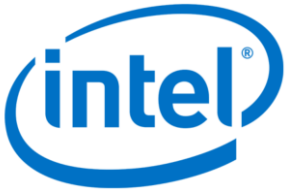
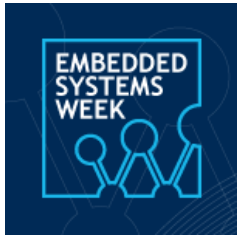
Analytical Performance Models for NoCs with Multiple Priority Traffic Classes

Sumit K. Mandal, Umit Y. Ogras
Arizona State University

Raid Ayoub, Michael Kishinevsky
Intel Corporation

CASES, October 15, 2019



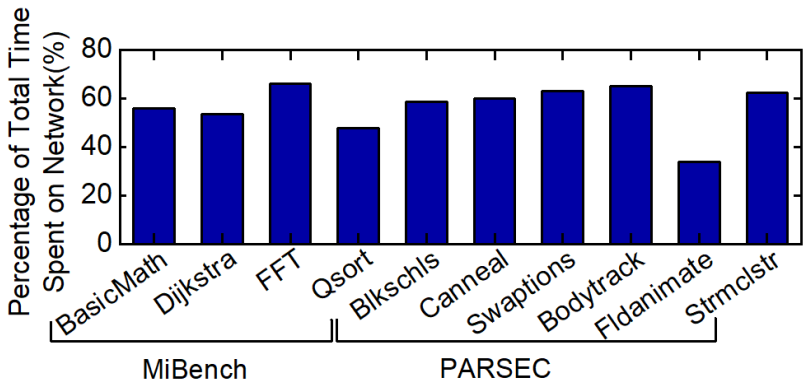


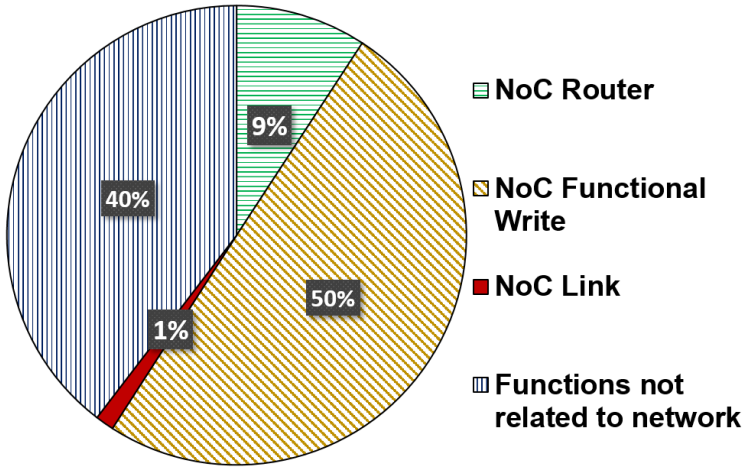
Outline

- **Need for analytical fabric model generation**
- **Background**
  - Networks-on-chip (NoC) used in industry
  - Prior work on NoC performance analysis
  - Prior work on queuing networks
- **Proposed network transformations**
- **Experimental results**
- **Conclusion and future work**

Performance Modeling for Emerging Applications

- **Examples of emerging applications**
  - Virtual reality, autonomous driving
  - Machine learning, AI
- **System modeling challenges**
  - Both SW/HW complexities grow
  - Long simulations needed for power, performance, and thermal analysis (***minutes instead of milliseconds***)
- **Research need**
  - Communication fabric: Central shared resource
  - Fast and accurate system level modeling
  - Automated generation of high-level performance models of **industrial SoCs**





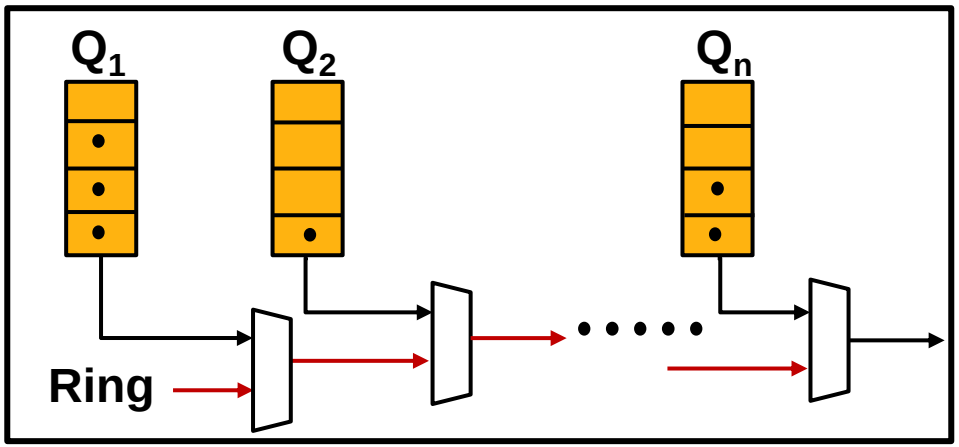
Outline

- Need for analytical fabric model generation
- **Background**
  - Networks-on-chip (NoC) used in industry
  - Prior work on NoC performance analysis
  - Prior work on queuing networks
- Proposed network transformations
- Experimental results
- Conclusion and future work

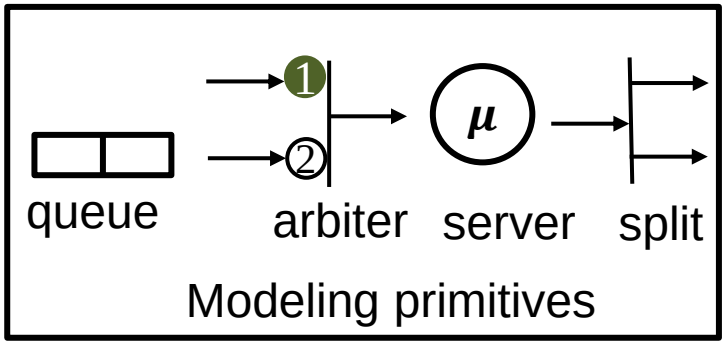
Priority Aware Networks-on-Chip Basics

- **Industrial NoCs use routers with priority arbitration to achieve predictable latency**
  - Packets in NoC have higher priority than new packets

Physical Network

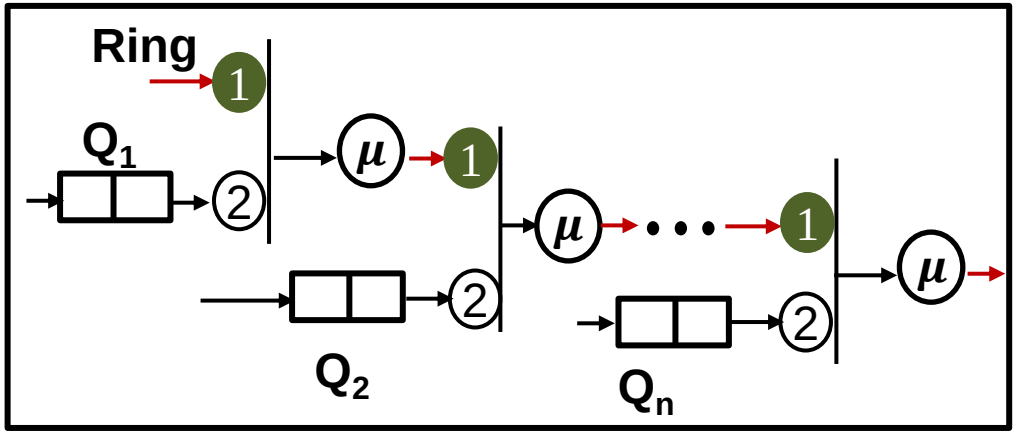


Notation



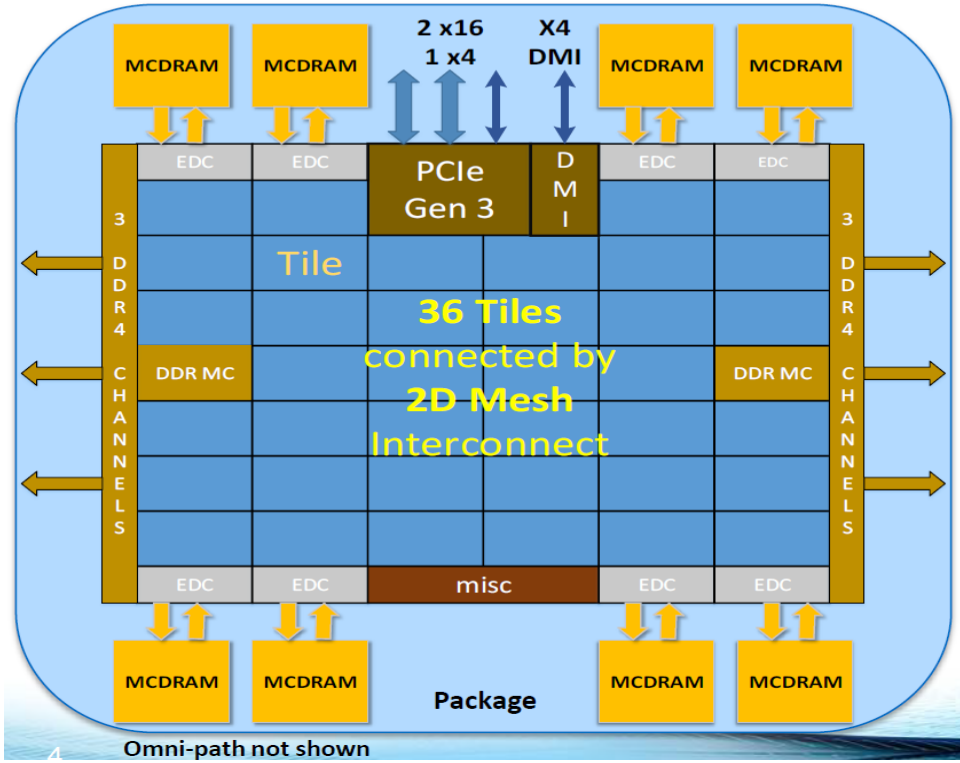
- Mux' in routers modeled as **priority arbiters** and **servers**
- Inputs with filled color denote higher priority

Abstract Model

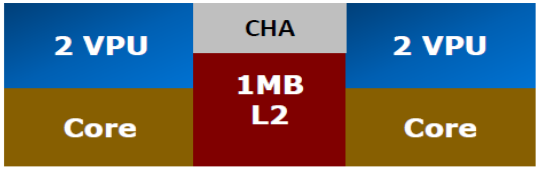


Example Fabric: Xeon Phi (KNL) Processor

Knights Landing Overview



TILE



Chip: 36 Tiles interconnected by **2D Mesh**
Tile: 2 Cores + 2 VPU/core + 1 MB L2

Memory: MCDRAM: 16 GB on-package; High BW
DDR4: 6 channels @ 2400 up to 384GB
IO: 36 lanes PCIe Gen3. 4 lanes of DMI for chipset
Node: 1-Socket only
Fabric: Omni-Path on-package (not shown)

Vector Peak Perf: 3+TF DP and 6+TF SP Flops
Scalar Perf: ~3x over Knights Corner
Streams Triad (GB/s): MCDRAM : 400+; DDR: 90+

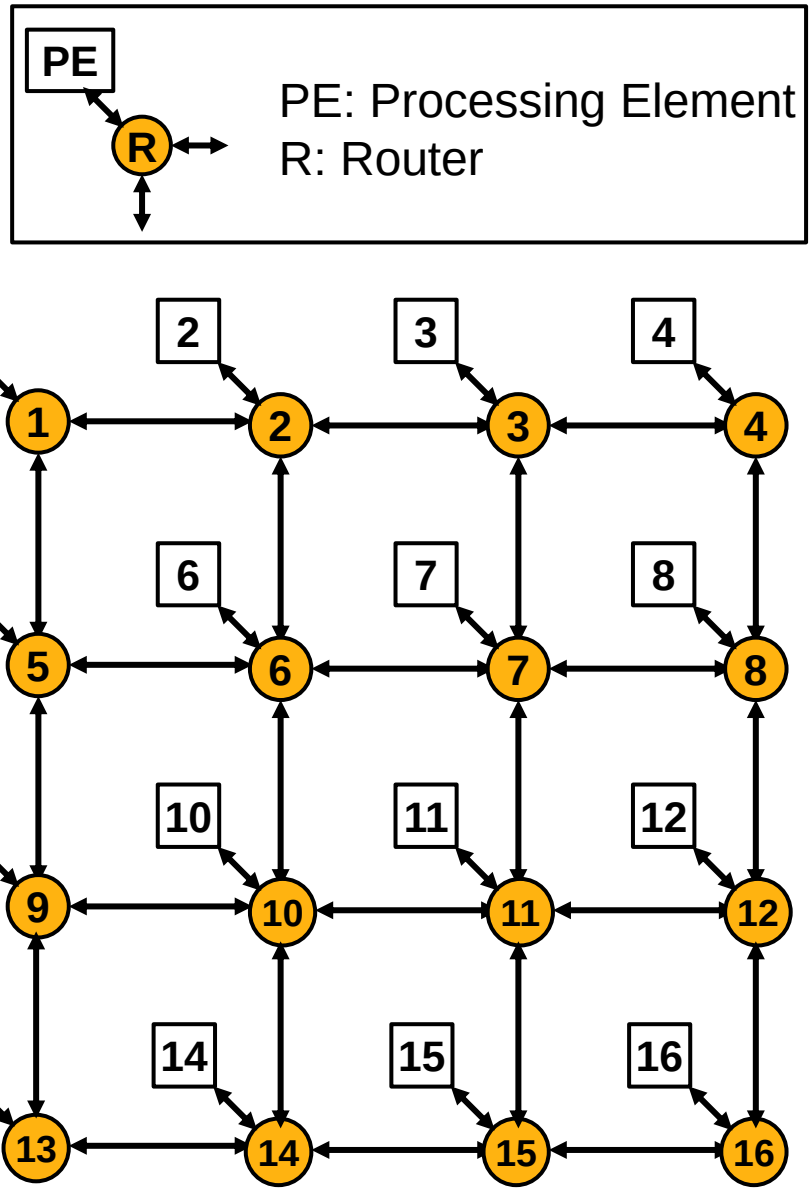
Source Intel: All products, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice. KNL data are preliminary based on current expectations and are subject to change without notice. 1Binary Compatible with Intel Xeon processors using Haswell Instruction Set (except TSX). 2Bandwidth numbers are based on STREAM-like memory access pattern when MCDRAM used as flat memory. Results have been estimated based on internal Intel analysis and are provided for informational purposes only. Any difference in system hardware or software design or configuration may affect actual performance.

Performance Analysis of Communication Fabric

- **A 4x4 mesh with YX routing**
- **Source to destination latency (L_{SD}) has four components**
  - Waiting time in source queue (W_Q^S)
  - Deterministic vertical latency (L_v)
  - Waiting time at the junction (W_Q^J)
  - Deterministic horizontal latency (L_h)
- **L_v and L_h depend on the source-destination pair and fabric topology**

$$L_{SD} = W_Q^S + L_v + W_Q^J + L_h$$

- **W_Q^S and W_Q^J depend on injection rates at different queues and need detailed analysis**



Outline

- Need for analytical fabric model generation
- Background
  - Networks-on-chip (NoC) used in industry
  - Prior work on NoC performance analysis
  - Prior work on queuing networks
- Proposed network transformations
- Experimental results
- Conclusion and future work

Prior Work on Performance Analysis of Networks

NoC	Priority-based	Multiple classes	Scalable
[1,2,5]	✗	✓	✓
[3,4]	✓	✗	✓

Off-chip Network	Priority-based	Multiple classes	Scalable
[6,7]	✓	✗	✓
[8]	✓	✓	✗

NoC	Priority-based	Multiple classes	Scalable
This work	✓	✓	✓

[1] Ogras et al. "An analytical approach for network-on-chip performance analysis." *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems* 29.12 (2010).

[2] Bogdan et al. "Non-stationary traffic analysis and its implications on multicore platform design." *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems* 30.4 (2011).

[3] Kiasari et al. "An analytical latency model for networks-on-chip." *IEEE Trans. on Very Large Scale Integration Systems*21.1 (2013).

[4] Kashif et al. "Bounding buffer space requirements for real-time priority-aware networks." *Design Automation Conference (ASP-DAC), 2014 19th Asia and South Pacific*.

[5] Qian et al. "A support vector regression (SVR)-based latency model for network-on-chip (NoC) architectures." *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems* 35.3 (2016).

[6] Bertsekas et al. *Data networks*. Vol. 2. New Jersey: Prentice-Hall International, 1992.

[7] Walraevens, Joris. *Discrete-time queueing models with priorities*. Diss. Ghent University, 2004.

[8] G. Bolch et al. Queuing Networks and Markov Chains. Wiley. 2006

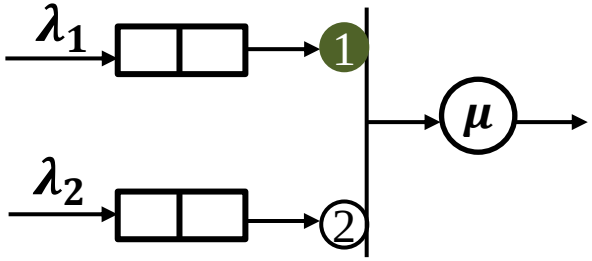
Background: Queuing Systems

λ = arrival rate, μ = service rate

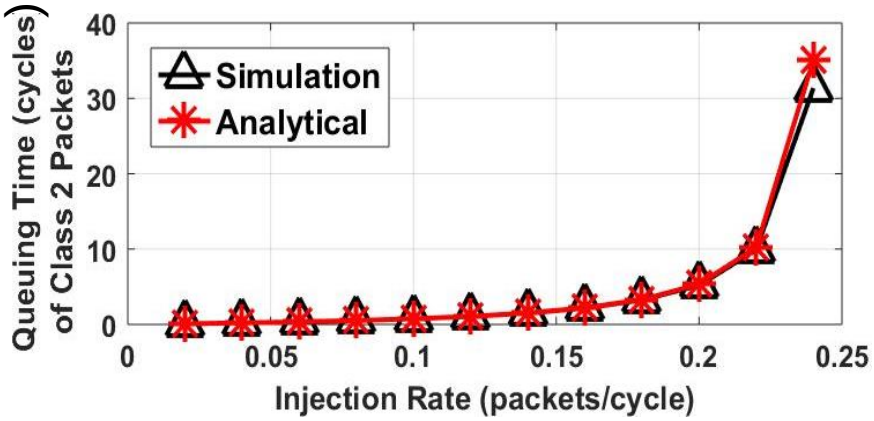
Server utilization (ρ) = $\frac{\lambda}{\mu}$

- **Kendall's notation for queuing discipline: A/B/m**
- **Arrival and departure may have different distribution (e.g. Poisson (M), Deterministic (D) , General (G)).**

Priority Rule: (1>2)



Simulation vs Analytical for Basic Priority



$$W_1 = \frac{R_1}{1-\rho_1},\ W_2 = \frac{R_2+\rho_1 W_1}{1-\rho_1-\rho_2},\ \rho_i = \frac{\lambda_i}{\mu_i}$$

W: average waiting time, T: service time, R: average residual time

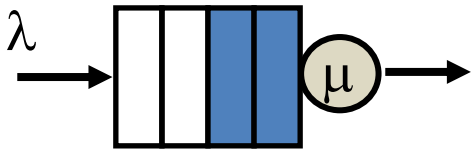
Residual Time for Single Queue Node

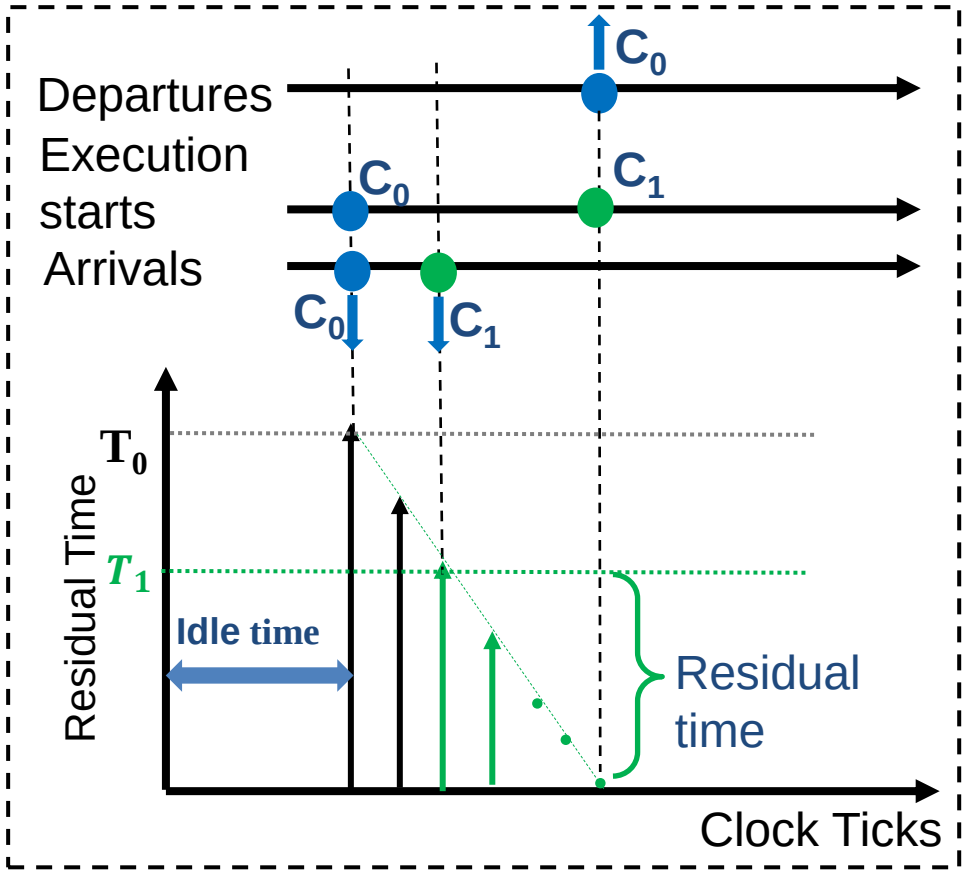
- **Residual time (R):** delay of serving the next token due to the remaining service time for currently processed token
- Arrival distribution is Geometric
  - $P\{X = k\} = p(1-p)^{k-1}$

$$R_{avg} = \frac{1}{T_{tot}} \sum_{i=1}^{M(T_{tot})} \left(\sum_{\tau=0}^{T_i - 1} \tau \right)$$

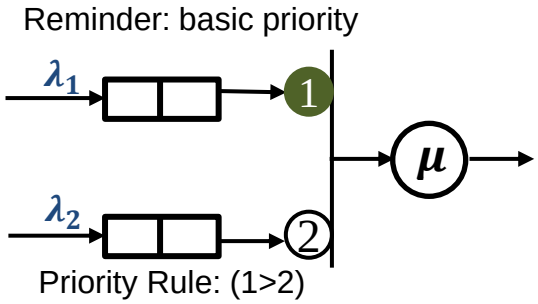
$$= \frac{1}{2} \lambda \left(\overline{T^2} - \overline{T} \right) \text{ (for Geo/G/1)}$$

$$= \frac{1}{2} \rho (T - 1) \text{ (for Geo/D/1)}$$





Residual Time for Basic Priority Queue (Geo/G/1)



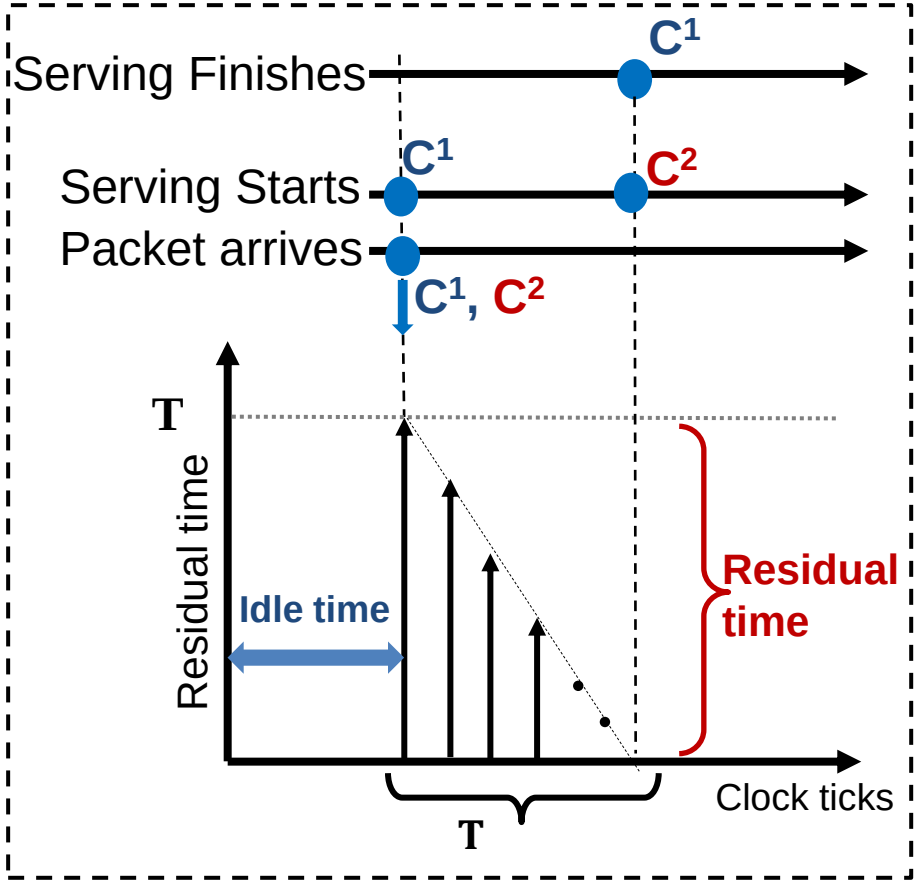
- **Expression for residual time of class 1 packets**

$$R_1 = \frac{1}{2}\rho_1(T-1) + \frac{1}{2}\rho_2(T-1)$$

- **Expression for residual time of class 2 packets**

$$R_2 = \frac{1}{2}\rho_1(T+1) + \frac{1}{2}\rho_2(T-1)$$

- **Class 2 packets have higher residual time due to lower priority**

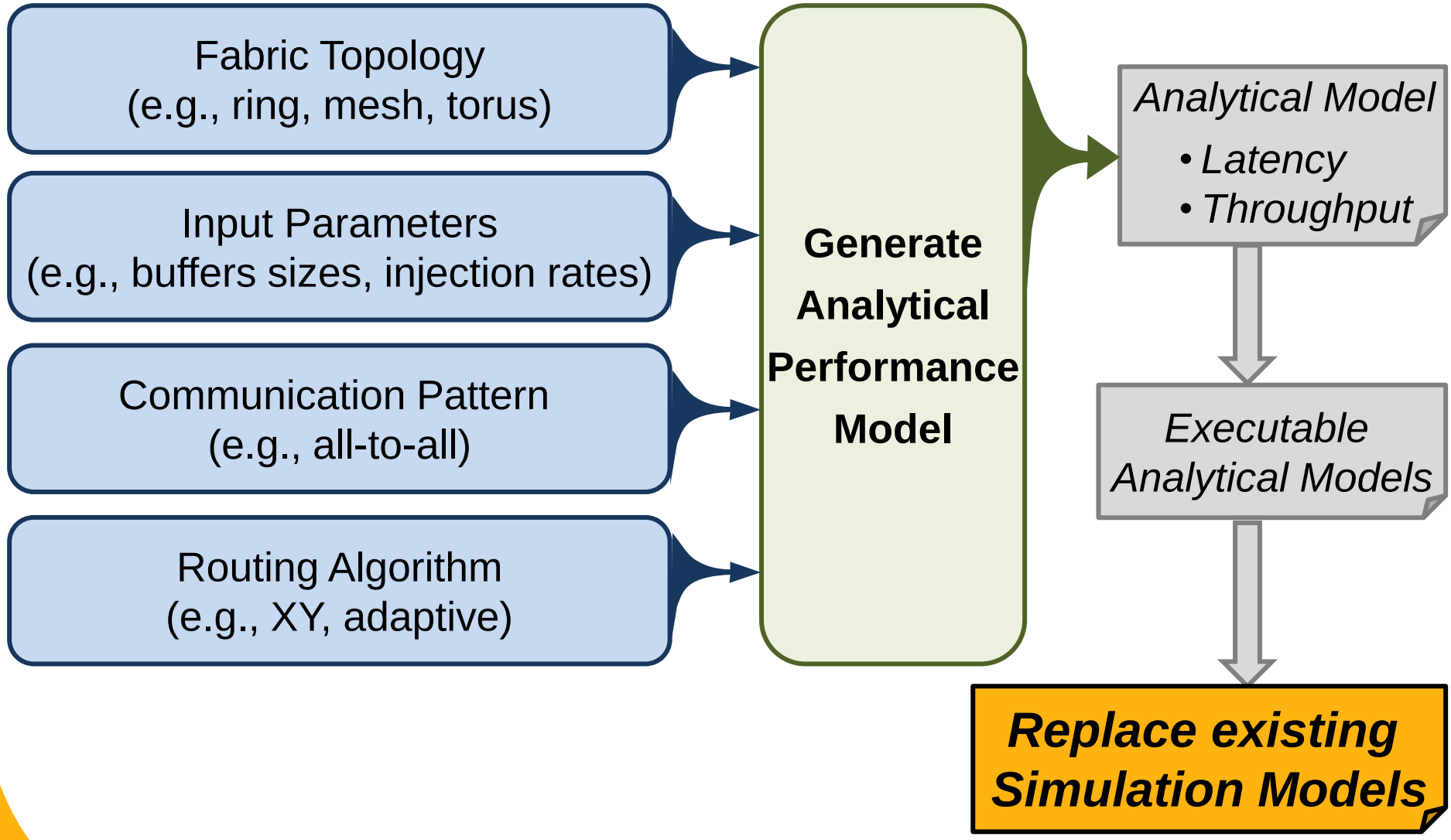


T = service time
M(T_{tot}) = total no. of packets arrived during time interval T_{tot}
τ = intermediate variable for sum

Outline

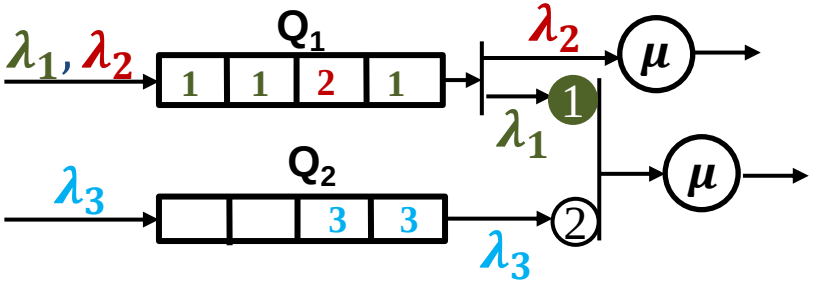
- Need for analytical fabric model generation
- Background
  - Networks-on-chip (NoC) used in industry
  - Prior work on NoC performance analysis
  - Prior work on queuing networks
- **Proposed network transformations**
- Experimental results
- Conclusion and future work

Overview of the Automated Flow

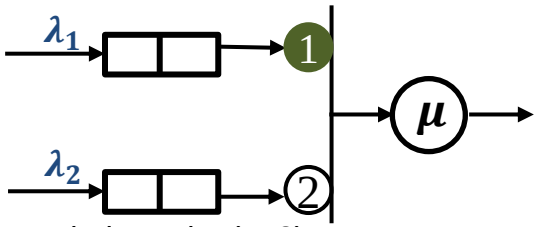


Limitation of Basic Priority Based Models (1)

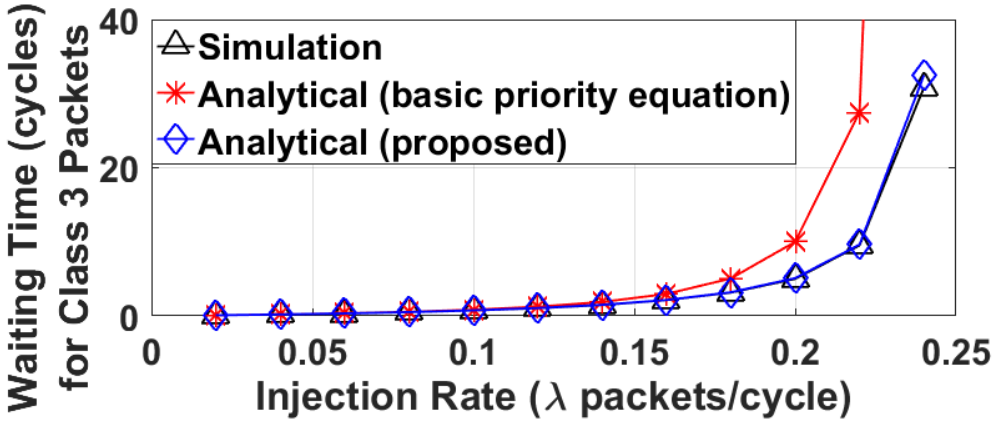
Split on high priority



Reminder: basic priority



Priority Rule: (1>2)

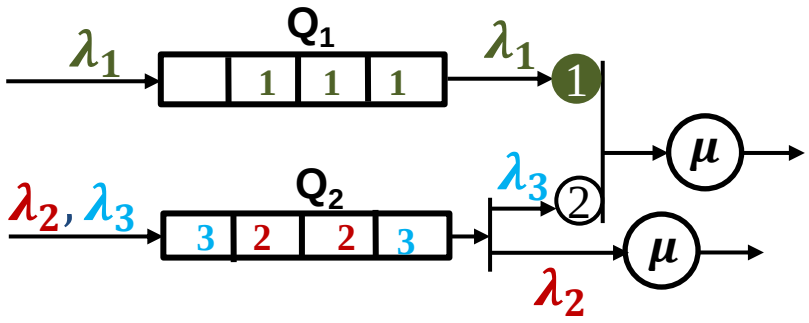


Applying basic priority equation for class 3 tokens results in **pessimistic solution**

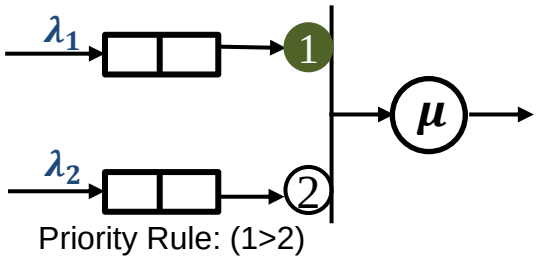
Reason: Not all tokens of class 1 will block tokens of class 3. Tokens of class 3 can occupy the server if a class 2 token is being served

Limitation of Basic Priority Based Models (2)

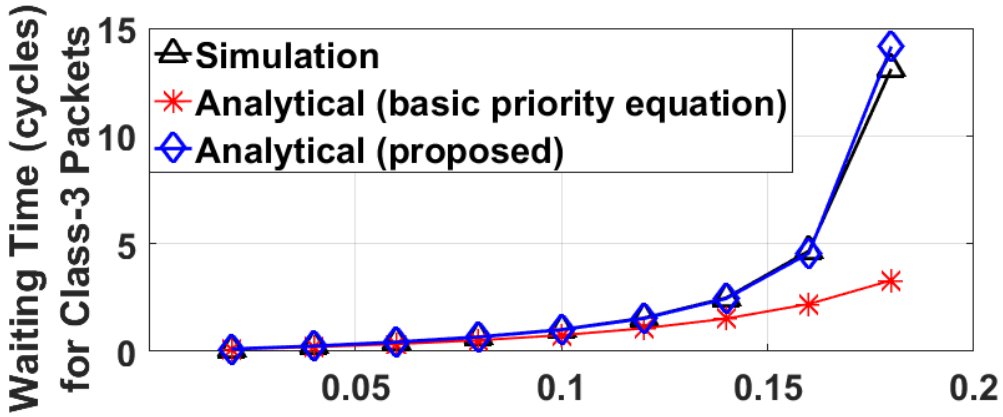
Split on low priority



Reminder: basic priority



Priority Rule: (1>2)



Applying basic priority equation for class 3 tokens results in **optimistic solution**

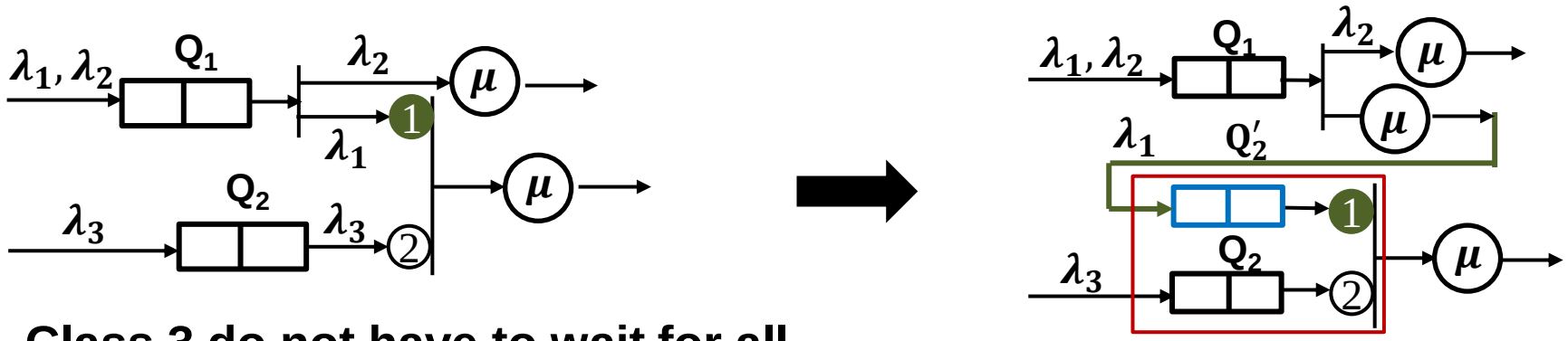
Reason: Tokens of class 2 will have effect of class 1 indirectly as class 2 tokens have to wait due to class 3 tokens

Proposed Network Transformations

- **Extend decomposition method [1] to handle priority arbitration based multi-class networks in industry**
- **We identified two transformations**
  - ST: structural transformation
  - RT: service rate transformation
- **Complex priority-based networks are decomposed iteratively to systems of equivalent queues using ST/RT**
- **Obtain a closed form analytical expression for the equivalent system**

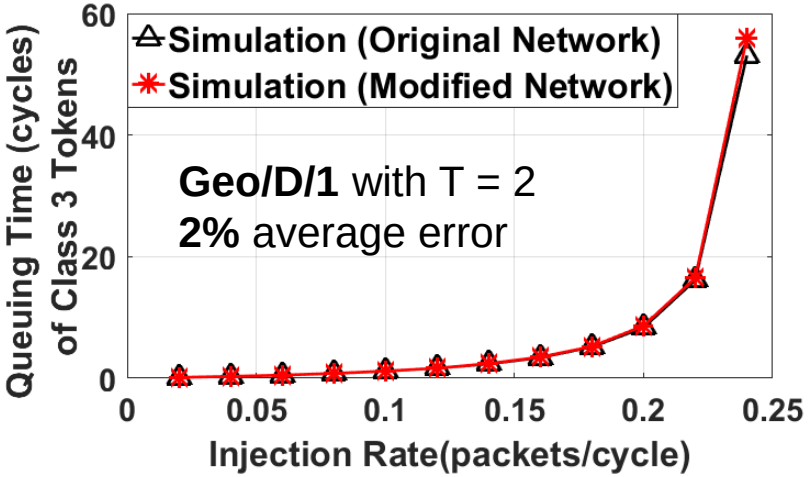
[1] G. Bolch et al. Queuing Networks and Markov Chains. Wiley. 2006

Structural Transformation (ST)



- **Class 3 do not have to wait for all packets in Q_1**
  - Class 3 and 2 can be served simultaneously
- **Class 3 packets will wait only when the server is busy serving class 1**
  - Need to decompose class 1 and class 2
- **Proposed transformation separates class 1 tokens and put in a virtual queue (Q_2)**
- **Equivalence is demonstrated by the result shown on the right**

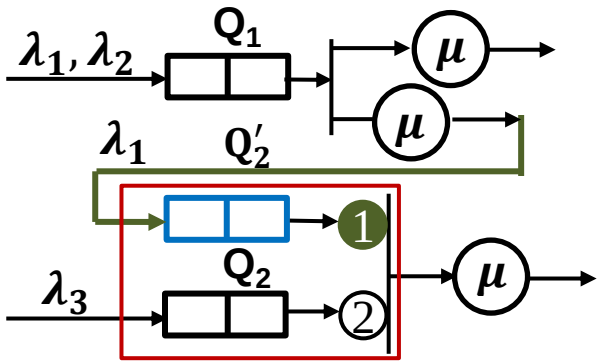
Comparison of Original and Modified Network



Analytical Model after Structural Transformation

- **ST enables us to derive closed form analytical equations**
- **Expression of residual time of class 1 in Q_2'**

$$R_1^{Q_2'} = \frac{1}{2}\rho_3(T-1) + \frac{\frac{1}{2}\rho_3 T\left(C_{A_1}^2 + 1 - \mu\right)}{2} - \frac{\rho_1\mu}{2}$$



- **Waiting time of class 1 in Q_2'**

$$W_1^{Q_2'} = \frac{R_1^{Q_2'}}{1-\rho_1}$$

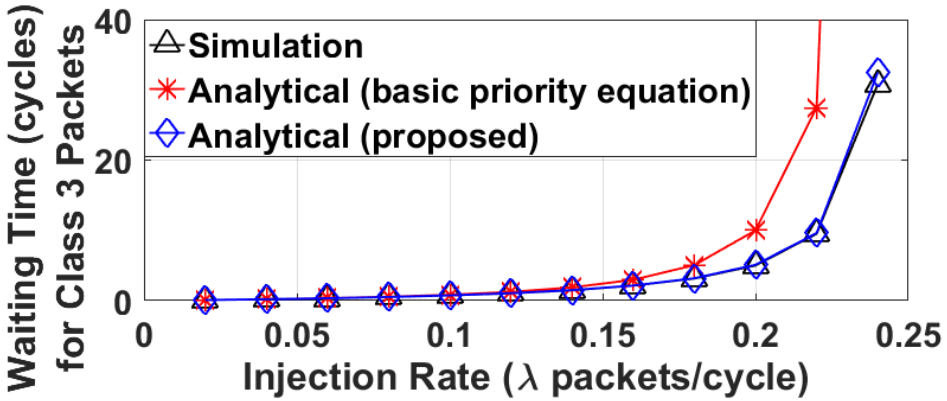
- **Residual time of class 3**

$$R_3 = R_1^{Q_2'} + \rho_1$$

- **Finally, waiting time of class 3**

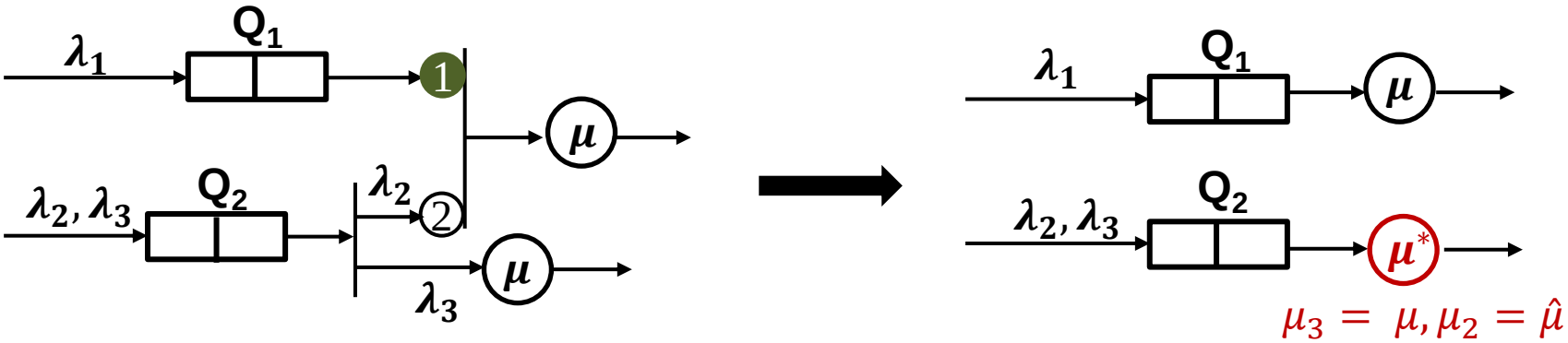
$$W_3 = \frac{R_3 + \rho_1 W_1^{Q_2'}}{1 - \rho_1 - \rho_3}$$

Comparison of Simulation and Analytical Models



Geo/D/1 with T = 2, **3%** average error

Service Rate Transformation (RT)



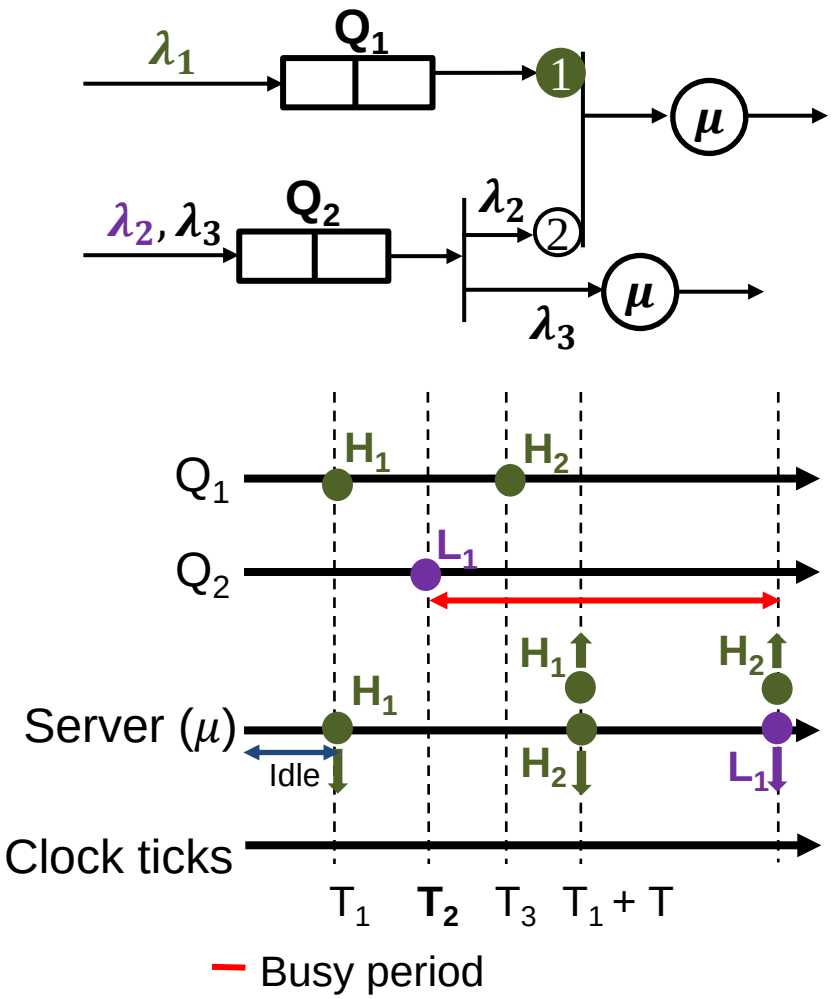
Observations

- **Packets in Q_1 effectively increase the service time of class 2 packet**
  - **Need to modify the service time of class 2 packet**
- **Challenging to model** since not all packets in Q_2 will wait for packets in Q_1
- **Insight:** Modified service time of class 2 is independent of incoming distribution of class 3

Proposed Approach

- **Decompose priority arbitration by modifying the service time of class 2**
- **Approximate first and second order moments of modified service time ($\hat{\mu}$)**
- $\hat{\mu} = \frac{1}{\hat{T}} = \frac{1}{T+\Delta T}$**, calculation of ΔT is shown next**

Service Rate Transformation (RT): 1st moment



Calculation of average busy period ($\Delta\bar{T}$)

Let p be the probability that the server is occupied by high priority token

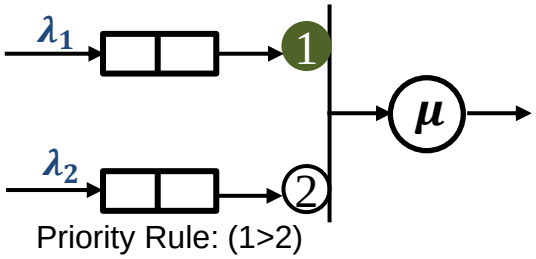
If low priority token is blocked once, it will see a busy period of $\frac{1}{T}\sum_{i=1}^{T} i = \frac{T+1}{2}$ in average

$$\Delta\bar{T} = \left(\frac{T+1}{2}\right)p(1-p) + \left(\frac{T+1}{2} + T\right)p^2(1-p) \ldots.$$

$$\Rightarrow \boldsymbol{\Delta\bar{T} = \frac{pT}{1-p} - \left(\frac{T-1}{2}\right)p}, \qquad p = \rho_1 + \lambda_1 R$$

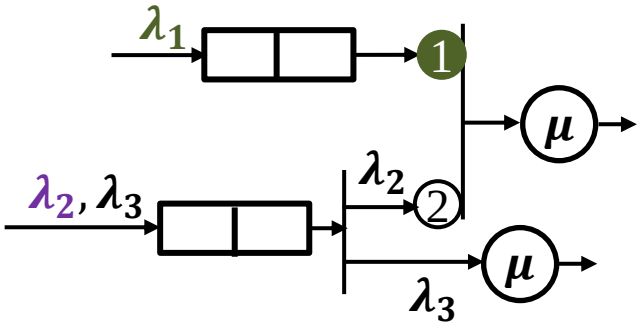
T: service time

Service Rate Transformation (RT): 2nd moment



Priority Rule: (1>2)

(a)



(b)

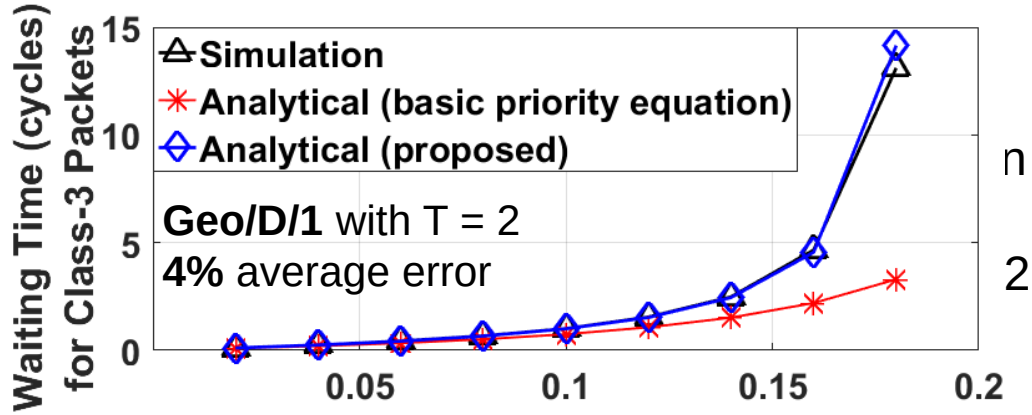
Simple Priority

$$W_2 = \frac{R_2 + \rho_1 W_1}{1 - \rho_1 - \rho_2}$$

Service Rate Transformation

$$W_2 = \frac{\widehat{R_2}}{1 - \widehat{\rho_2}} + \overline{\Delta T}, \widehat{\rho_2} = \lambda_2 (T + \overline{\Delta T})$$

Comparison of Simulation and Analytical Models



$$\widehat{R_2} = (W_2 - \overline{\Delta T})(1 - \widehat{\rho_2})$$

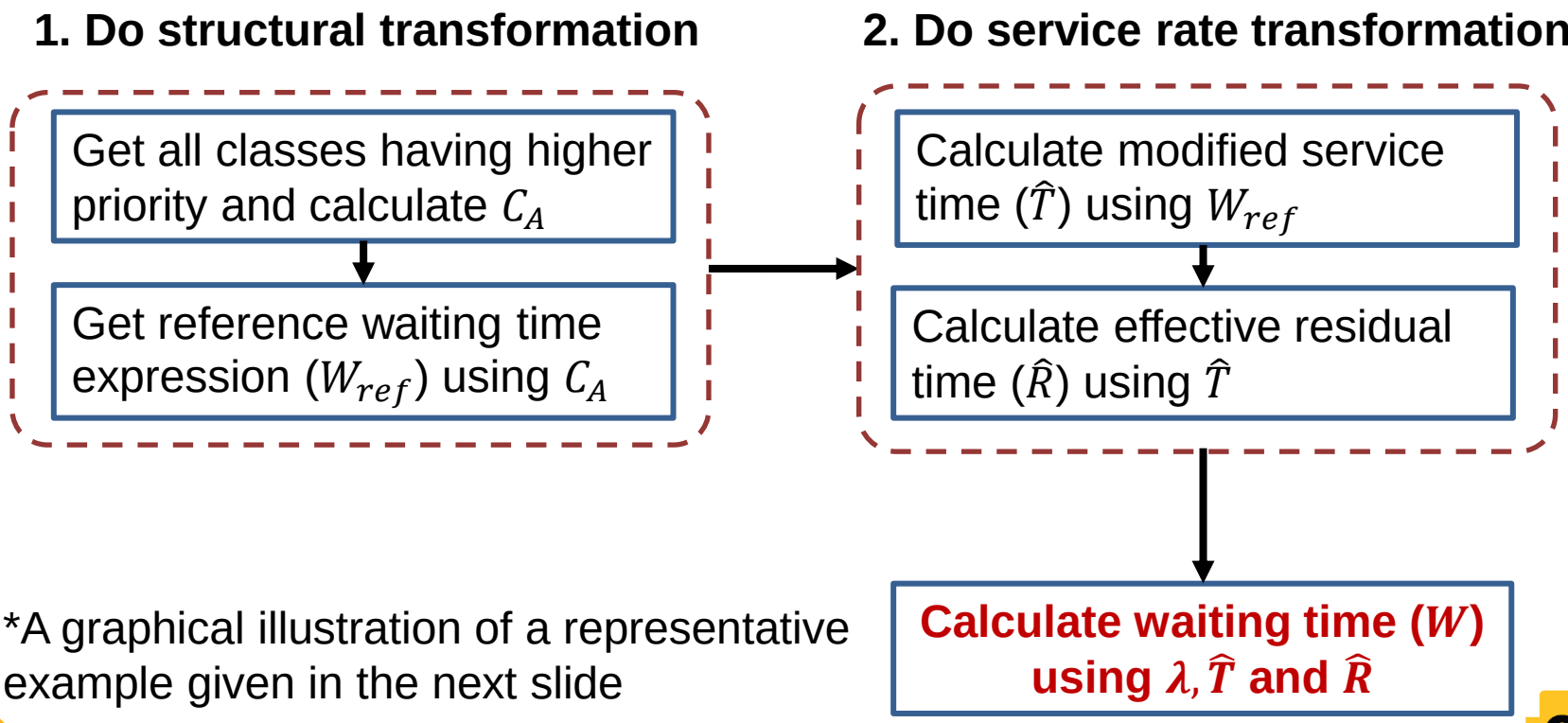
$$W_3 = \frac{\widehat{R_2} + R_3}{1 - \rho_3 - \widehat{\rho_2}}, W_2 = W_3 + \Delta T$$

Model Generation Flow*

Input: **Injection rates for all traffic classes (λ), Network topology, Routing algorithm**

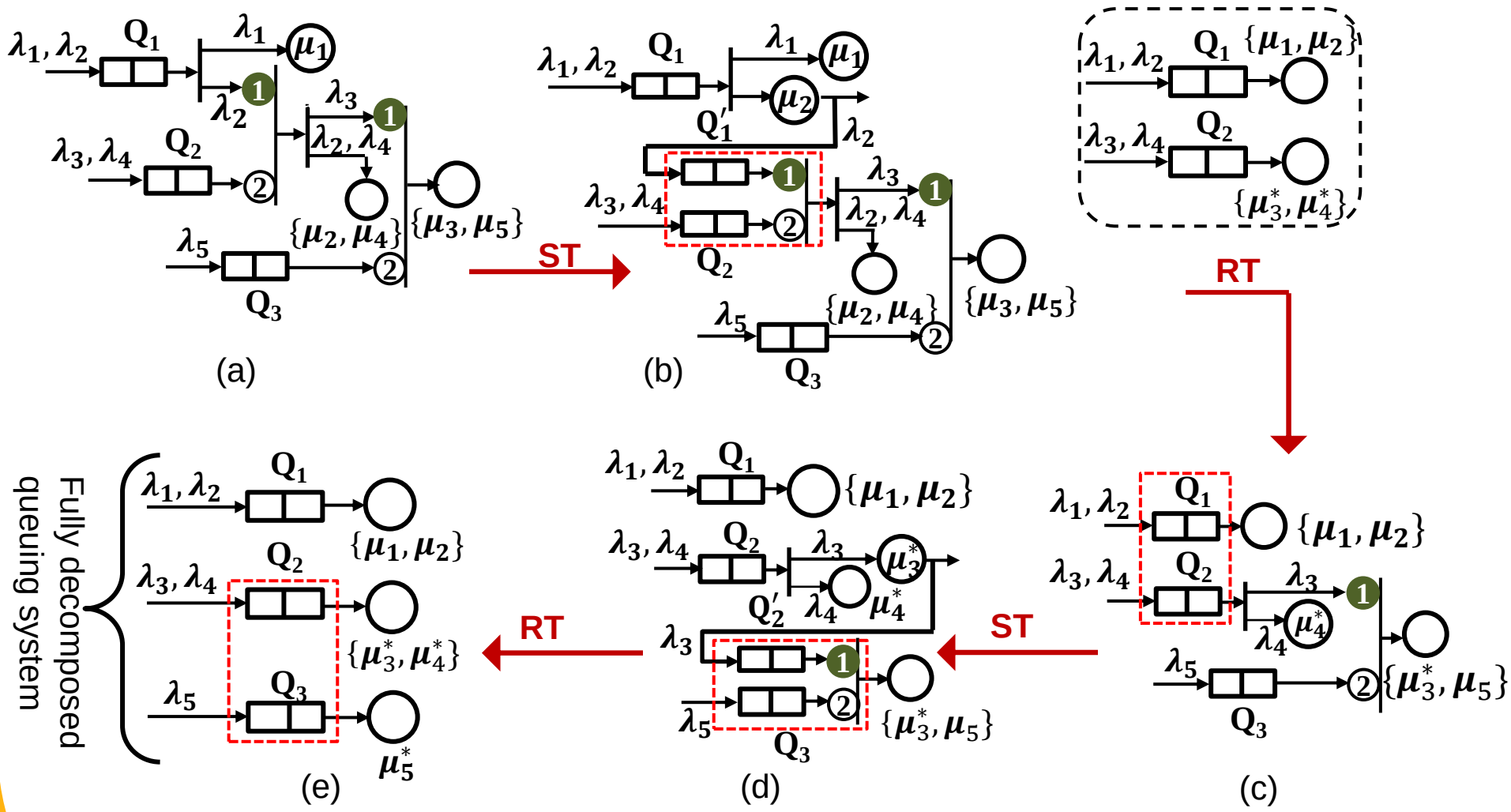
Output: **Waiting time for all traffic classes**

For each Queue and traffic class:



*A graphical illustration of a representative example given in the next slide

A Representative Example



μ_j^*: modified service rate of class-j

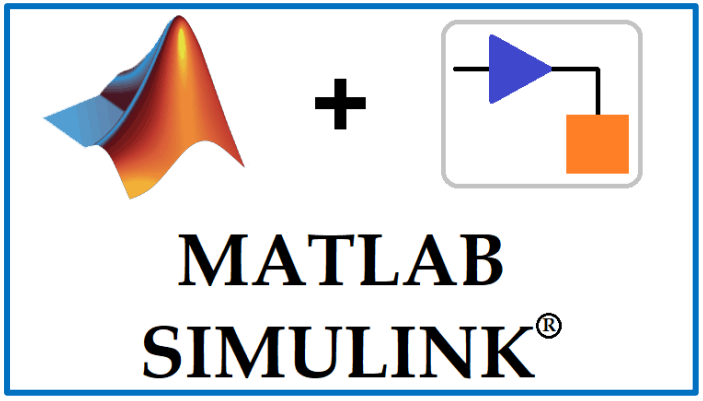
ST → Structural Transformation RT → Service Rate Transformation

Outline

- Need for analytical fabric model generation
- Background
  - Networks-on-chip (NoC) used in industry
  - Prior work on NoC performance analysis
  - Prior work on queuing networks
- Proposed network transformations
- **Experimental results**
- Conclusion and future work

Experimental Setup

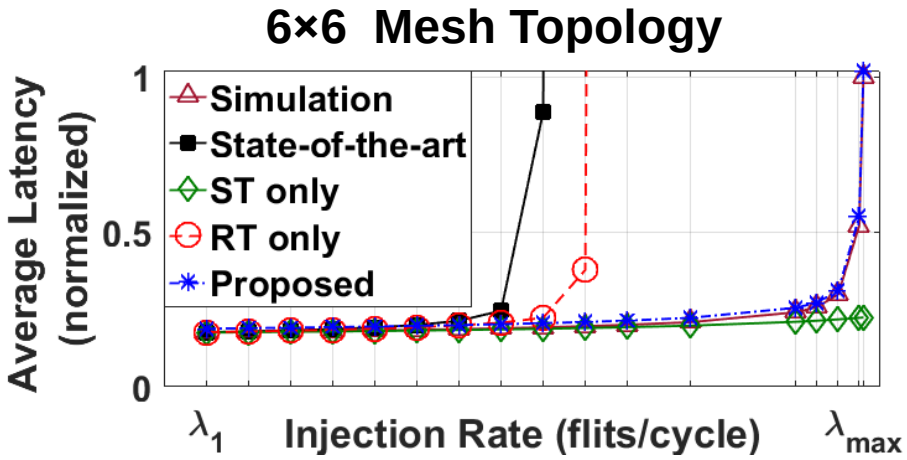
- **We evaluated the proposed analytical models on**
  - Ring
  - Mesh
- **Simulation parameters**
  - Simulation length: 10M cycles
  - Warm-up period: 5000 cycles
- **Traffic load**
  - Sweep from a very light load to λ_{max}
  - λ_{max} is the injection rate at which the maximum server utilization is 1

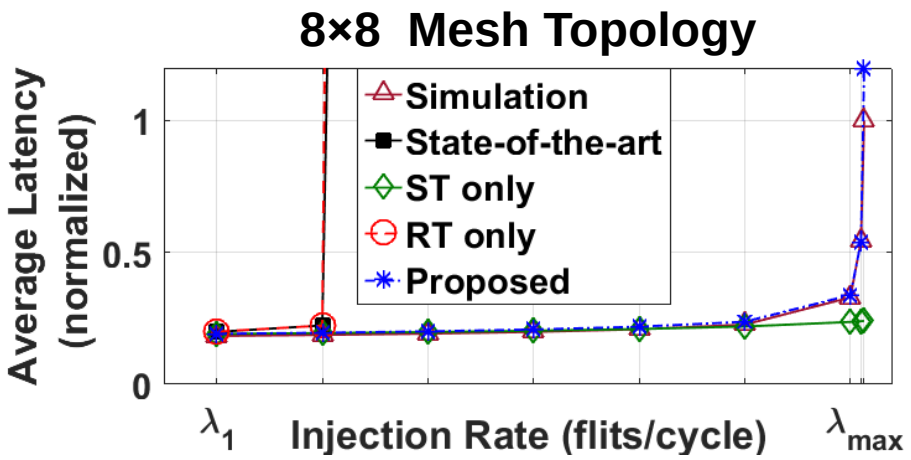


xPLORE

Evaluation on xPLORE: Mesh Topology

- Traffic pattern is all to all with YX routing
- Injection rate for each source destination pair is equal

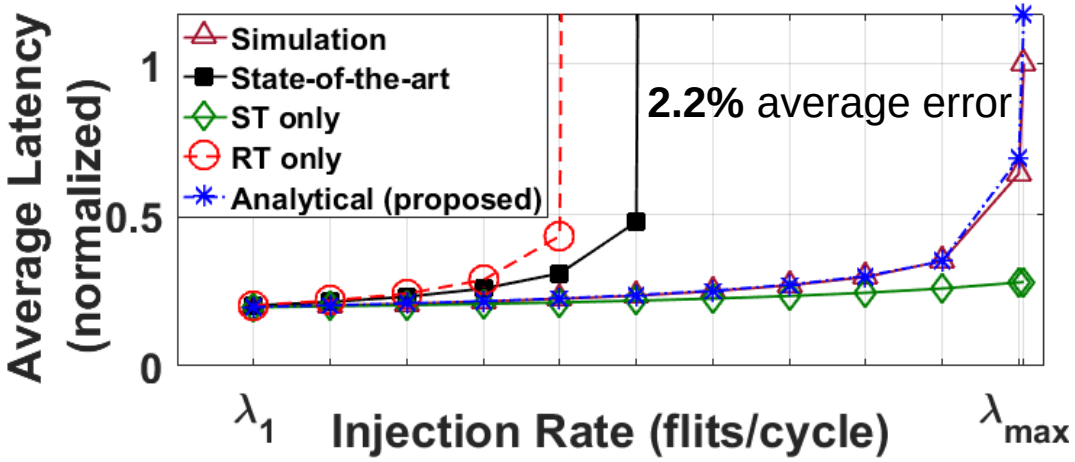




- Achieve less than 4% error compared to simulation
- Proposed analytical models are 2-3 orders of magnitude faster than simulation models

Verification with Intel® Xeon® Scalable Server

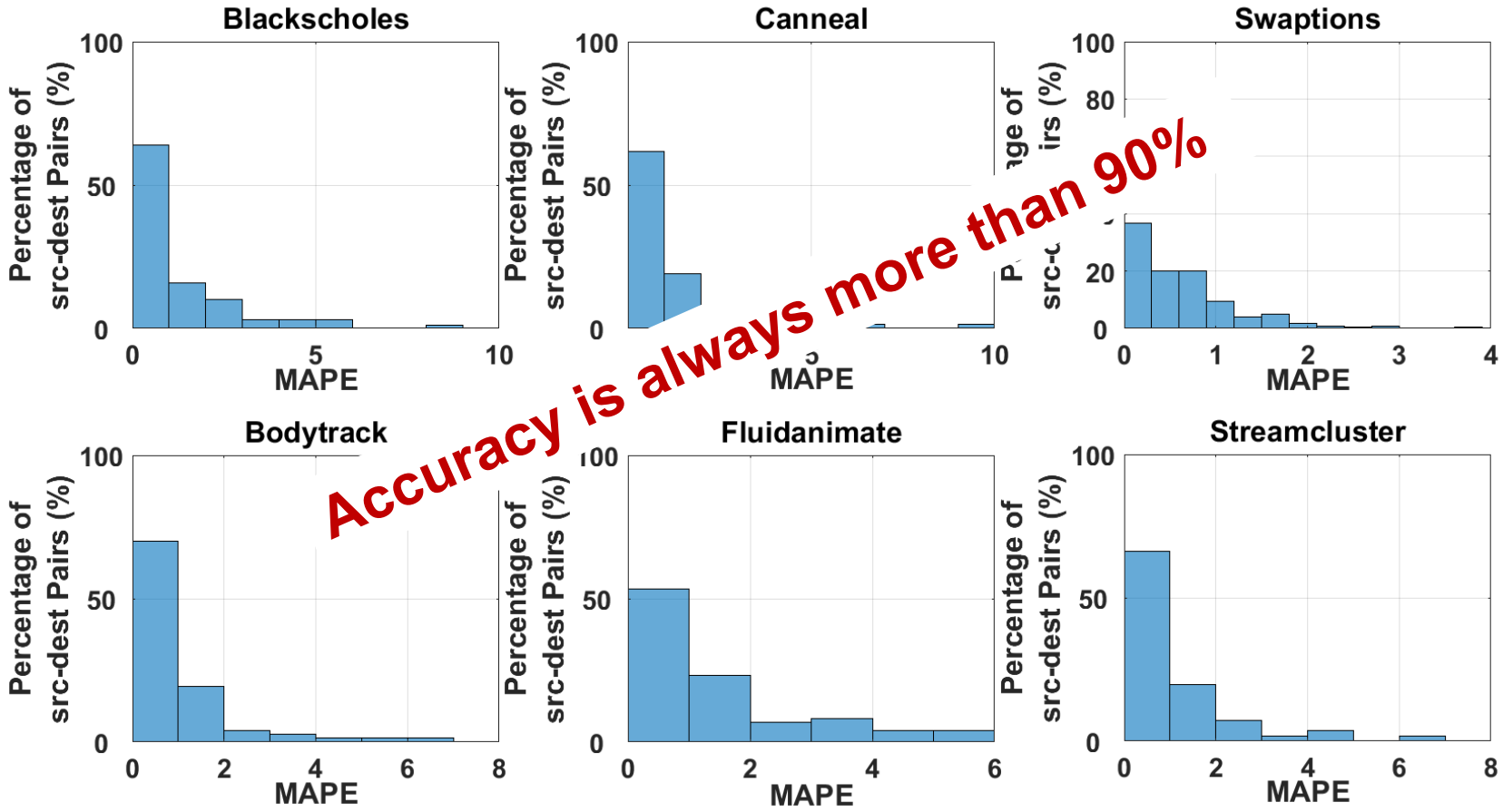
- **One variation of scalable Intel® Xeon®**
  - 26 tiles with Core + LLC, 2 memory controllers on a 6x6 mesh
- **Synthetic injection**
  - All cores send packets to all caches
- **Validated latencies for cache-coherency flow**



	Model Accuracy (%)	
LLC Hit Rate (%)	Address Network	Data Network
100	98.8	93.9
50	97.7	98.1
0	97.7	98.0

Evaluation with Real Applications

- **Collected real app traces from gem5 in Full System mode**
  - Applications (16-threaded) from PARSEC suite
  - Average statistics over 1M cycles



Outline

- Need for analytical fabric model generation
- Background
  - Networks-on-chip (NoC) used in industry
  - Prior work on NoC performance analysis
  - Prior work on queuing networks
- Proposed network transformations
- Experimental results
- **Conclusion and future work**

Conclusion and Future Work

- **Industrial NoCs use priority-based routers**
- **Most NoC performance analysis techniques assume fair arbitration**
- **Priority-based models in literature do not consider multiple traffic classes**
- **Presented the first technique that handles both priority and multiple traffic classes**

Machine learning (ML)
- Queuing network is high dimensional and non-linear problem
- Unknown model structure
- Generated ML models may incur high runtime overhead

Increasing flexibility

Increasing accuracy

First principles – queuing theory
- No existing solution
- Solutions exist for non-priority aware network

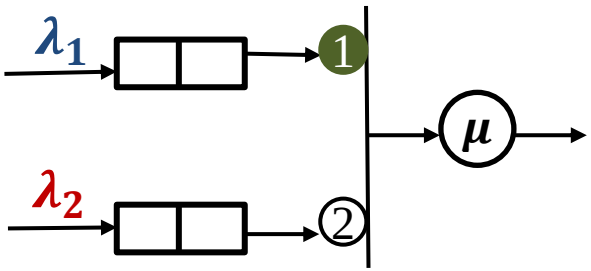
Complex queuing network

THANK YOU

BACK-UP

Residual Time for Basic Priority Queue (Geo/G/1)

Priority Rule: (1>2)

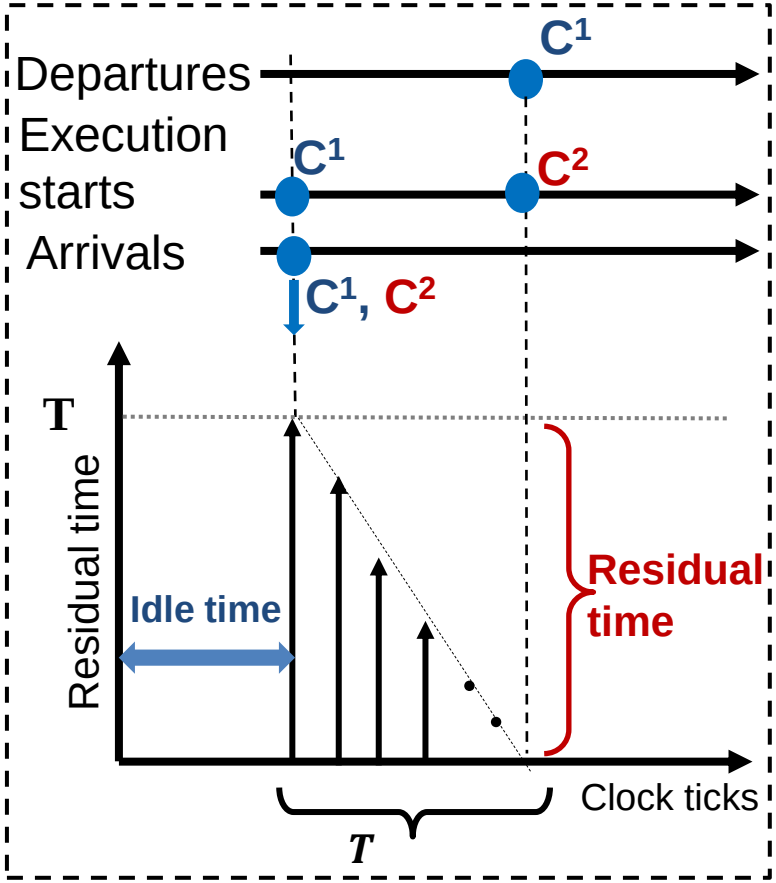


$$R_1 = \frac{1}{T_{tot}} \sum_{i=1}^{M_1(T_{tot})} \left(\sum_{\tau=0}^{T-1} \tau \right) + \frac{1}{T_{tot}} \sum_{i=1}^{M_2(T_{tot})} \left(\sum_{\tau=0}^{T-1} \tau \right)$$

$$= \frac{1}{2}\rho_1(T-1) + \frac{1}{2}\rho_2(T-1)$$

$$R_2 = \frac{1}{T_{tot}} \sum_{i=1}^{M_1(T_{tot})} \left(\sum_{\tau=0}^{T} \tau \right) + \frac{1}{T_{tot}} \sum_{i=1}^{M_2(T_{tot})} \left(\sum_{\tau=0}^{T-1} \tau \right)$$

$$= \frac{1}{2}\rho_1(T+1) + \frac{1}{2}\rho_2(T-1)$$



T = service time

$M(T_{tot})$ = total no. of tokens arrived during time interval T_{tot}

τ = intermediate variable for sum

Residual Time for Single Queue Node

- **Residual time (R):** delay of serving the next token due to the remaining service time for currently processed token
- Arrival distribution is Geometric
  - $P\{X = k\} = p(1-p)^{k-1}$

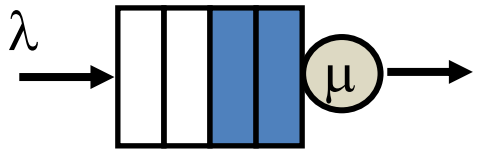
$$R_{avg} = \frac{1}{T_{tot}} \sum_{i=1}^{M(T_{tot})} \left(\sum_{\tau=0}^{T_i - 1} \tau \right)$$

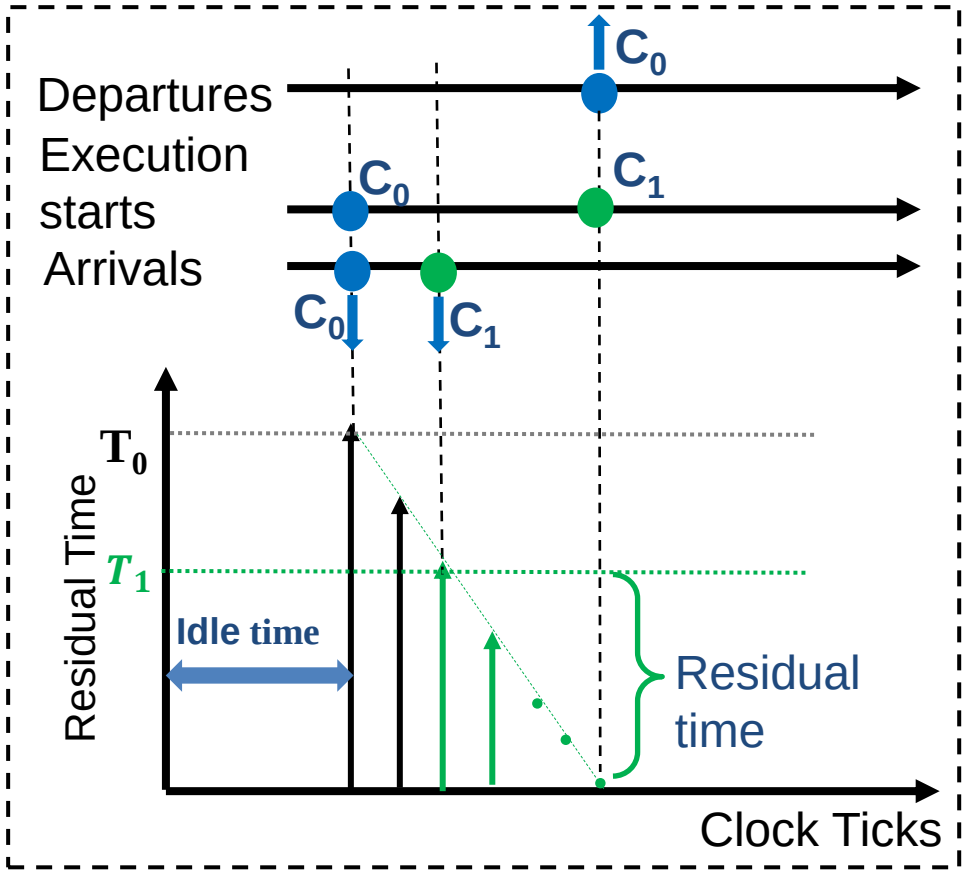
$$= \frac{1}{T_{tot}} \sum_{i=1}^{M(T_{tot})} \frac{1}{2} T_i (T_i - 1)$$

$$= \frac{M(T_{tot})}{T_{tot}} \frac{\sum_{i=1}^{M(T_{tot})} \frac{1}{2} T_i (T_i - 1)}{M(T_{tot})}$$

$$= \frac{1}{2} \lambda \left(\overline{T^2} - \overline{T} \right) \text{ (for Geo/G/1)}$$

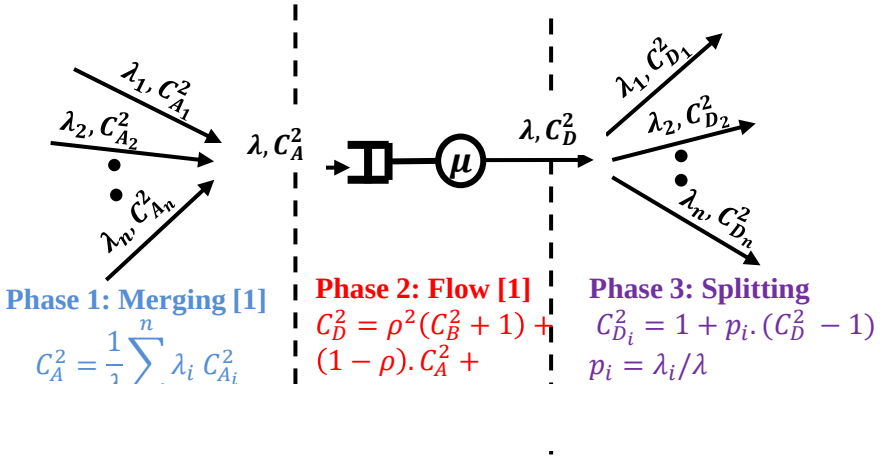
$$= \frac{1}{2} \rho (T - 1) \text{ (for Geo/D/1)}$$





Decomposition Method

- **Can handle complex network with multi-class traffic <u>but limited to non-priority networks</u>**
- **Decompose the queuing network into individual queues of type G/G/1**
- **Approximate input/output traffic distribution of these G/G/1 queues using analytical expressions**

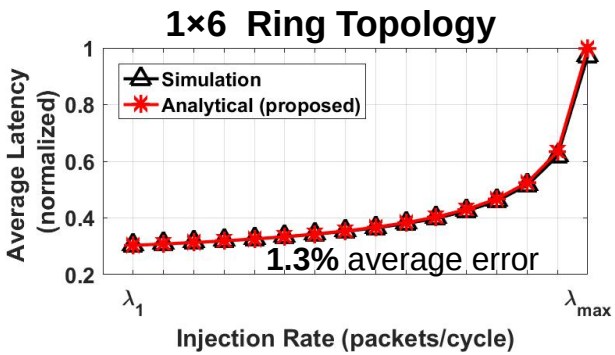


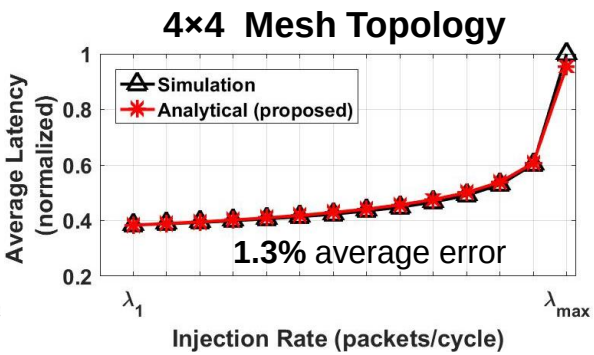
Coefficient of variation = $\frac{standard\ deviation}{mean}$

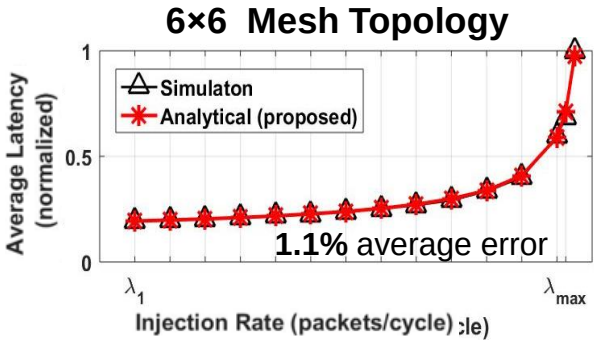
[1] Pujolle, Guy, and Wu Ai. "A solution for multiserver and multiclass open queueing networks." *INFOR: Information Systems and Operational Research* 24.3 (1986): 221-230.

Evaluation on Simulink

- Simulation models have been built in **Simulink based** on priority aware network architecture
- We observe **less than 2% error** between simulation and analysis for rings and mesh



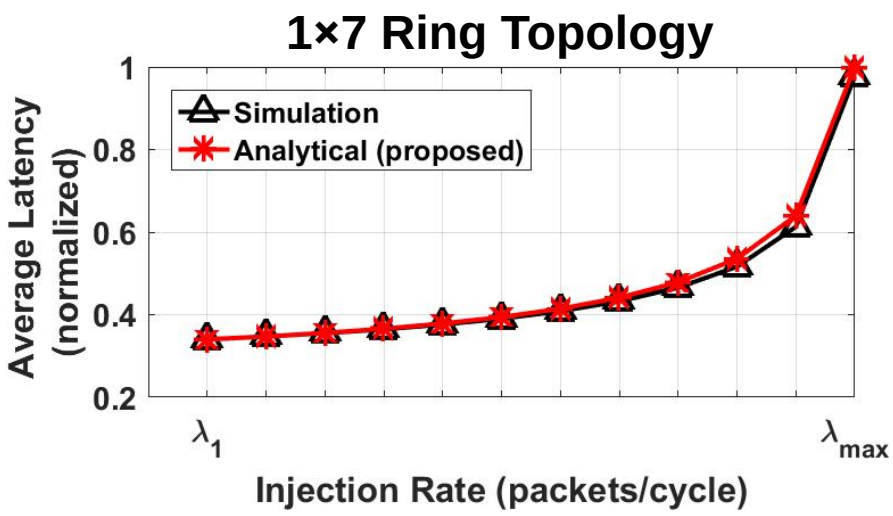


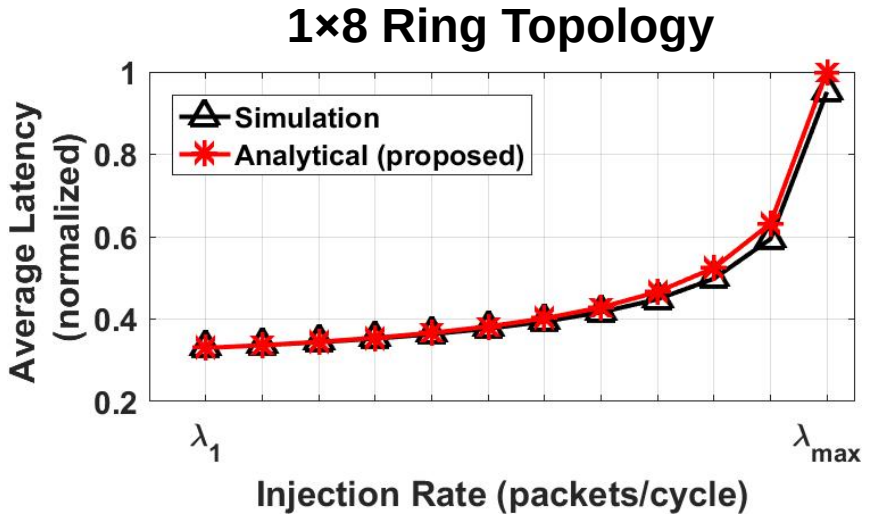


- Traffic pattern is all to all (i.e. each node is sending tokens to all nodes) with YX routing
- Injection rate for each source destination pair is equal

Evaluation on xPLORE: Ring Topology

- We observe less than 2% error between simulation and analysis for rings

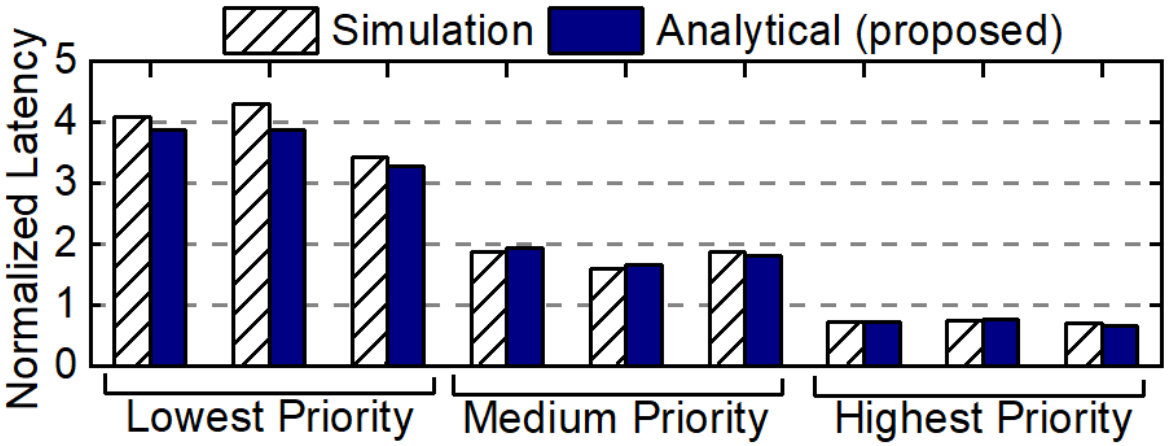




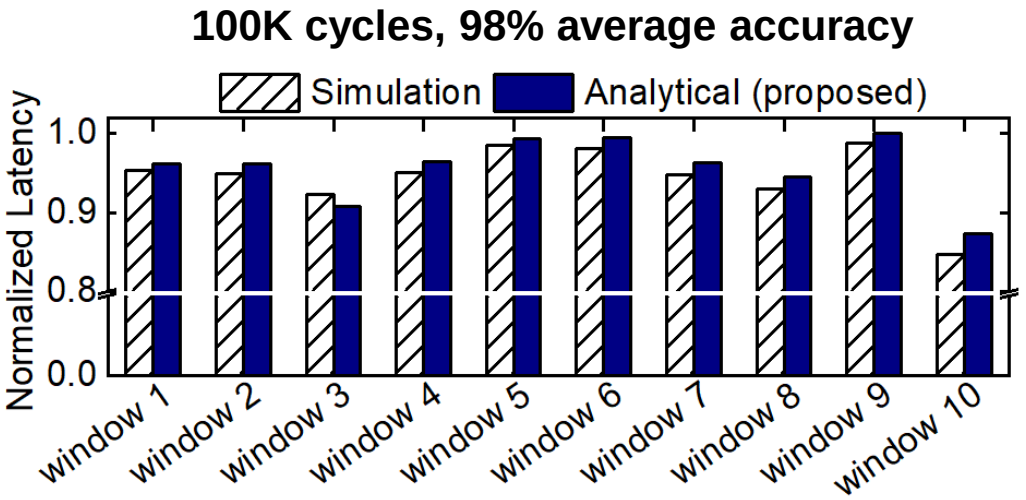
- xPLORE is a System-C based simulator for priority aware NoCs
- Traffic pattern is all to all (i.e. each node is sending tokens to all nodes) with YX routing
- Injection rate for each source destination pair is equal

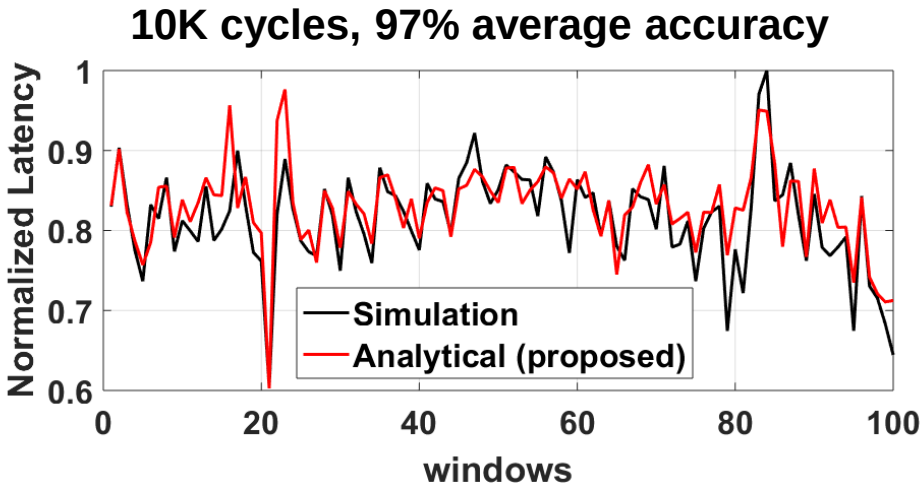
Per Class Latency Comparison for Intel® Xeon®

- **Average accuracy for lowest priority class is 91%**
  - Medium and highest priority class show 99% accuracy



Real Application (Streamcluster) Finer Grained





Simulation Time Comparison

Full-System Simulation Time with 4×4 Mesh NoC		
With Garnet 2.0 Simulation	With Proposed Analytical Models	Speedup
12466 s	4986 s	2.5×