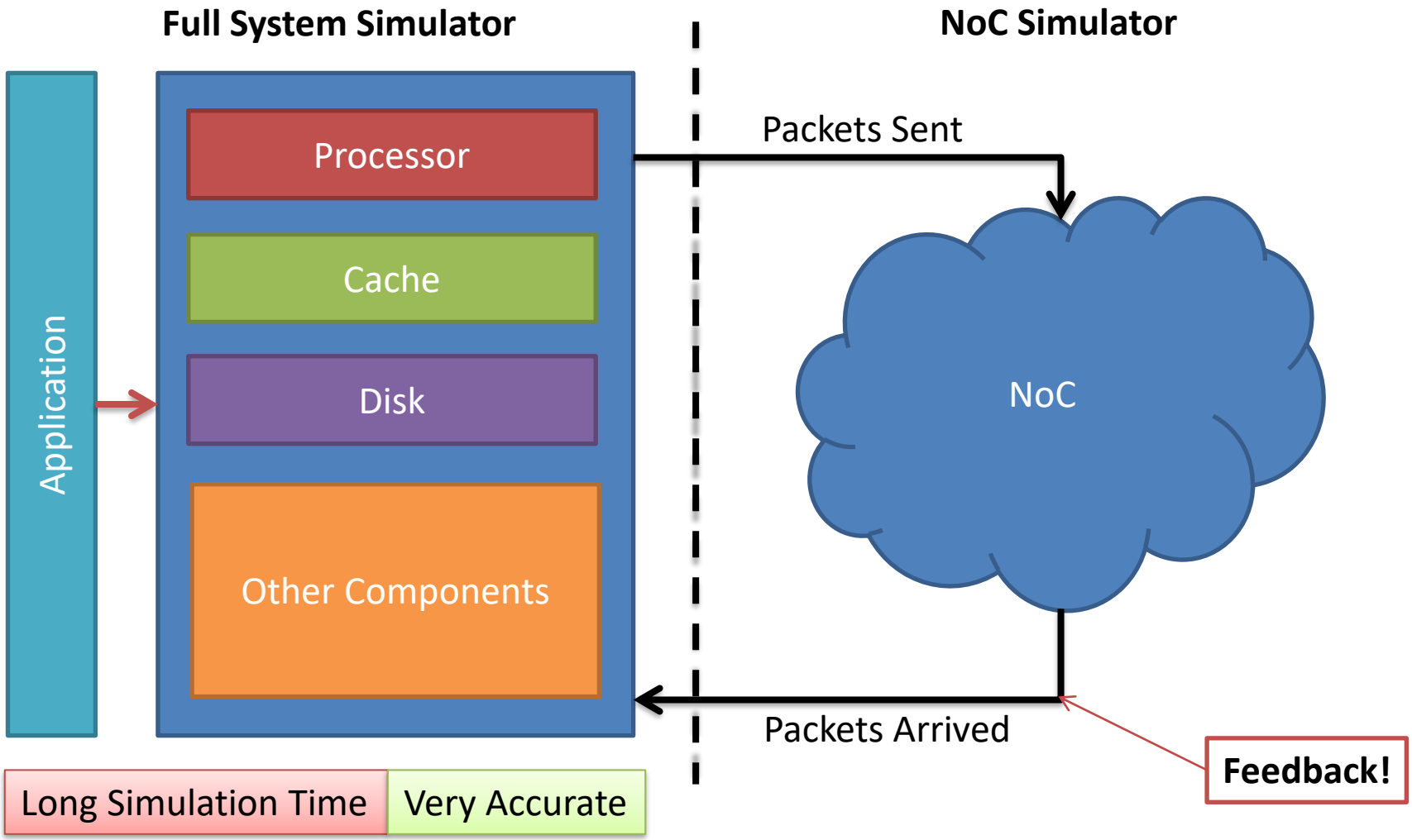
Cache Coherent Synthetic Traffic Models

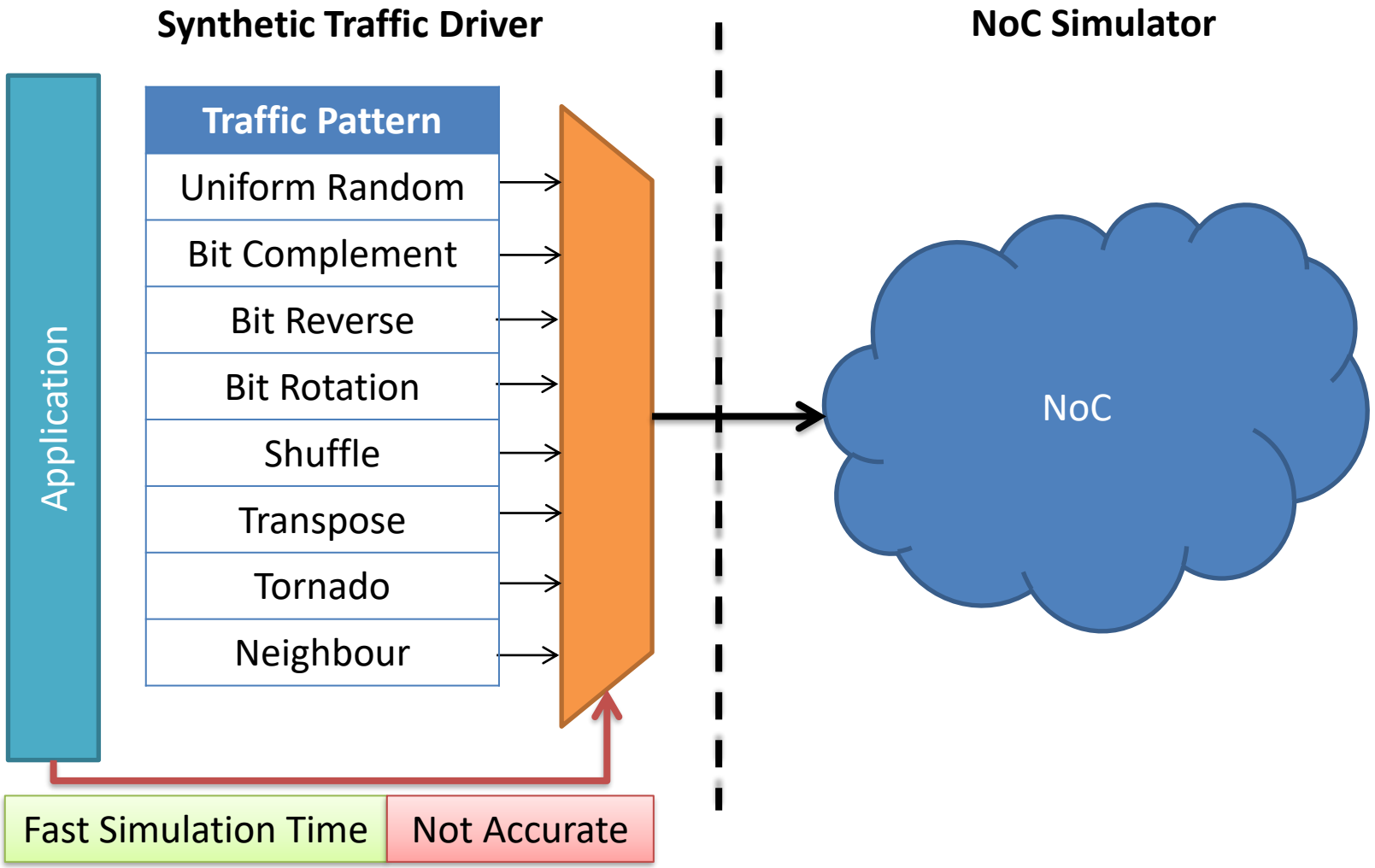
Mario Badr

Supervisor: Natalie Enright Jerger

Evaluating NoC Performance

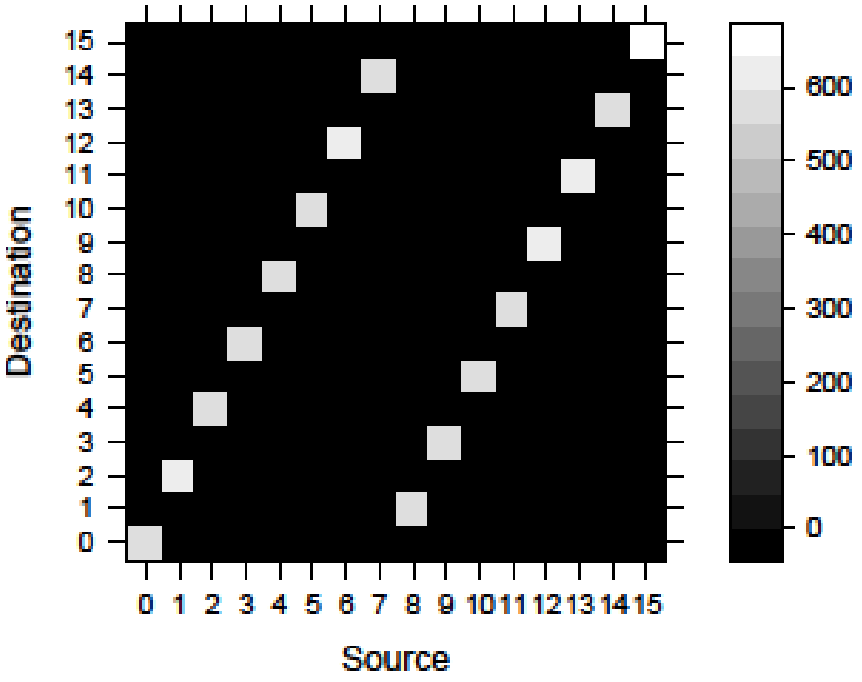


Evaluating NoC Performance



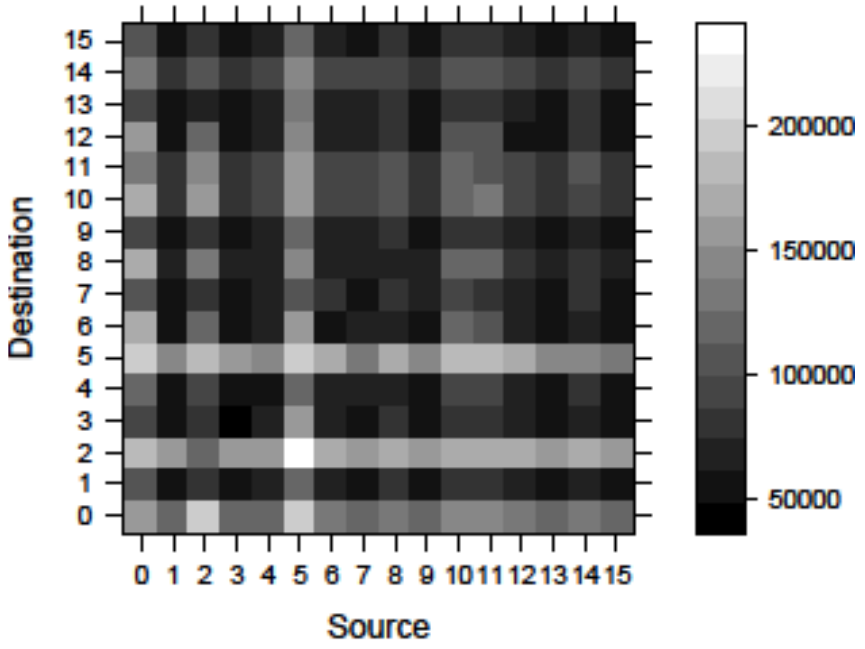
Spatial Behaviour of Traffic

Synthetic Traffic



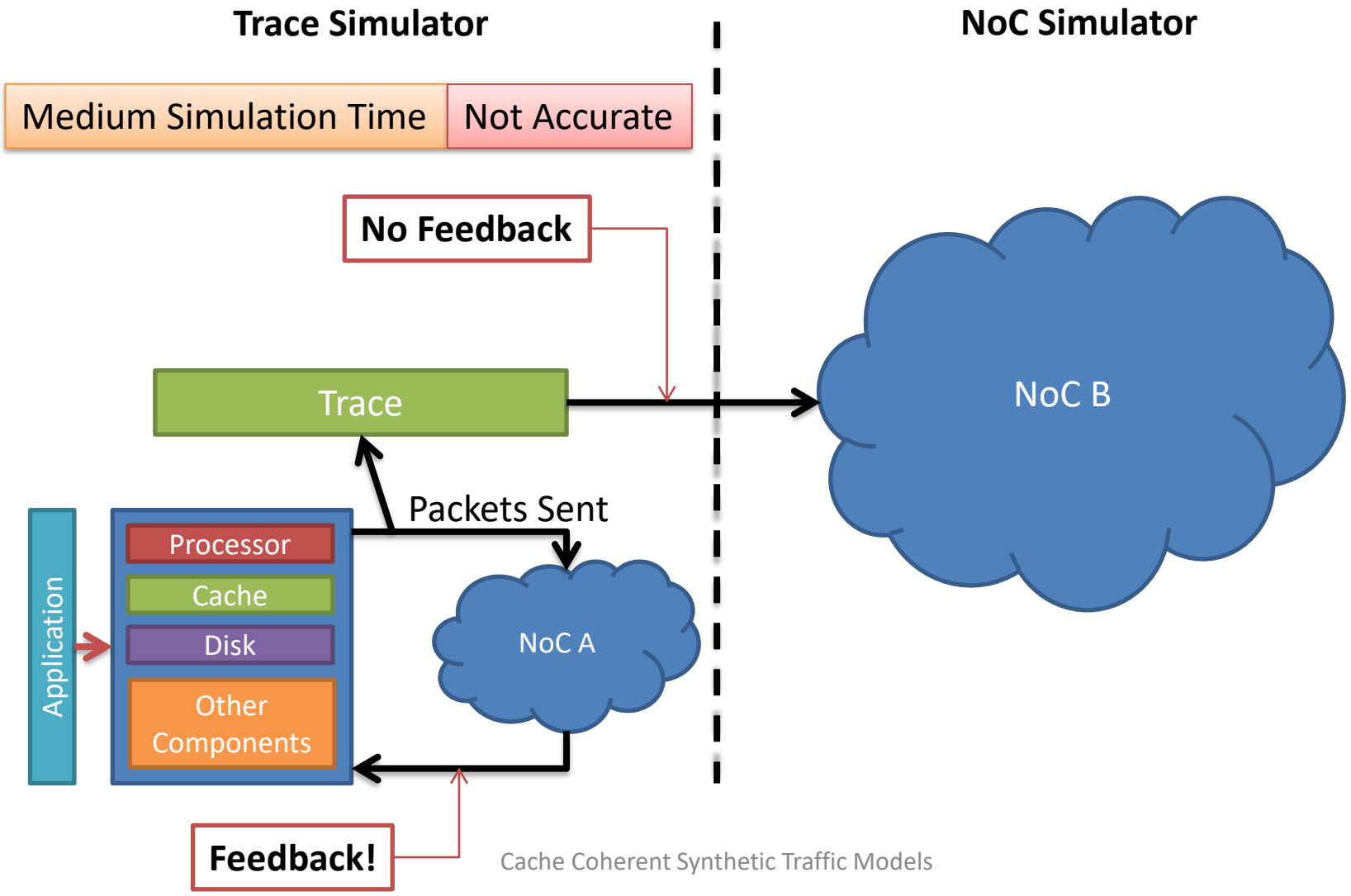
Shuffle Traffic Pattern

Full System Simulation

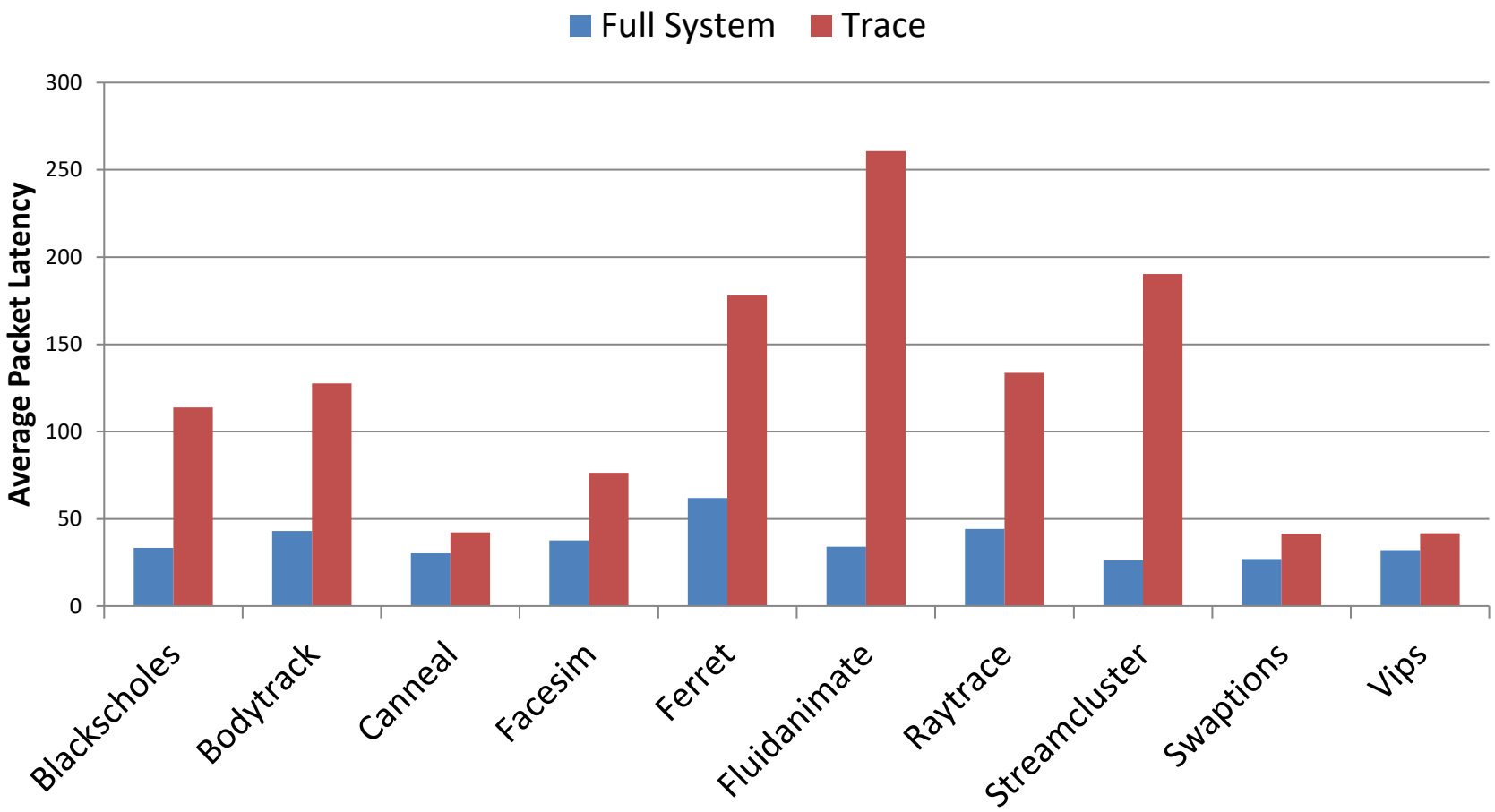


FFT Benchmark

Evaluating NoC Performance



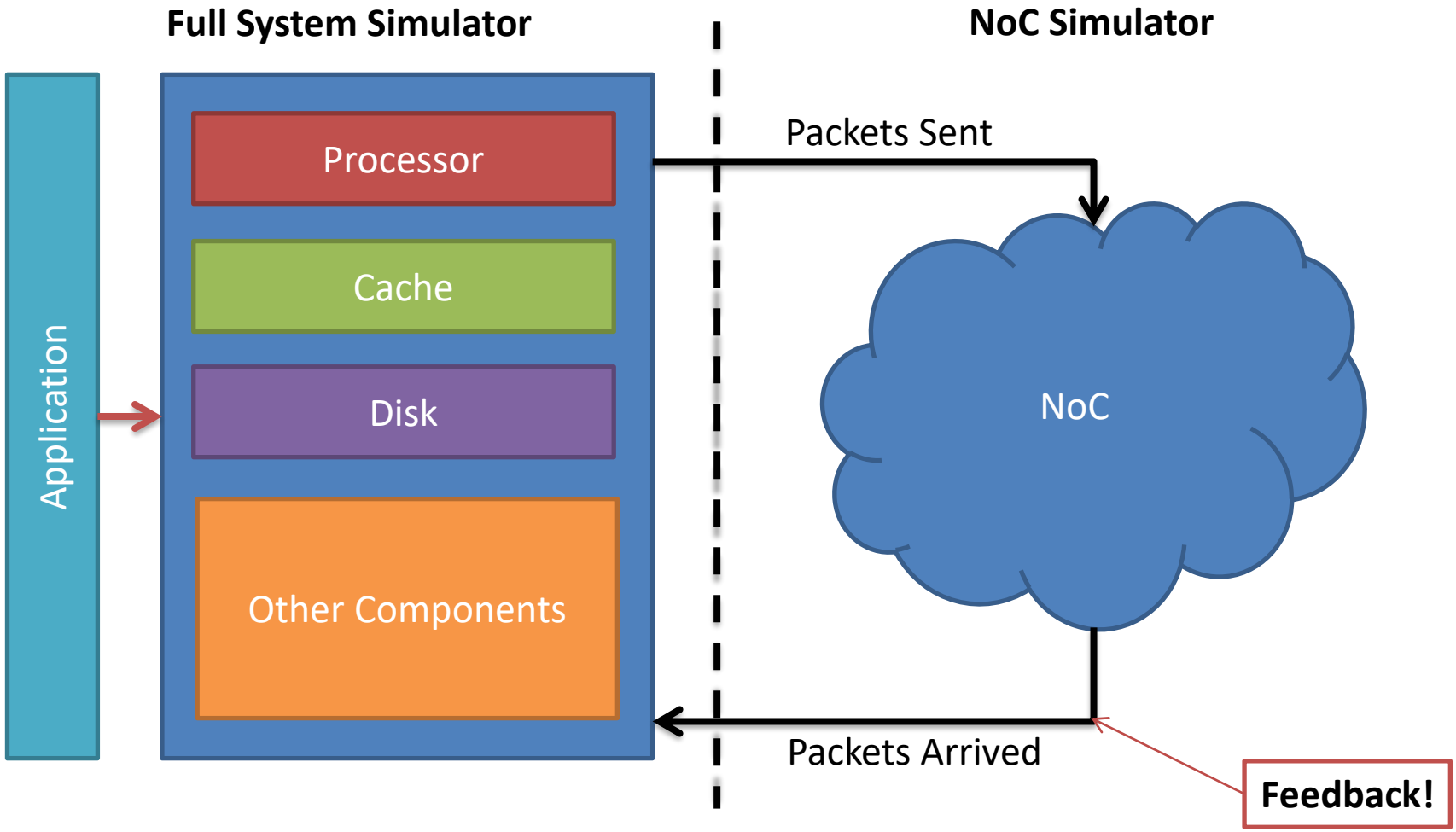
NoC Performance Comparison



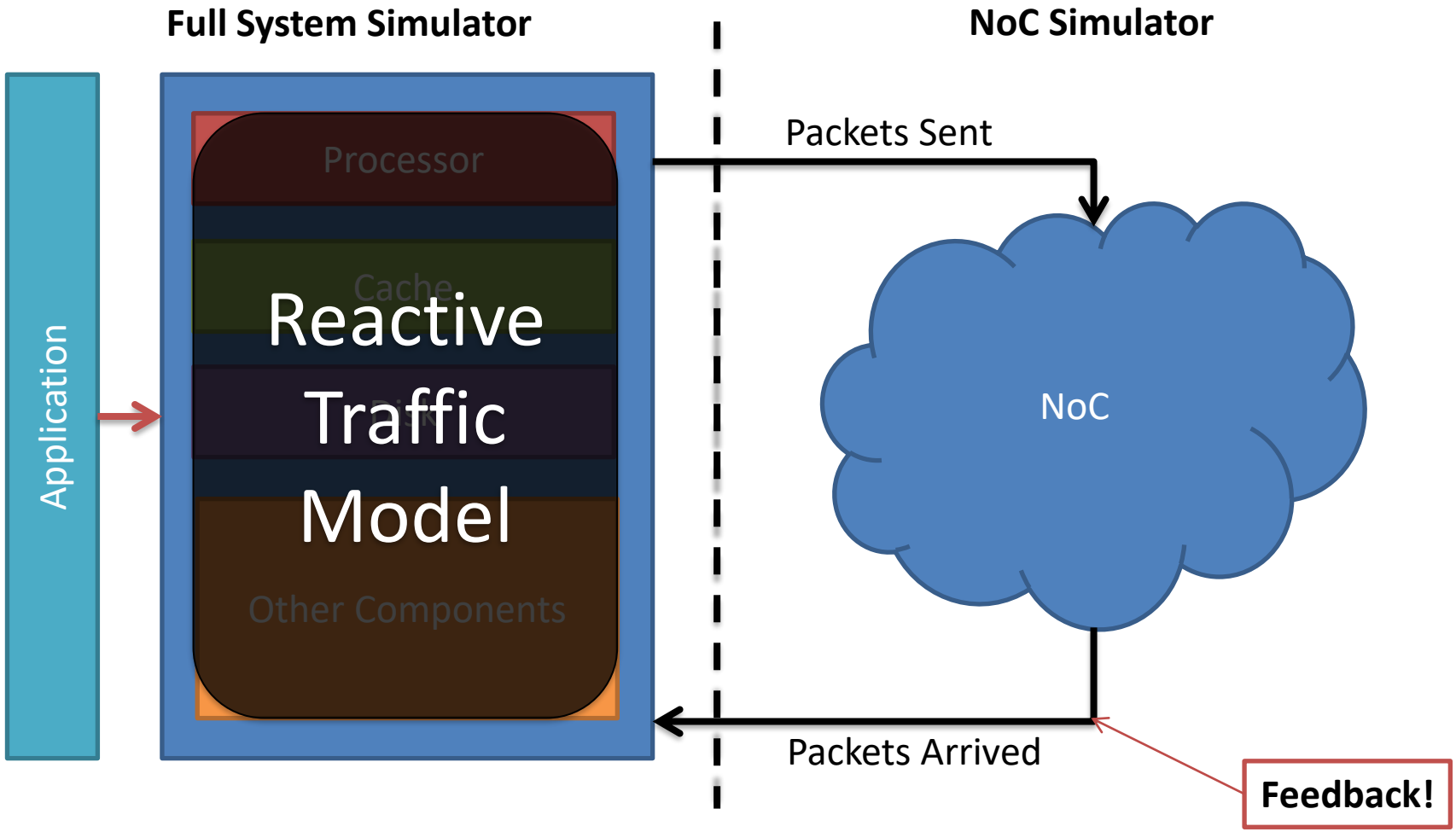
Simulation Methodology Review

- Full System Simulation
  - Model Each Component
  - Very Accurate
  - Long Simulation Time
- Synthetic Traffic
  - Traffic Patterns Based on Applications
  - Not Accurate
  - Very Short Simulation Time
- Trace Simulation
  - Most Temporal & Spatial Behaviour Captured
  - Short Simulation Time
  - Not Accurate (lack of feedback)

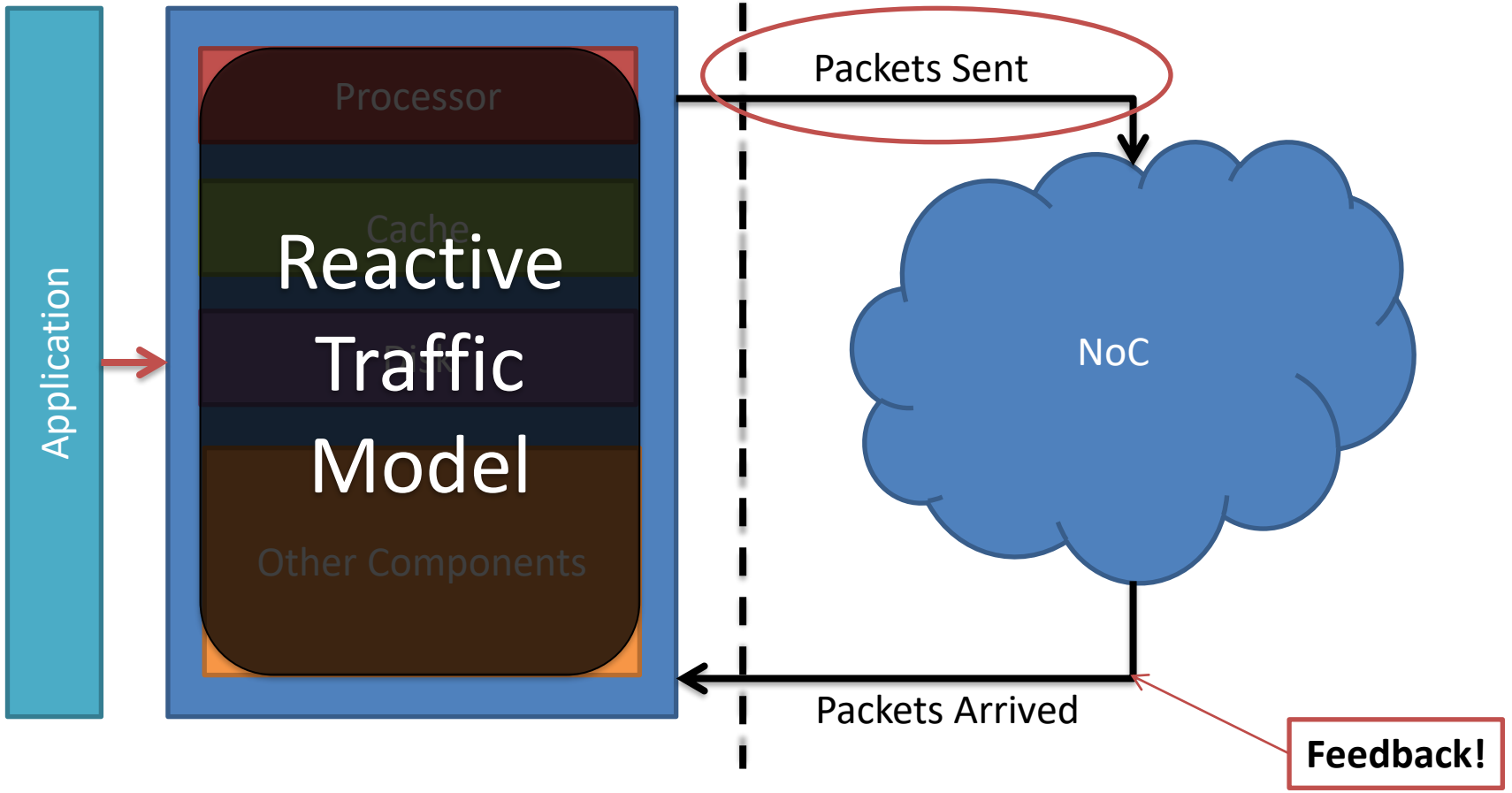
Overview


Full System Simulator
NoC Simulator
Application
Processor
Cache
Disk
Other Components
Packets Sent
NoC
Packets Arrived
Feedback!

Overview

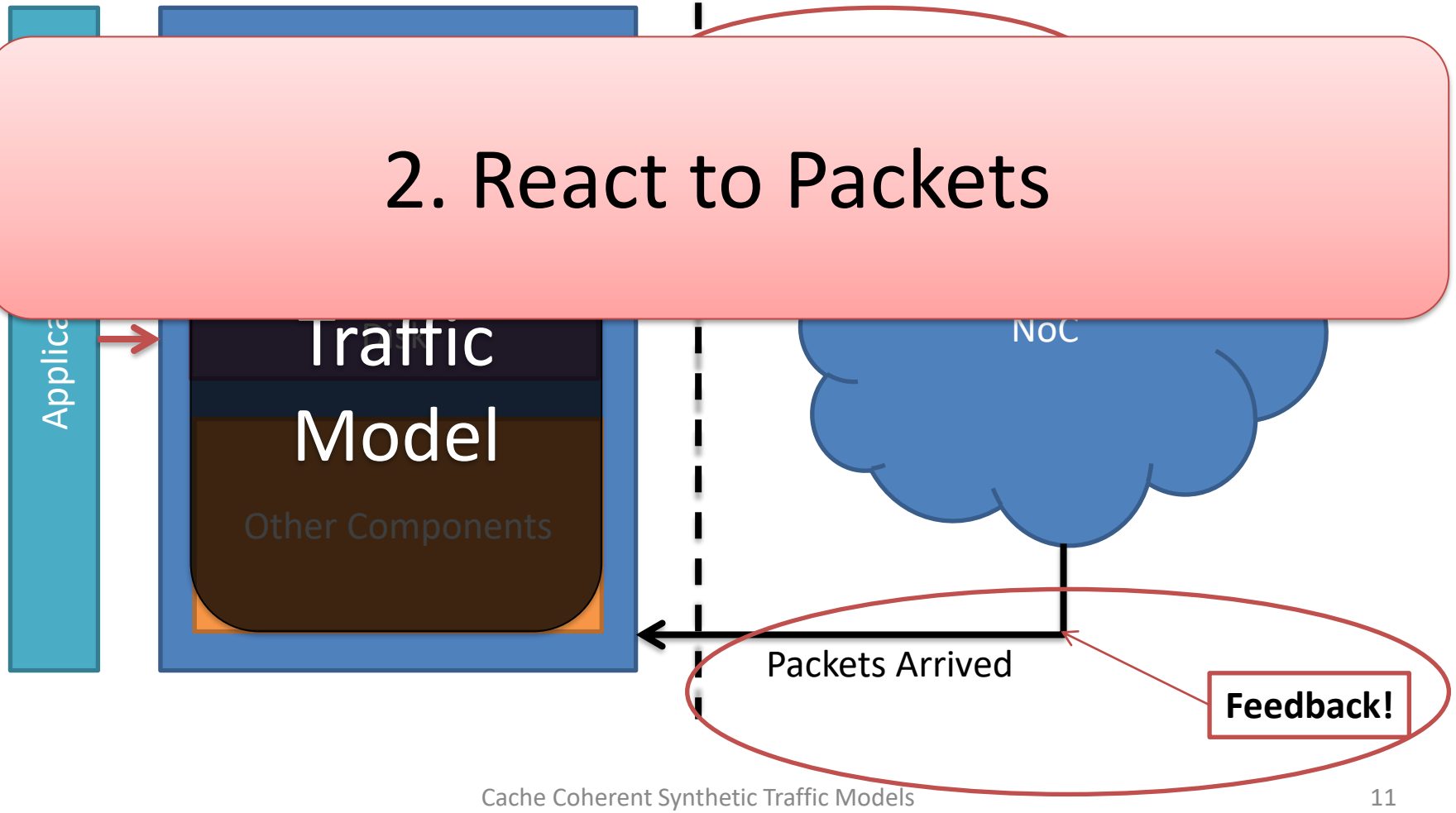

Full System Simulator
NoC Simulator
Application
Processor
Cache
Reactive
Traffic
Model
Other Components
Packets Sent
NoC
Packets Arrived
Feedback!

1. Send Packets

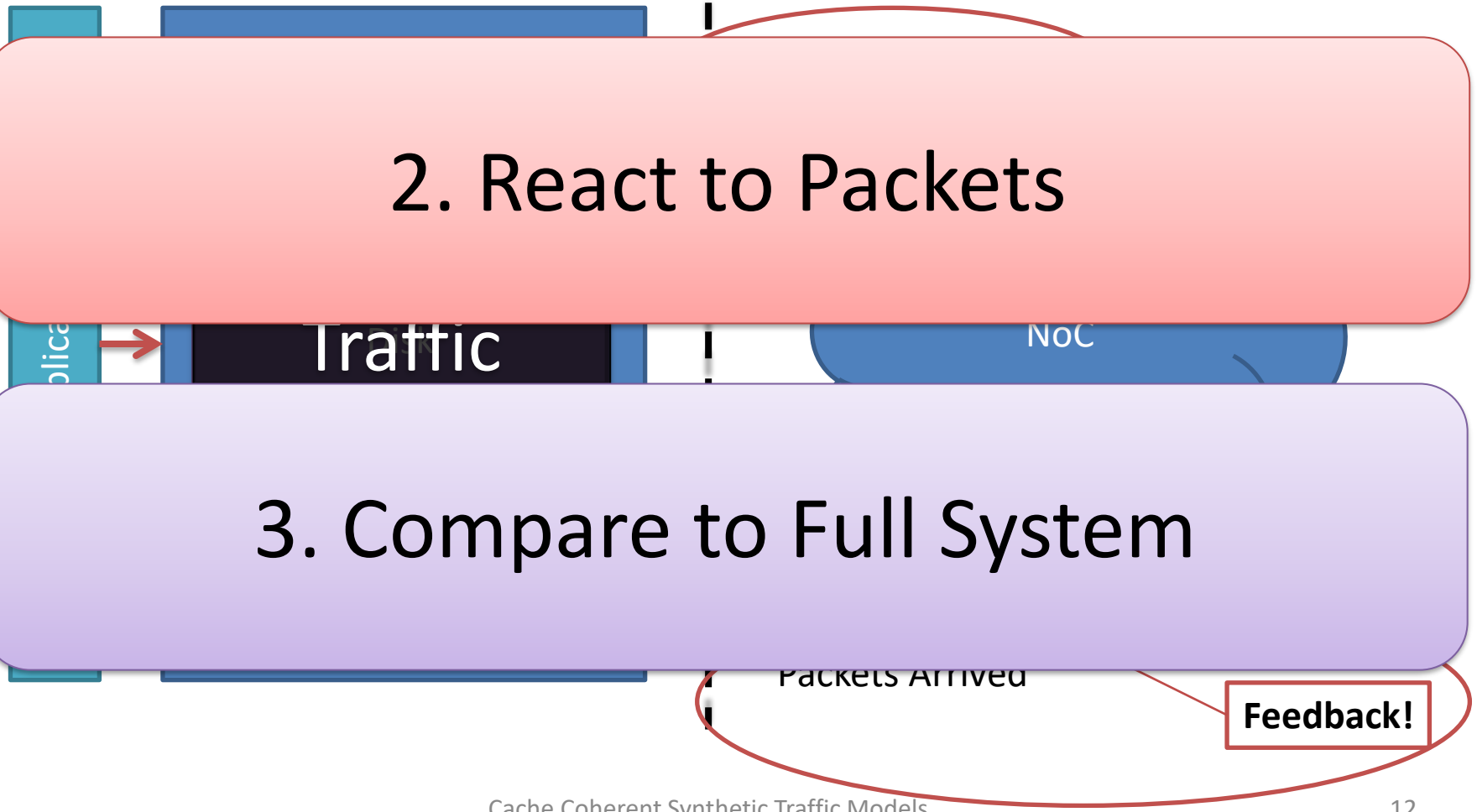


1. Send Packets

2. React to Packets



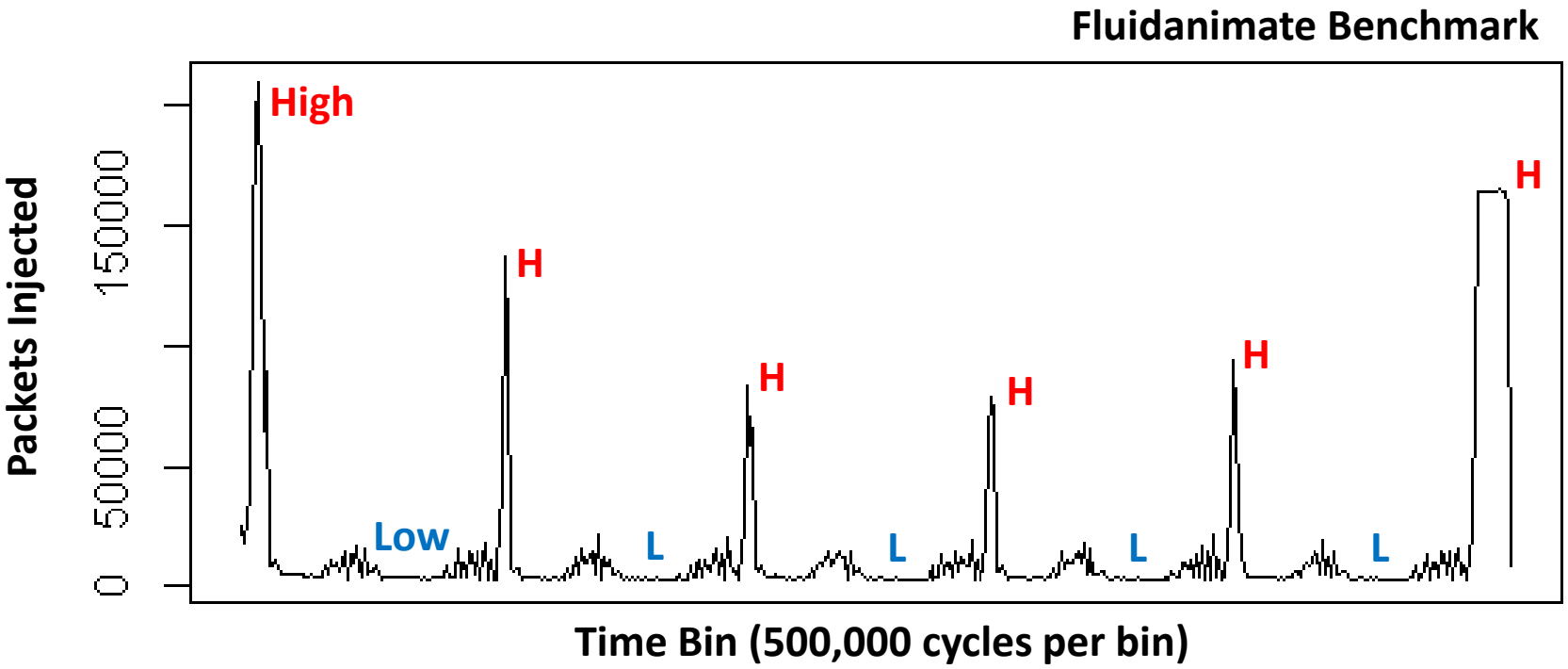
1. Send Packets



2. React to Packets

3. Compare to Full System

Sending Packets



Applications Have Time-Varying Behaviour; they go through ***phases***

Modelling Time-Varying Behaviour

Want: Send Packets with Time-Varying Behaviour

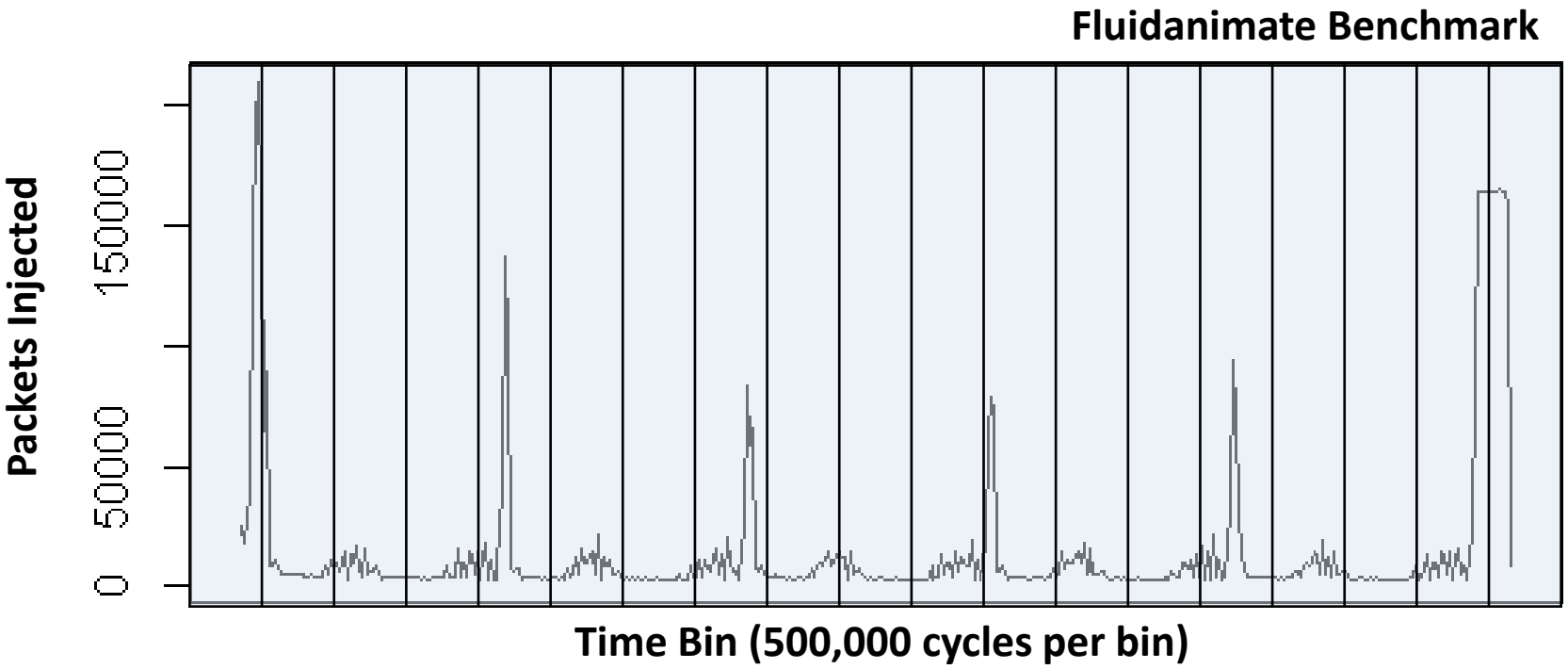
Need: Methodology to create and group phases

1. Divide traffic into ***Intervals***
2. Represent intervals with ***Feature Vectors***
3. Group feature vectors with ***Clustering***

Need: Methodology to transition from one phase to another

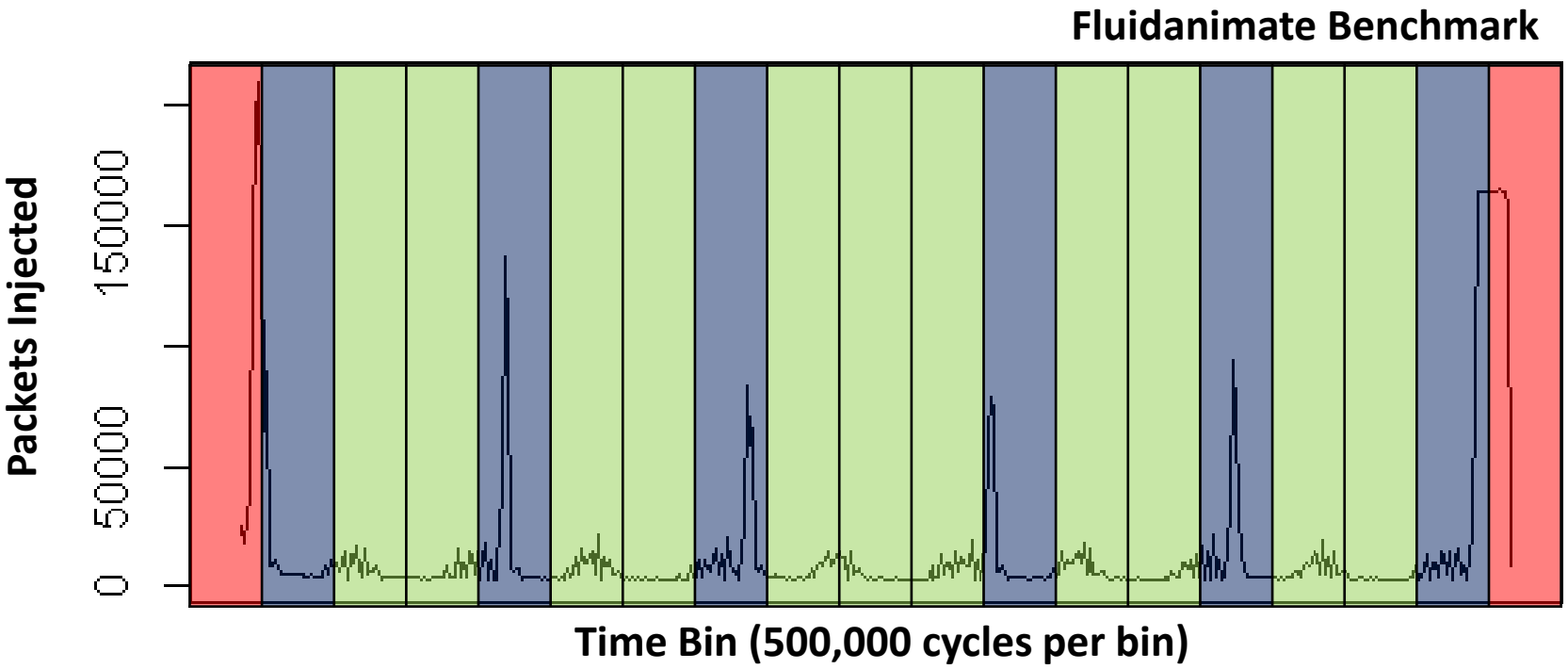
- Markov Chains

Dividing Into Intervals



Intervals are a fixed size.

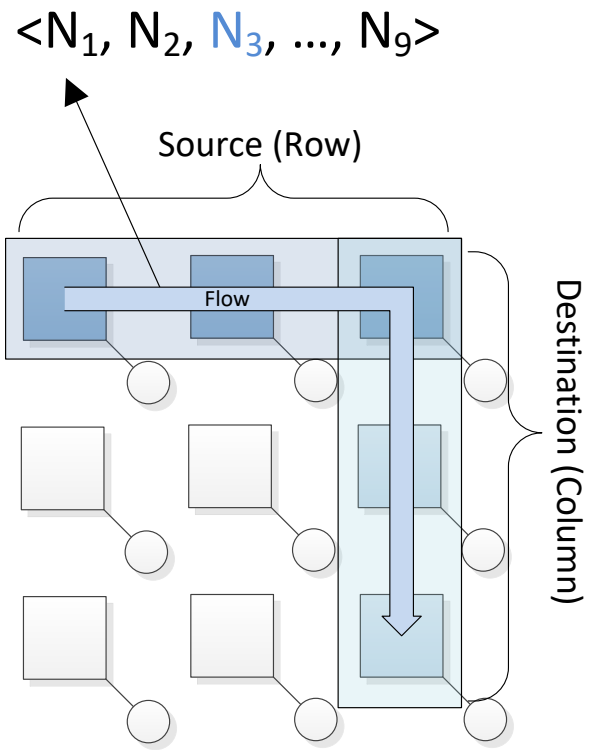
Dividing Into Intervals



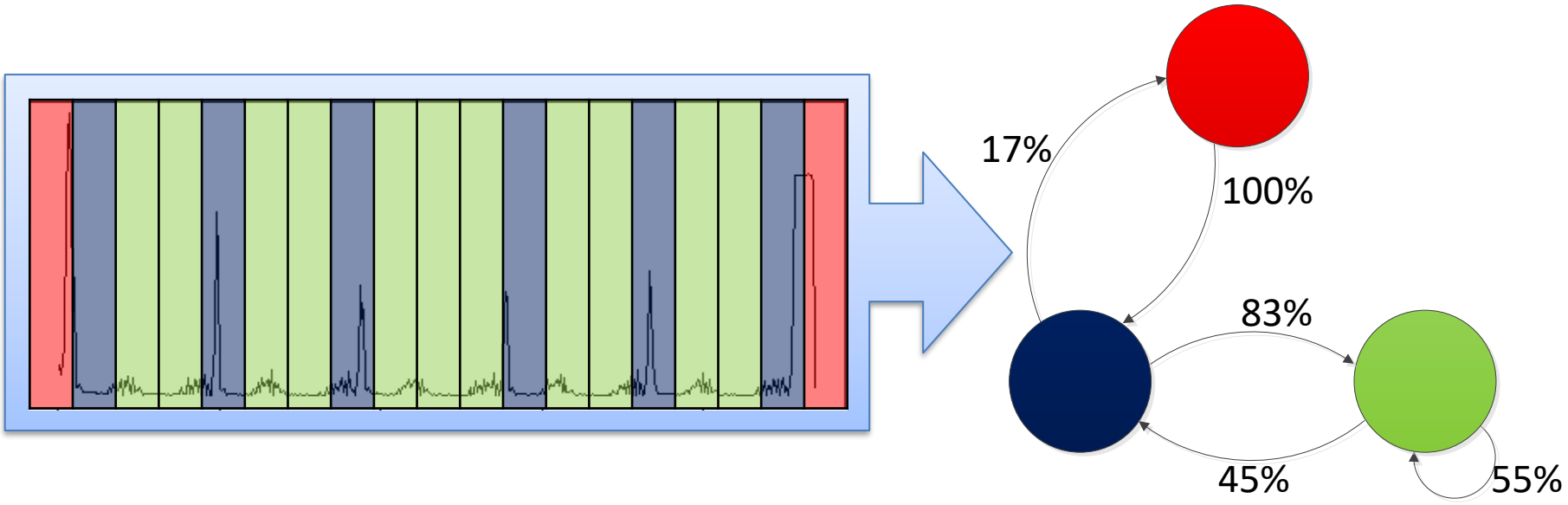
Visually we see: High, Low+High, and Low Intervals

Creating Phases

- Characterize intervals with vectors:
  1. Total Injection (TI)
  2. Coherence Composition
  3. Node Injection (NI)
  4. Row-Column Flow (RCFlow)
  5. Per-Node Flow (Flow)
- Group vectors with clustering algorithms



Phase Transitions



Given current state, there is some probability to transition to the next state.

Modelling Time-Varying Behaviour

Want: Send Packets with Time-Varying Behaviour

Need: Methodology to create and group phases

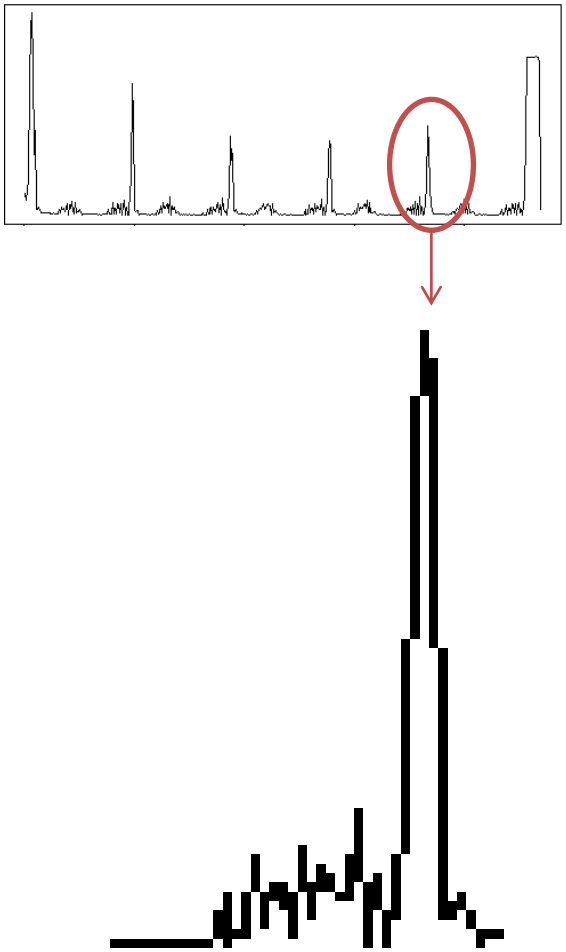
1. Divide traffic into ***Intervals***
2. Represent intervals with ***Feature Vectors***
3. Group feature vectors with ***Clustering***

Need: Methodology to transition from one phase to another

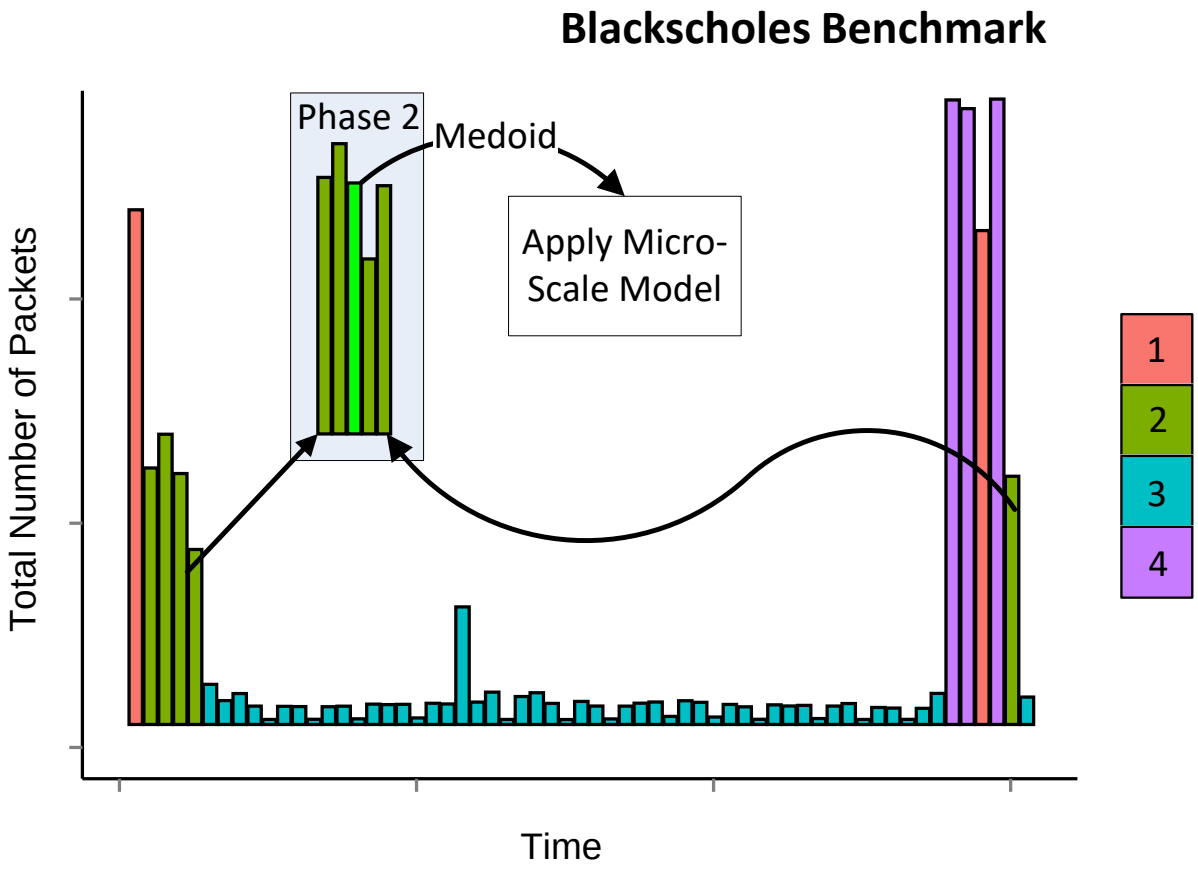
- Markov Chains

Inside An Interval

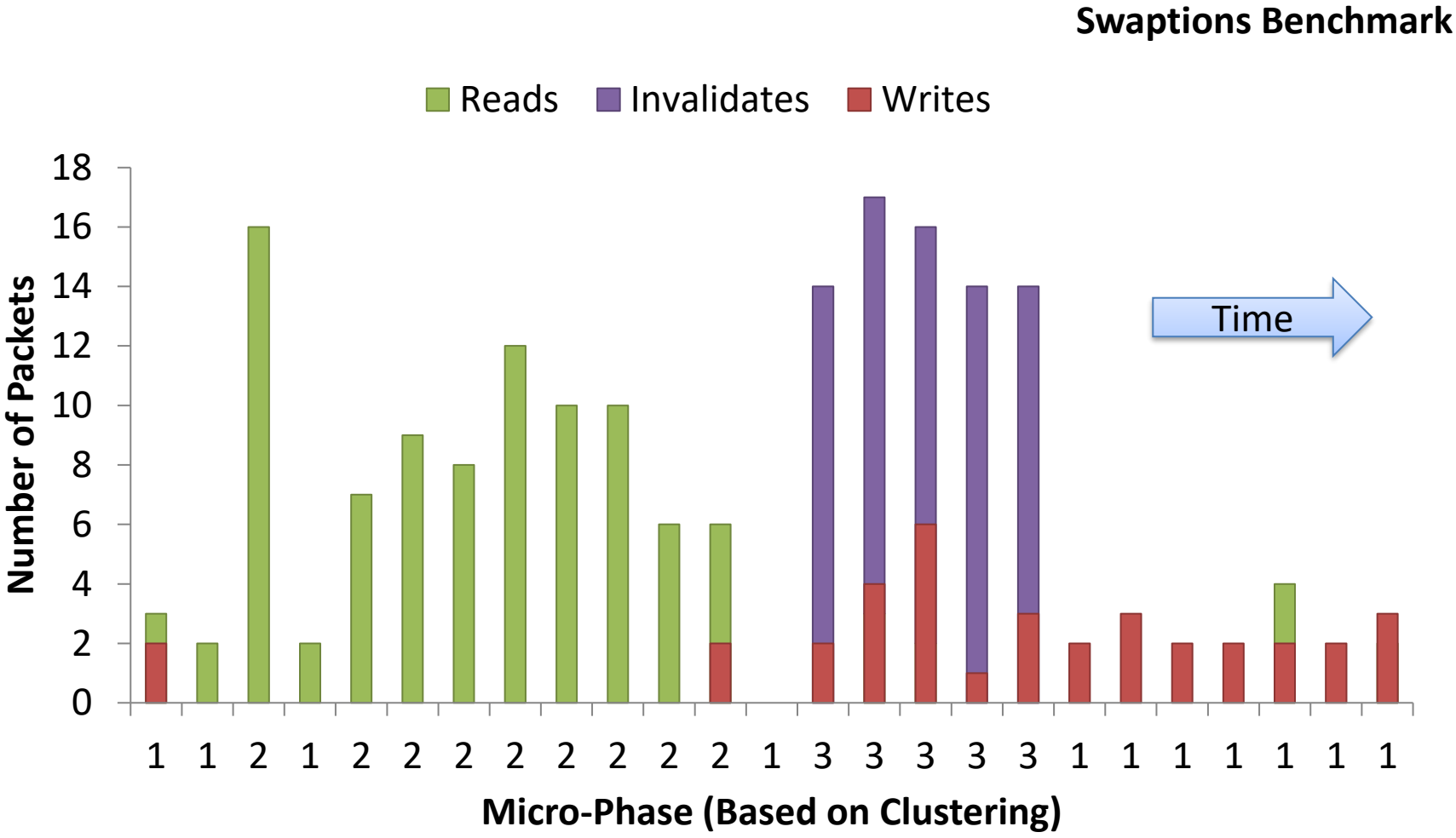
- Intervals also have time-varying behaviour!
- Create two levels:
  1. Macro Interval (big)
  2. Micro Interval (small)
- Repeat phase grouping for second level
- Hierarchical Model



Macro-Scale Model



Micro-Scale Model



Modelling Cache Coherent Behaviour

Want: Send Packets in a Cache Coherent manner

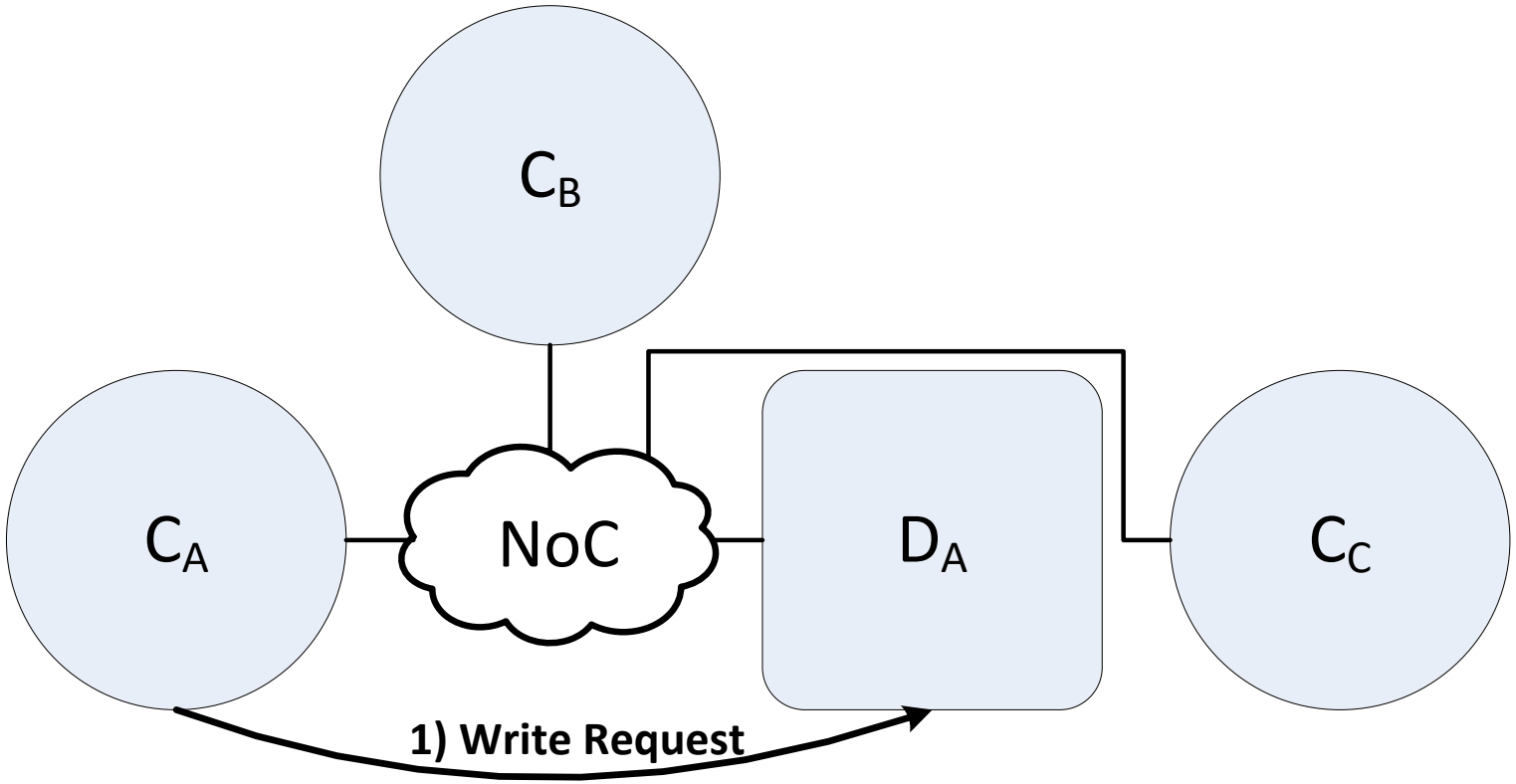
Need: Model for ***initiating Cache Coherent transactions***

1. Write Requests
2. Read Requests

Need: Model for Cache Coherence ***reactions***

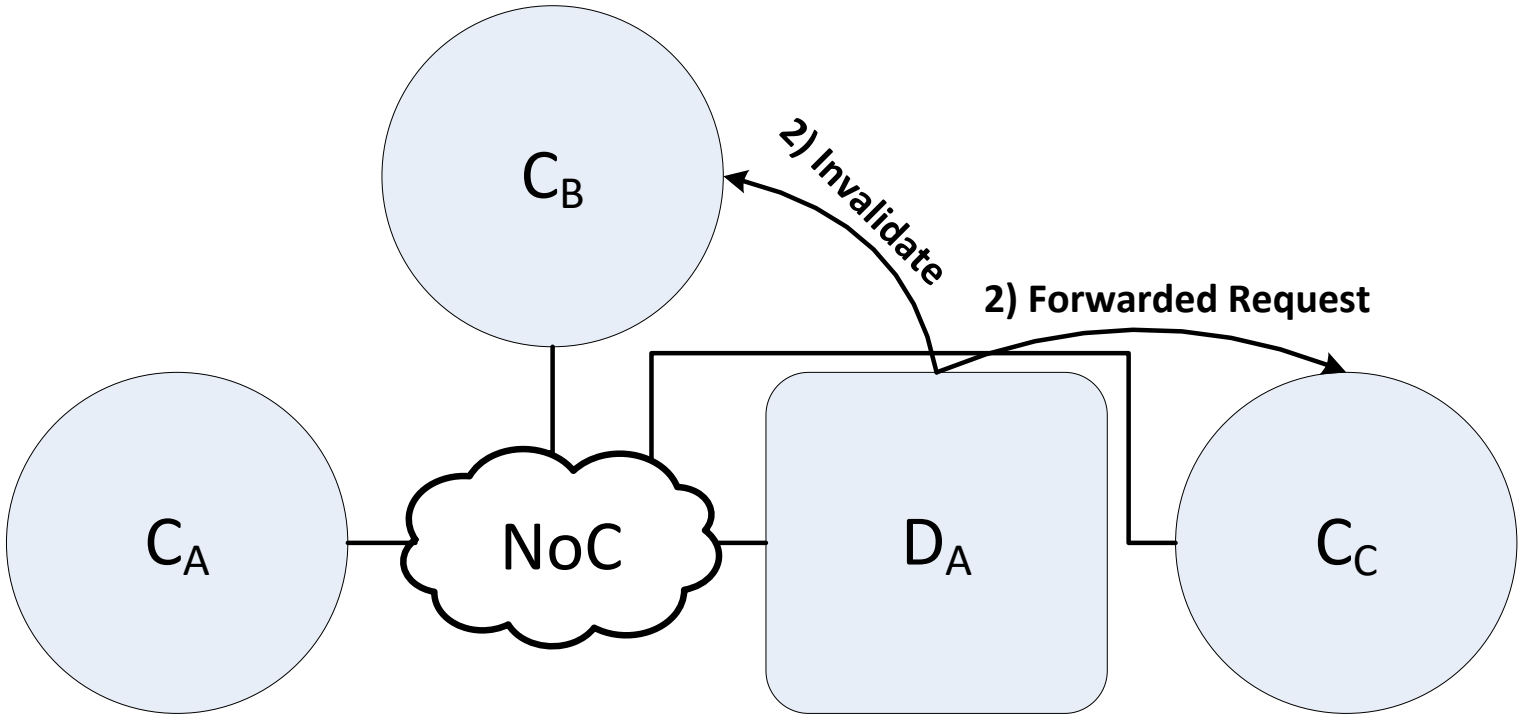
1. Accessing Memory
2. Invalidate Behaviour
3. Other Behaviour

Cache Coherence Example



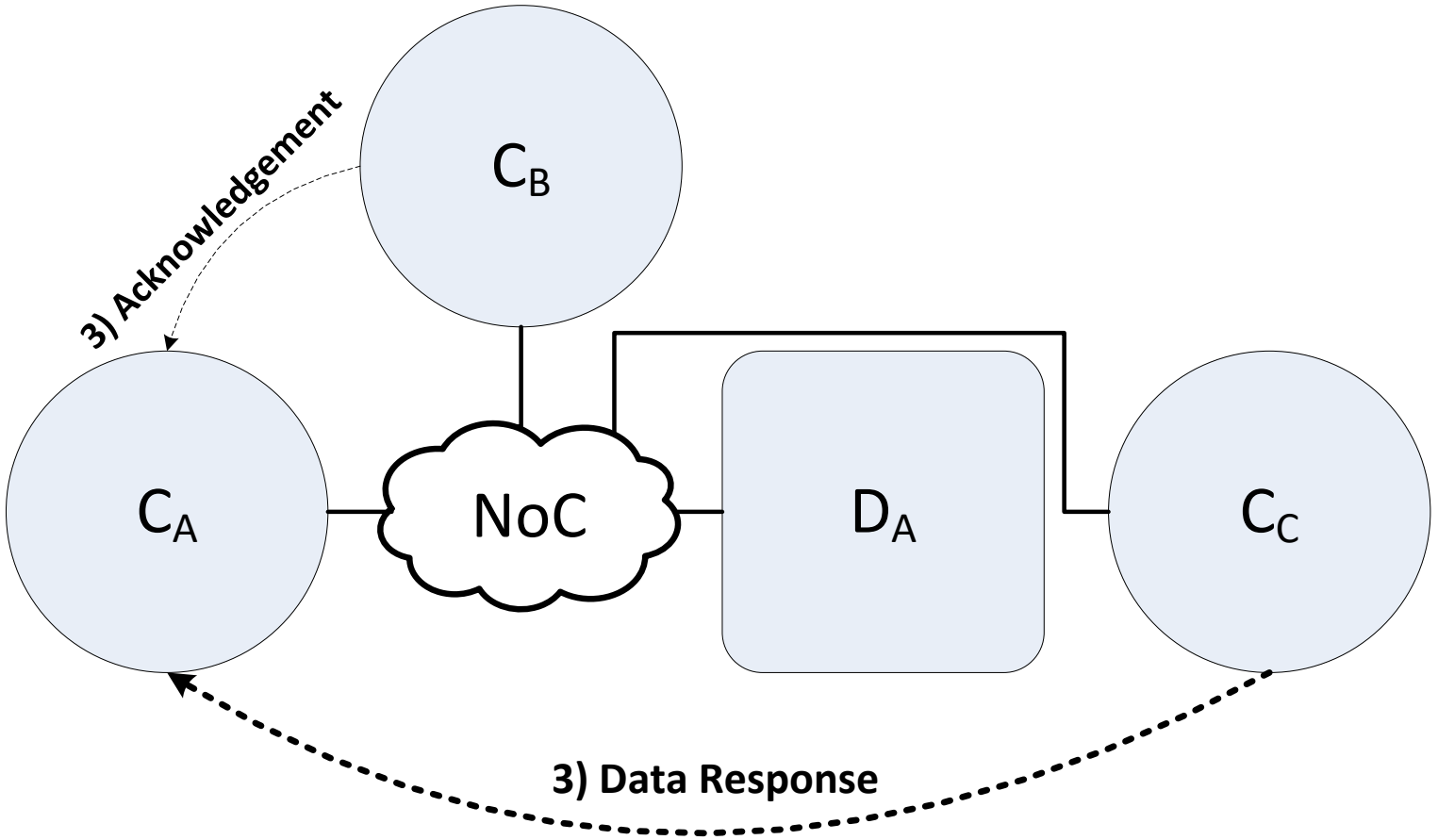
Two Sharers. Write Request initiates a transaction.

Cache Coherence Example



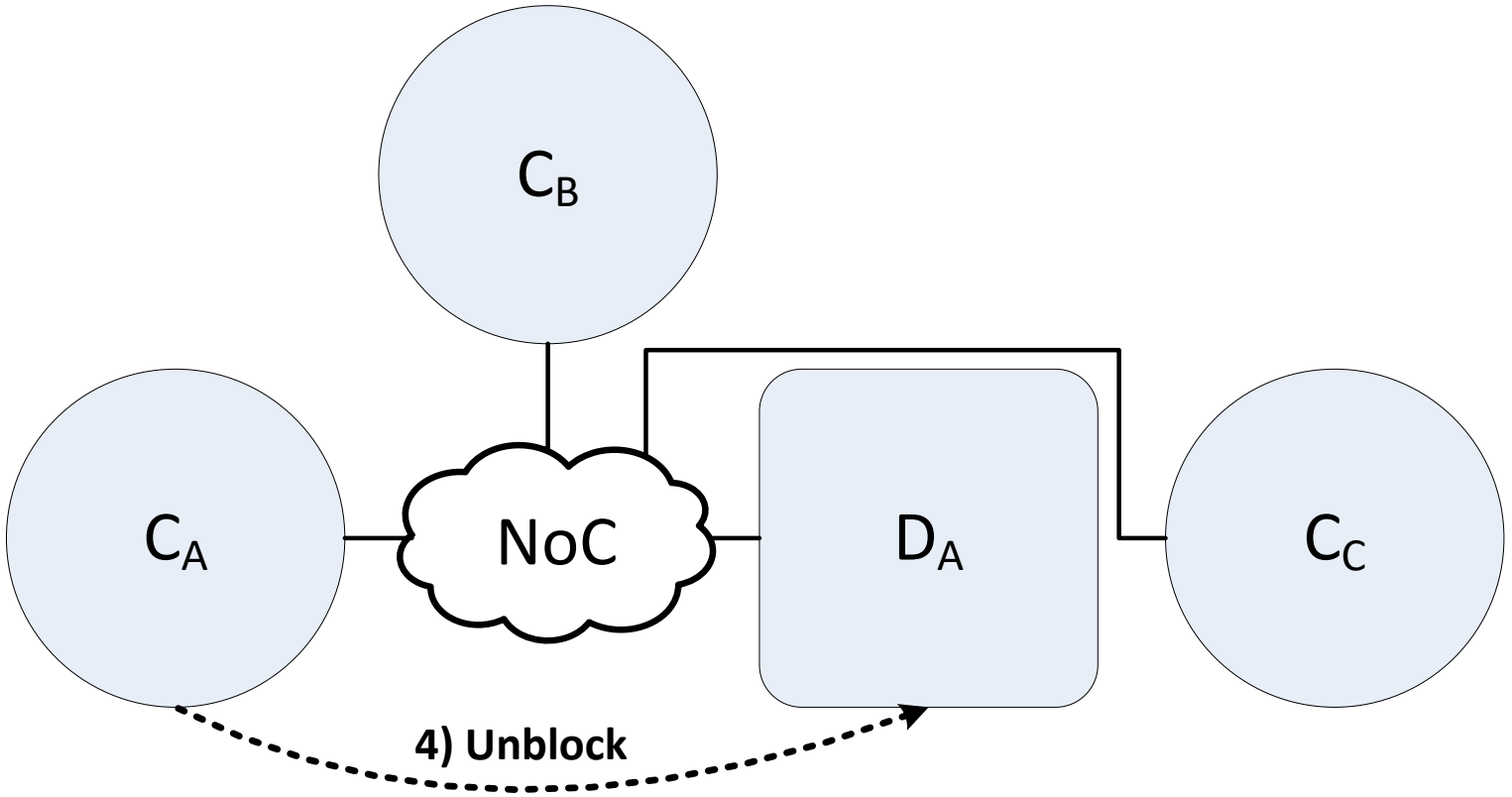
Write Request arrives at D_A

Cache Coherence Example



Invalidate arrives at C_B. Forward Request arrives at C_C.

Cache Coherence Example



Unblock signals end of the transaction.

Modelling Cache Coherent Behaviour

Active Model

- Write/Read Requests
  - How many?
  - Destination?
- Injected uniformly over a micro interval

Reactive Model

- Forward Requests
  - Probability of forwarding a request
  - Destination?
- Invalidate(s)
  - How many? (Could be zero)
  - Destination(s)?
- Other
  - Simplified Coherence Protocol

Modelling Cache Coherent Behaviour

Want: Send Packets in a Cache Coherent manner

Need: Model for ***initiating Cache Coherent transactions***

1. Write Requests
2. Read Requests

Need: Model for Cache Coherence ***reactions***

1. Accessing Memory
2. Invalidate Behaviour
3. Other Behaviour

Model Parameters

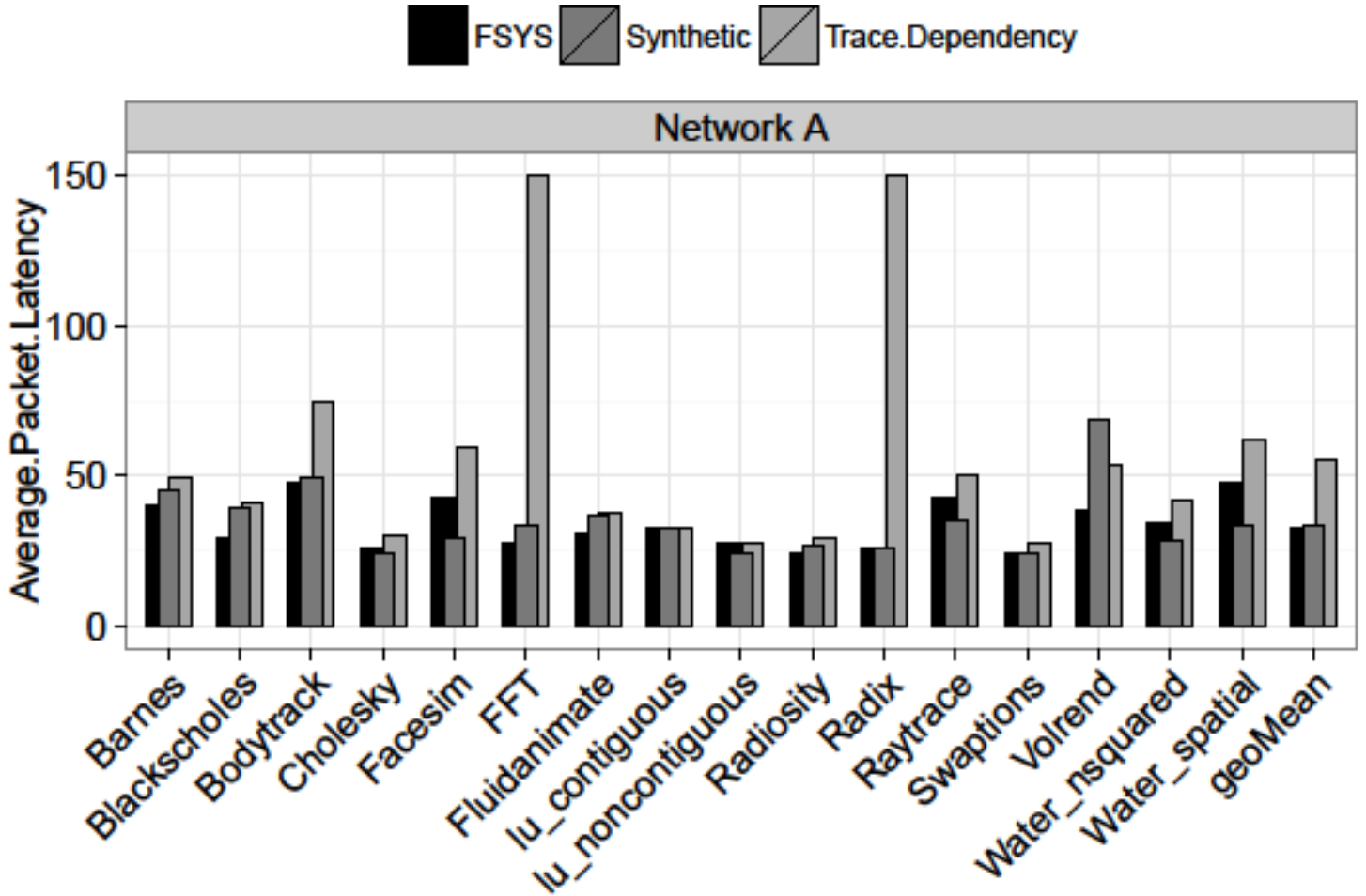
- Clustering Quality
  - Which feature vector?
  - Number of traffic phases
  - Which interval size?

	Macro-Level	Micro-Level
Feature Vector	Node Injection	RCFlow
Formal Method	Calinksi-Harabasz	L-Method
Interval Size	500,000	200

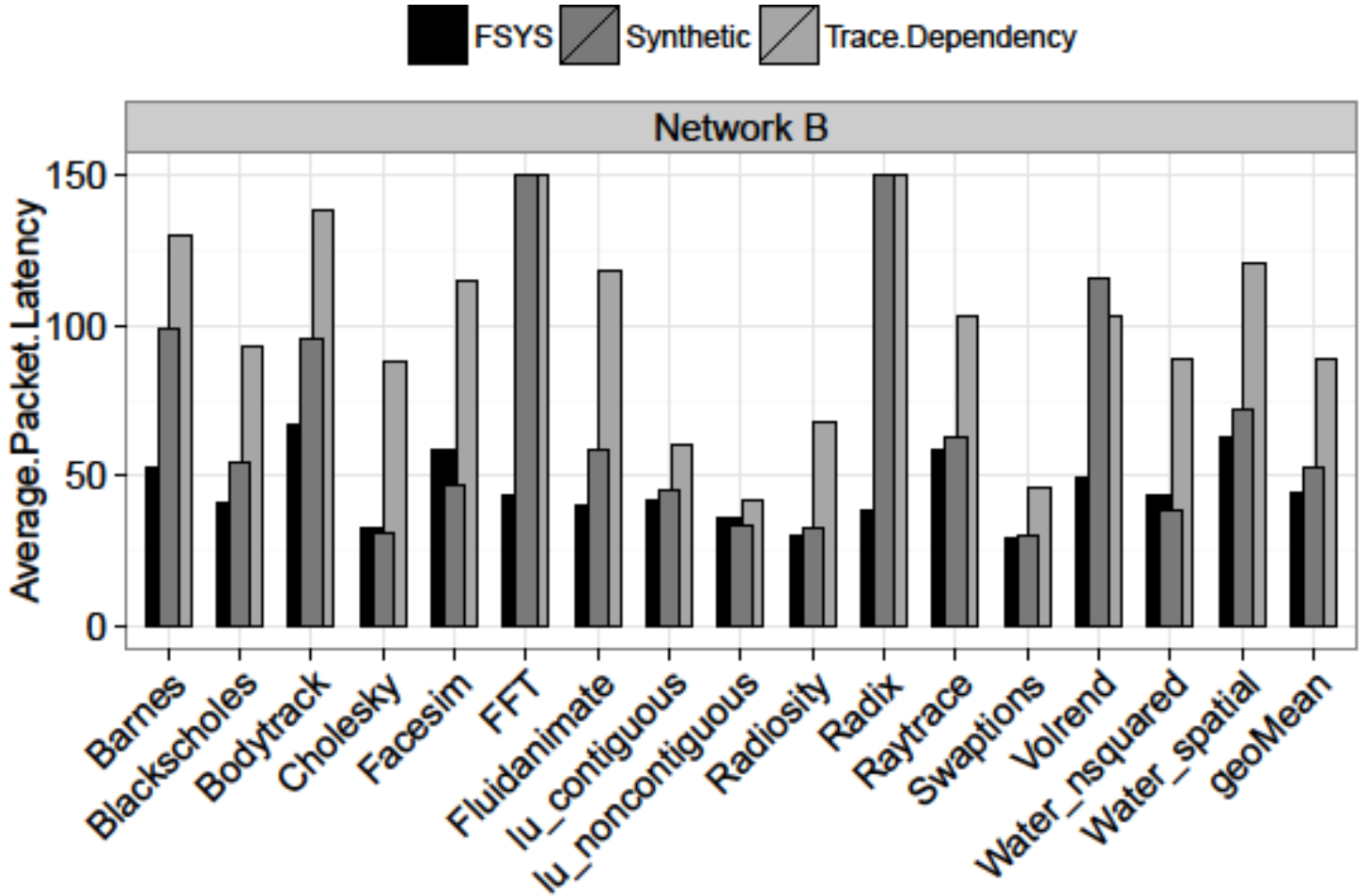
Network Configurations

Network	A	B	C
Topology	Mesh	Mesh	Flattened Butterfly
Channel Width	8 bytes	4 bytes	4 bytes
Virtual Channels	2 per port	2 per port	4 per port
Routing Algorithm	XY	Adaptive YX-XY	UGAL
Buffer Depth	8 flits		
Router Pipeline	4 stages		

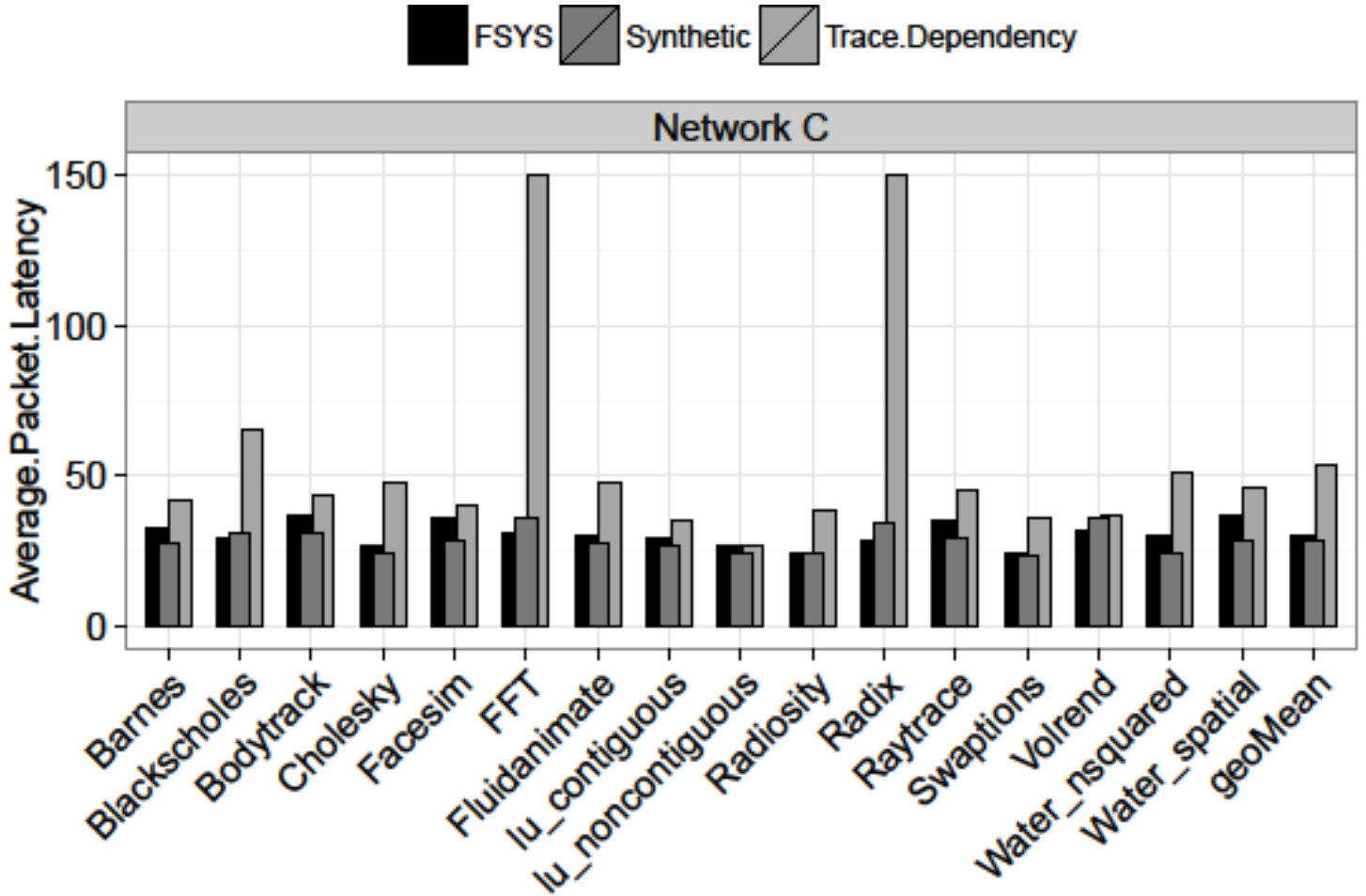
NoC Performance – Latency



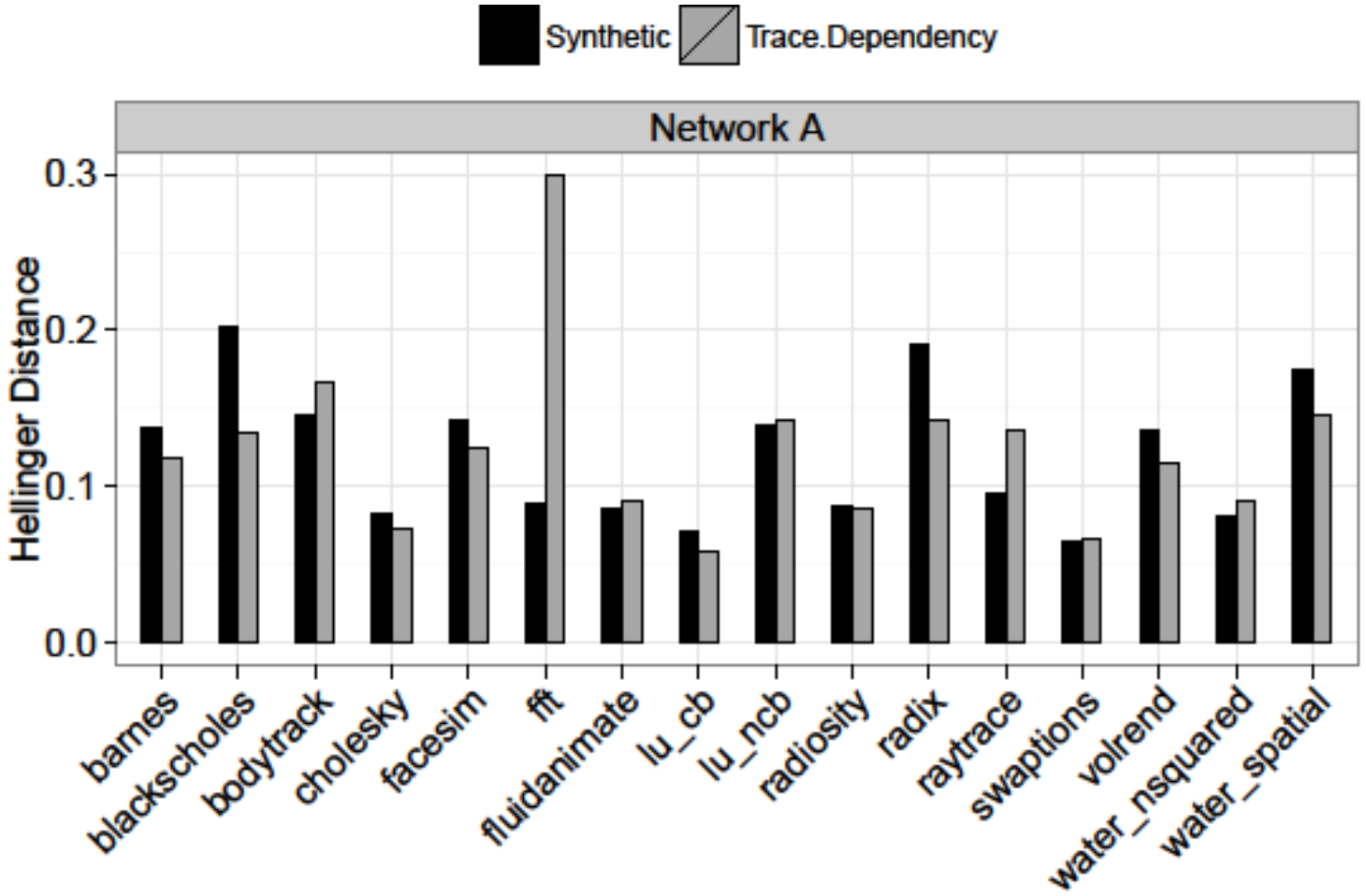
NoC Performance – Latency



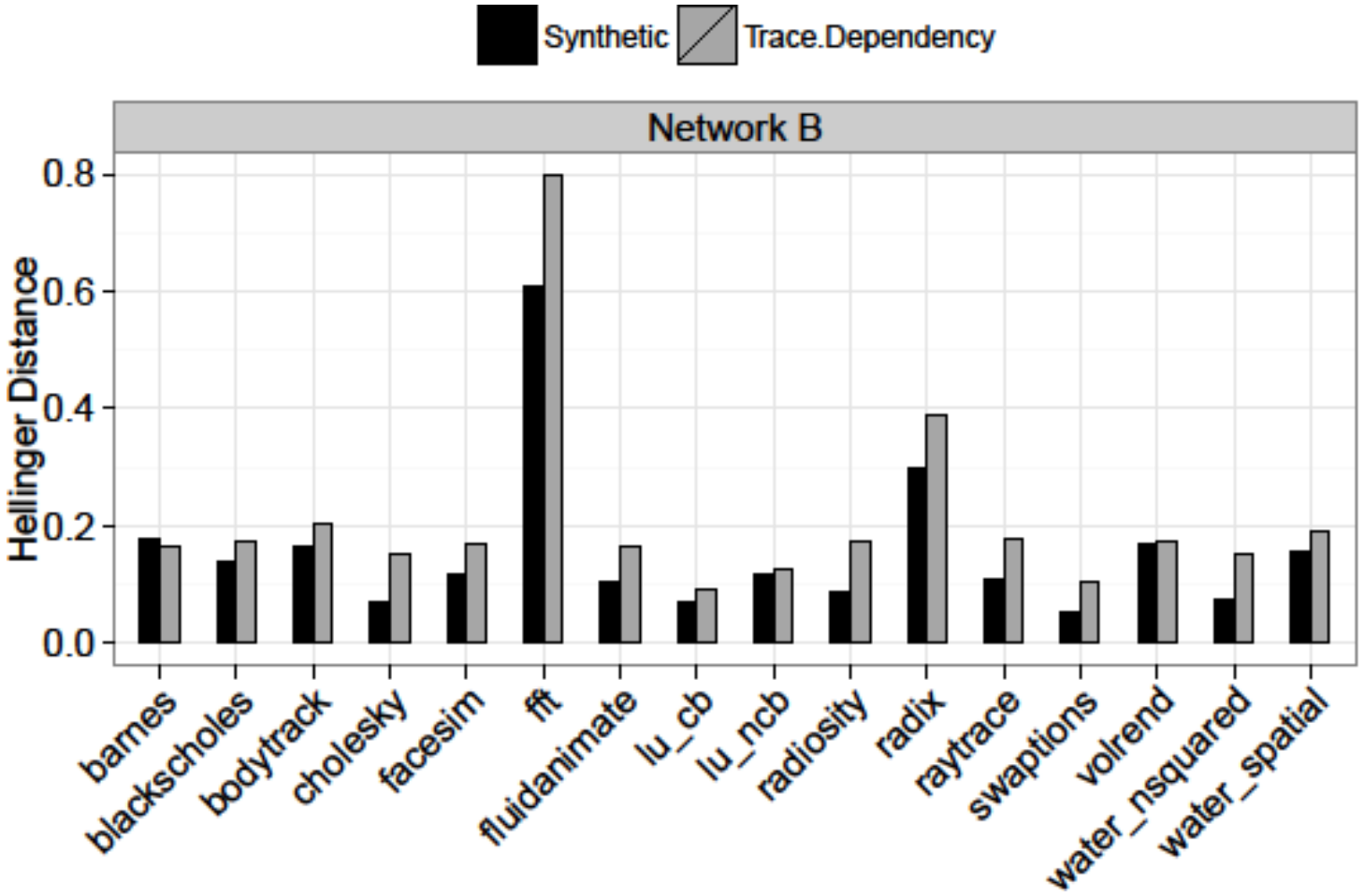
NoC Performance – Latency



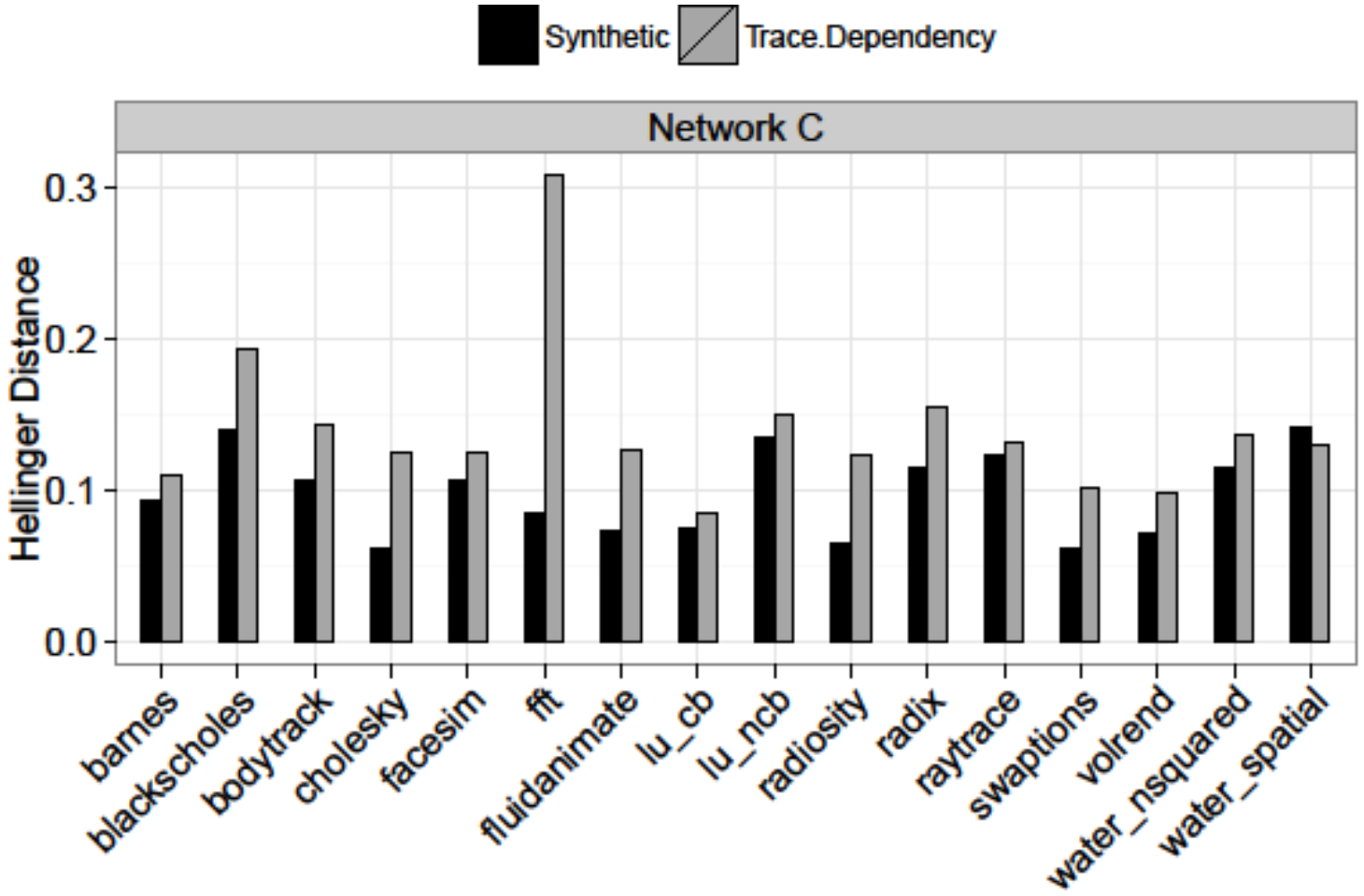
NoC Performance – Latency Distribution



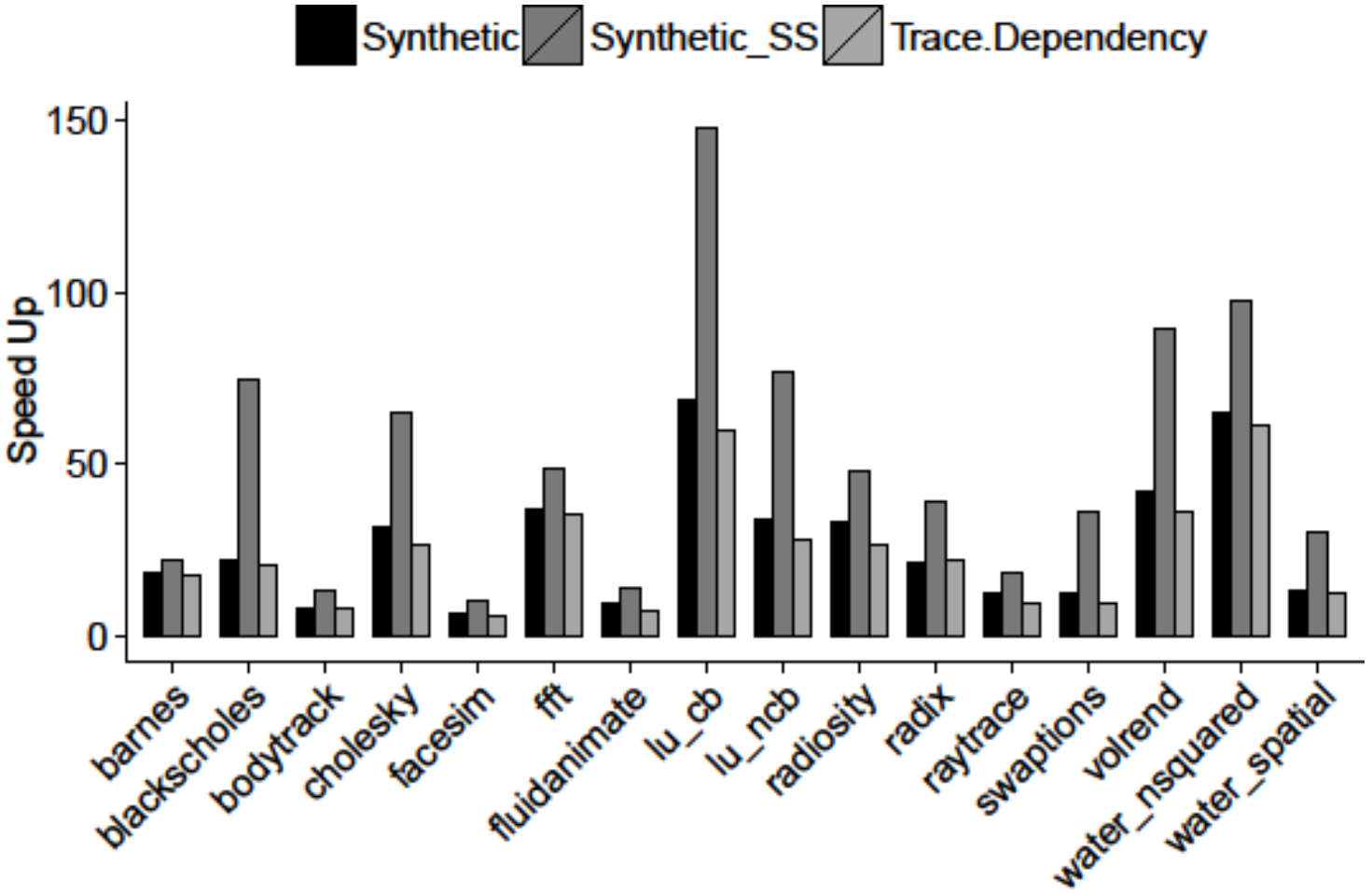
NoC Performance – Latency Distribution



NoC Performance – Latency Distribution



Speed Up


Synthetic
Synthetic_SS
Trace.Dependency
Speed Up
150
100
50
0
barnes
blackscholes
bodytrack
cholesky
facesim
fft
fluidanimate
lu_cb
lu_ncb
radiosity
radix
raytrace
swaptions
volrend
water_nsquared
water_spatial

Conclusion

- Implemented Synthetic Traffic Models that are
  - Accurate: 10.5% average erro
  - Fast: Over 50x average speed up
  - Coherent: Packets resemble cache coherent traffic
- Future Work
  - Sweeping micro-architectural configurations