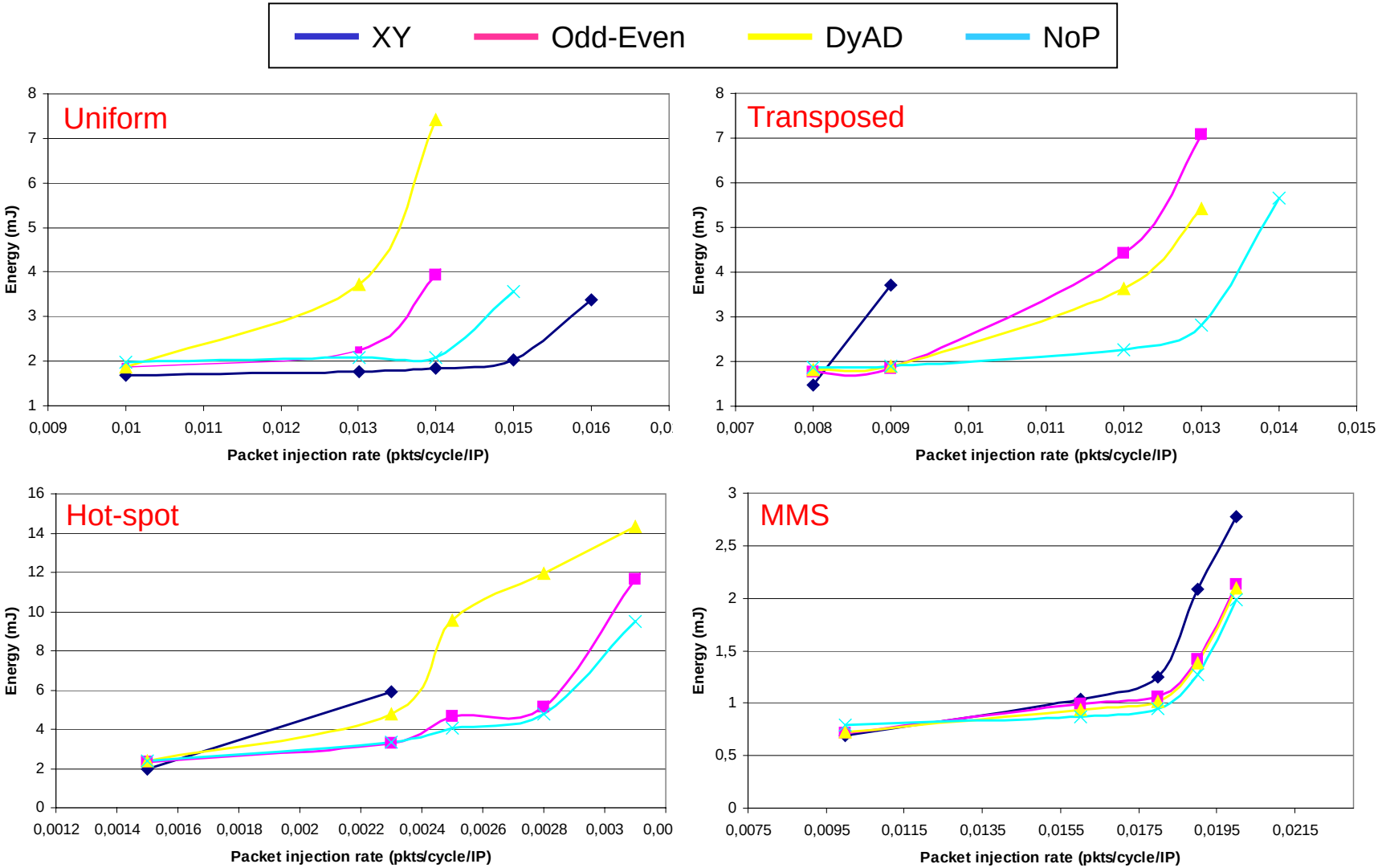
Area overhead due to NoP is less then 9%

Power Analysis



	Energy (nJ)		
	Head	Body	Idle
XY	0,151	0,135	0,013
Odd-Even	0,178	0,152	0,016
DyAD	0,182	0,161	0,019
NoP-OE	0,189	0,168	0,023

Energy


XY
Odd-Even
DyAD
NoP
Uniform
Energy (mJ)
Packet injection rate (pkts/cycle/IP)
Transposed
Energy (mJ)
Packet injection rate (pkts/cycle/IP)
Hot-spot
Energy (mJ)
Packet injection rate (pkts/cycle/IP)
MMS
Energy (mJ)
Packet injection rate (pkts/cycle/IP)

Summary

- NoP Selection Policy
  - ➔ Can be coupled with any routing function
  - ➔ Performance improvement (55% on average)
  - ➔ Low area overhead (<10%)
  - ➔ Energy efficient
- Future work
  - ➔ Application of NoP to highly adaptive routing algorithms (APSRA)