Observations in Darwin

Jon Schuringa, Arben Lila, Günter Remsak, Harmen van As

IEEE 802 plenary meeting
Vancouver, B.C.
July 8-12, 2002

Contents

- Introduction
- Four Scenarios
  - Large Rings
  - Staggered Bottlenecks
  - On-Off Sources
  - High/low bandwidth flows
- Conclusions

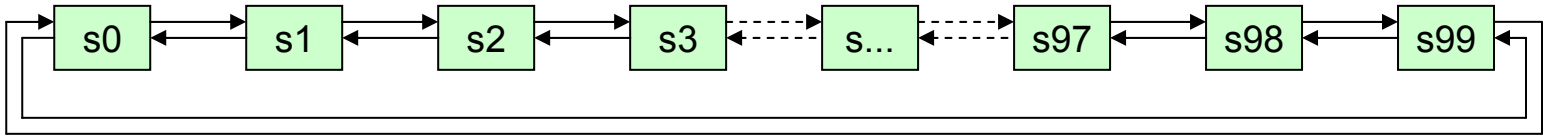
Other simulation results appended

Introduction

- Darwin
  - Simulation model according to latest fairness description (cls09_fairness_701.pdf)
- Cyclic Queuing Multiple Access (CQMA)
  - New name for IKN proposal
  - Presented at the 7th European Conference on Networks & Optical Communications, Darmstadt, Germany, June 18-21, 2002
  - CQMA-1: Optimal scheduler
  - CQMA-2: Simpler, scalable, sub-optimal in some scenarios
    - All CQMA simulations in this presentation are done with CQMA-2

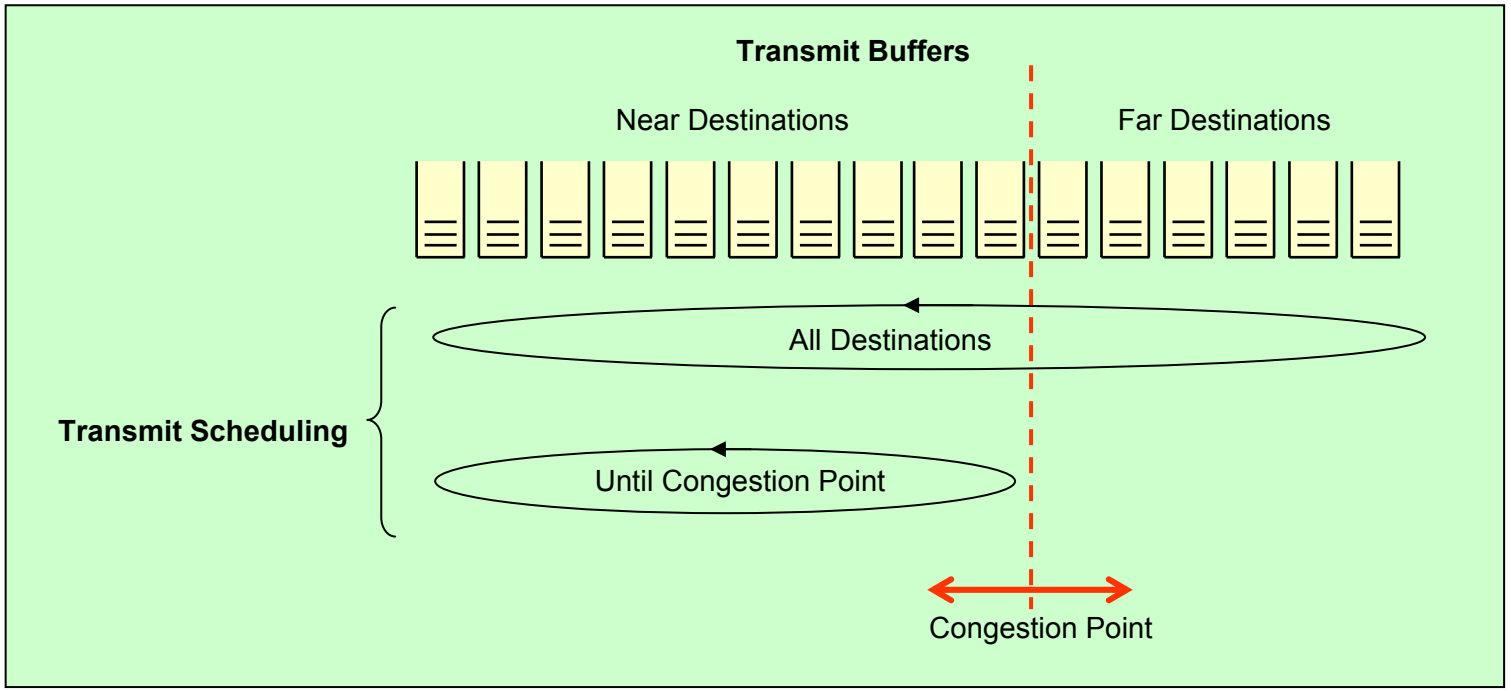
Large Rings (1)

- Scenario:
  - 100 Stations
  - Dual OC-12 Ring, all traffic routed over clockwise ringlet
  - Low priority traffic
  - Uniform saturated, all to all traffic (100*99 flows)



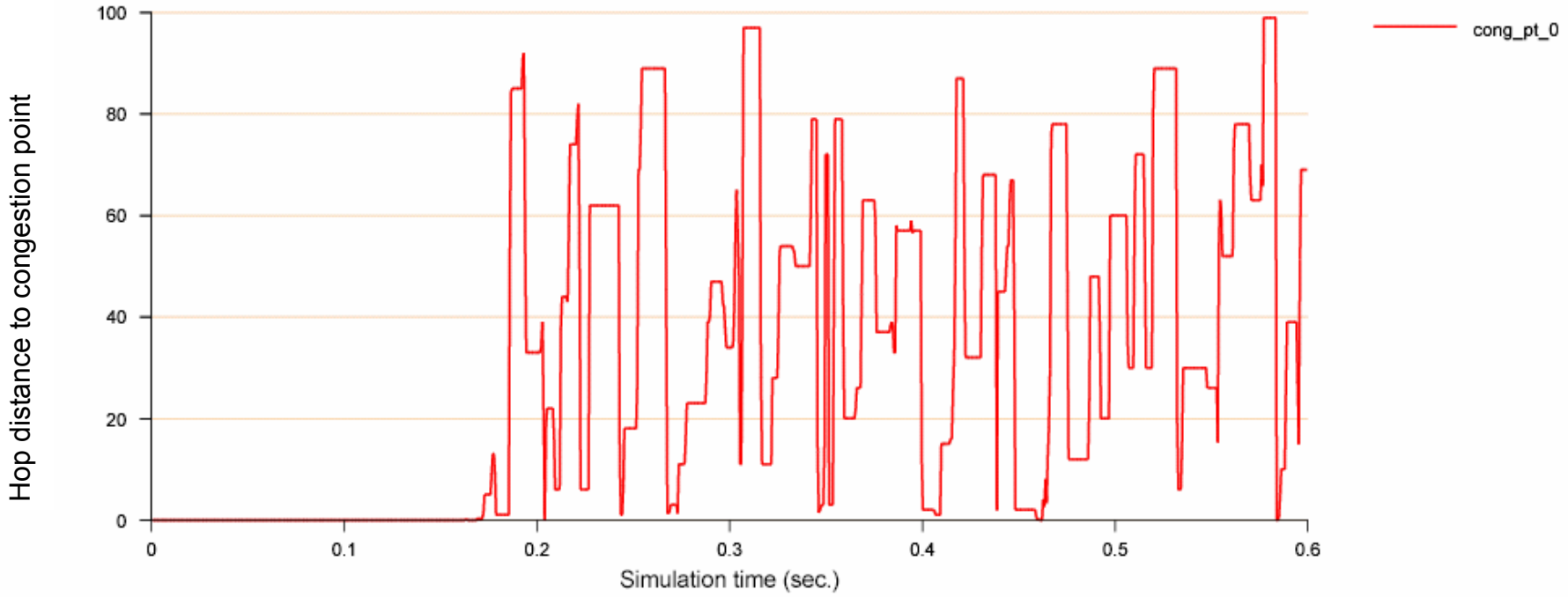
Large Rings (2)

- Darwin single choke implementation:
  - Station can be in two states if allowed to send:
    - To all destinations (as long as *add_rate_congested* < *allowed_rate_congested*)
    - To destinations until congestion point
  - Congestion point can move



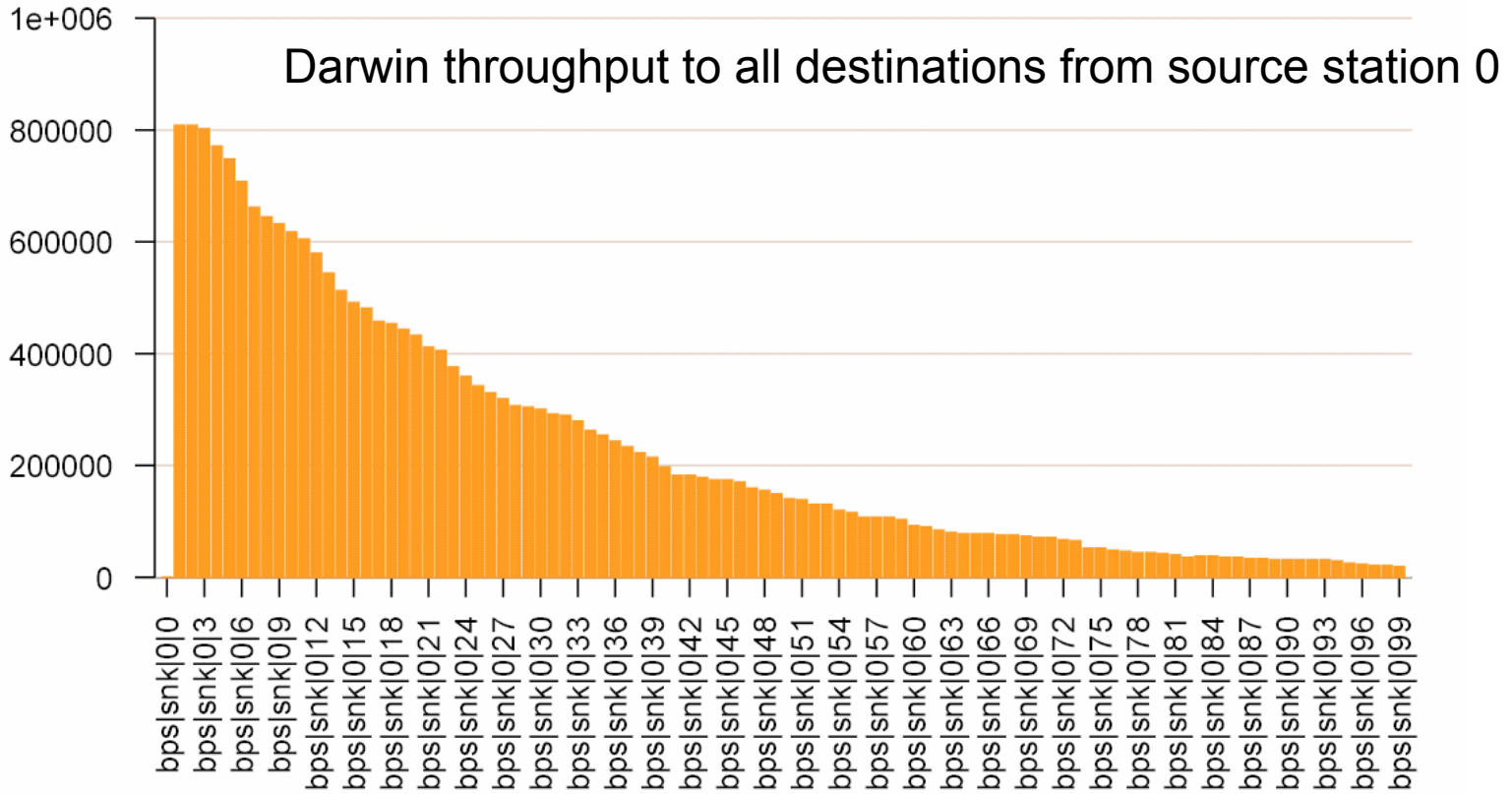
Large Rings (3)

Hop distance to congestion point observed by node 0



- Congestion point is not stable and can be anywhere

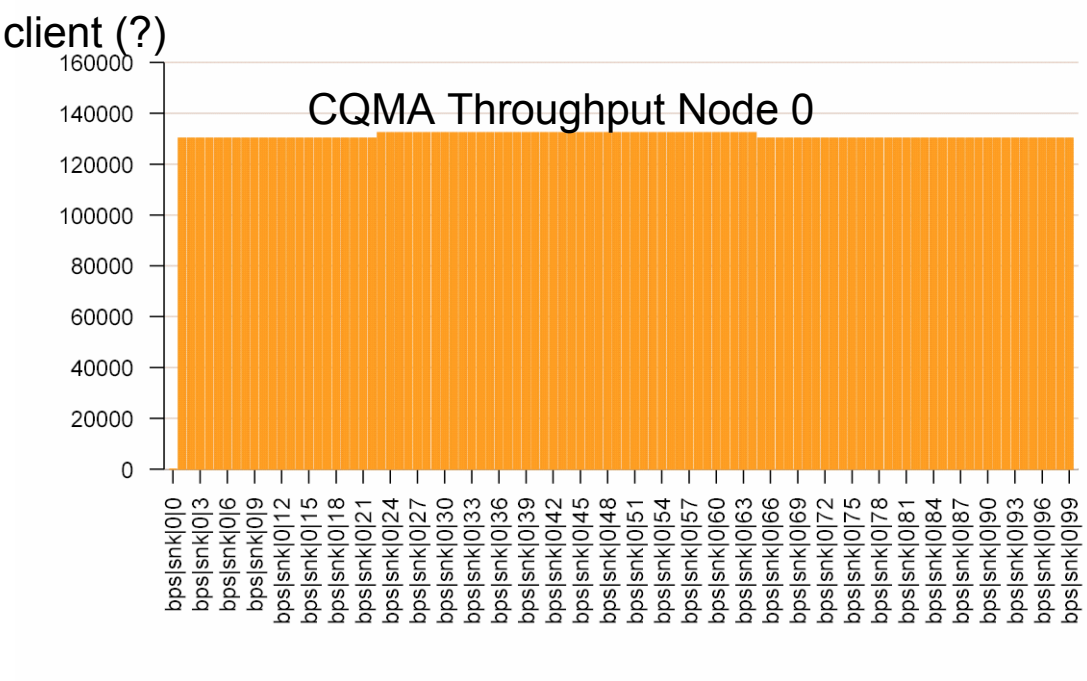
Large Rings (4)



- All source stations get a fair share, but this is not equally divided among the destinations.

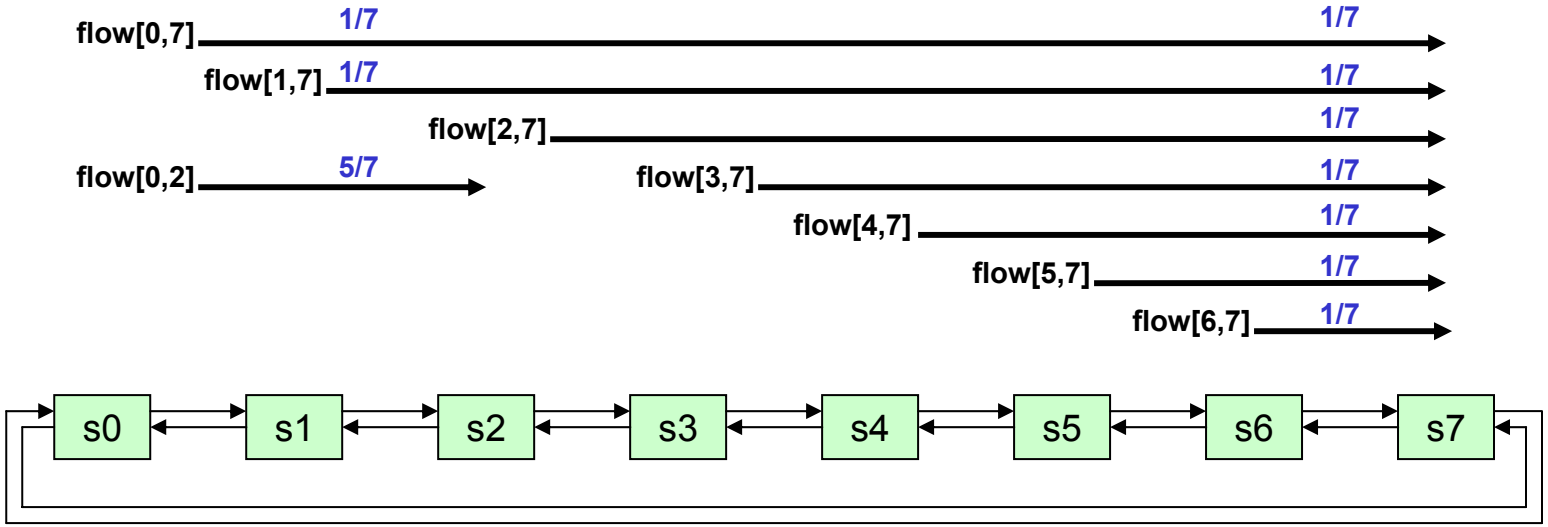
Large Rings (5)

- Nearby destinations get more bandwidth, far away destinations get almost nothing
  - Effect is even stronger in aggressive mode
  - Effect is already visible for rings with about 25 stations
  - Queuing discipline (e.g., round robin, WFQ,…) in MAC client is not the main reason
- Solution:
  - Multi choke with “Intelligent” client (?)
  - Other fairness algorithms



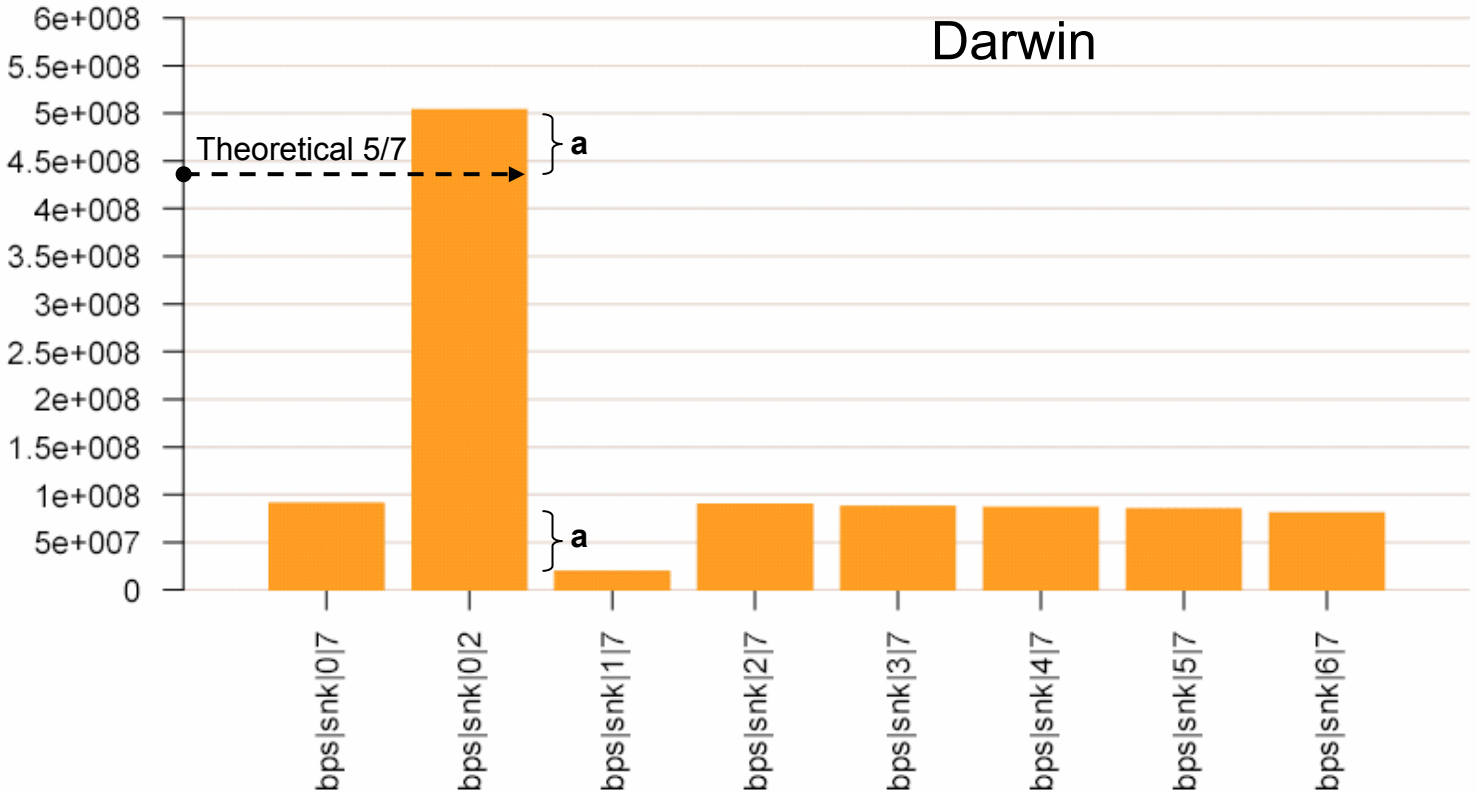
Staggered Bottlenecks (1)

- Two bottlenecks, link 6→7 and link 1→2
- Problem: Station 0 could see only the strongest bottleneck 6→7, thereby giving flow 0→2 too much (6/7) and flow 1→7 not enough bandwidth.



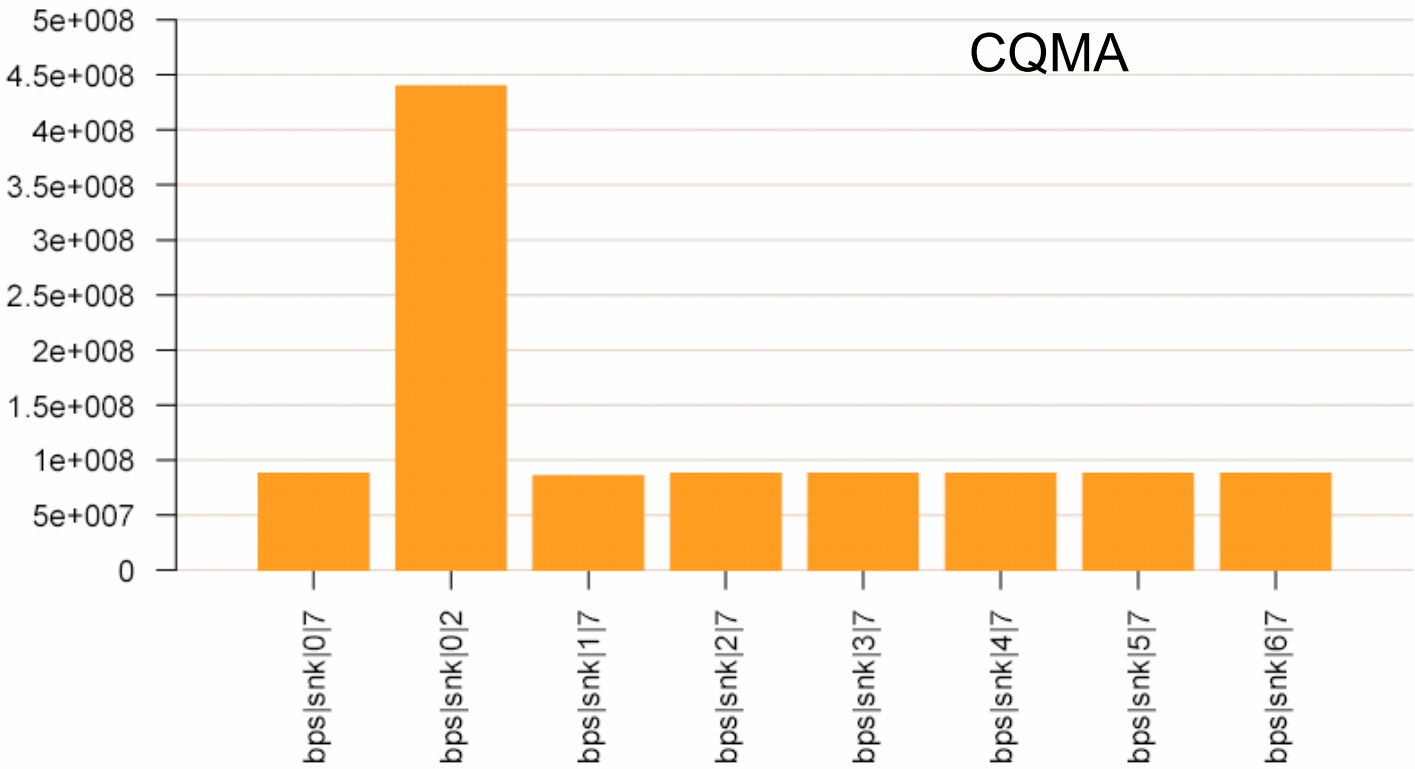
Staggered Bottlenecks (2)

- Strongest bottleneck is moving between link 6→7 and 1→2, causing a very small throughput between stations 1 and 7, too much between 0 and 2



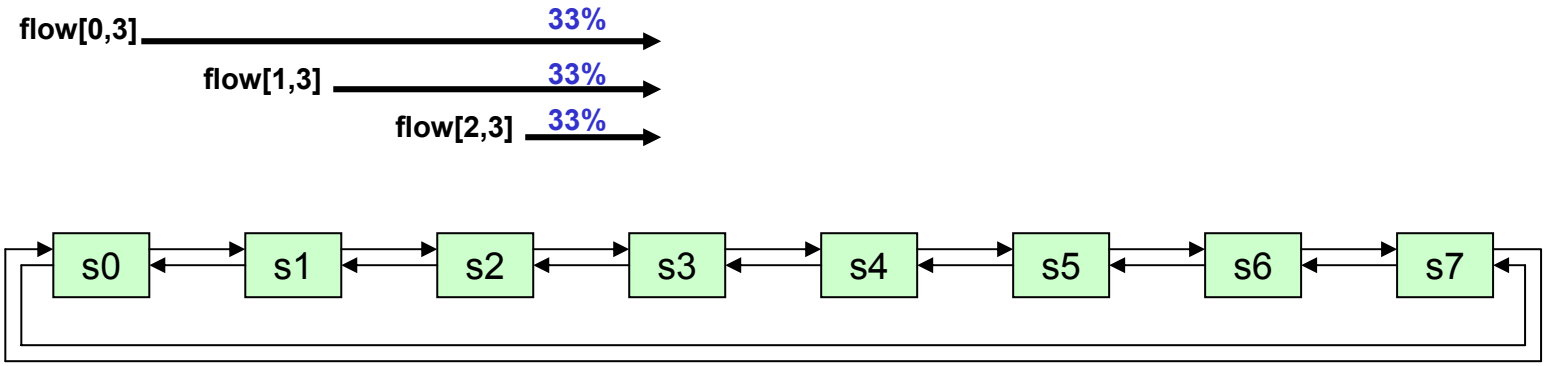
Staggered Bottlenecks (3)

- Multi choke, or other fairness algorithms could solve the problem
- CQMA shows correct behavior



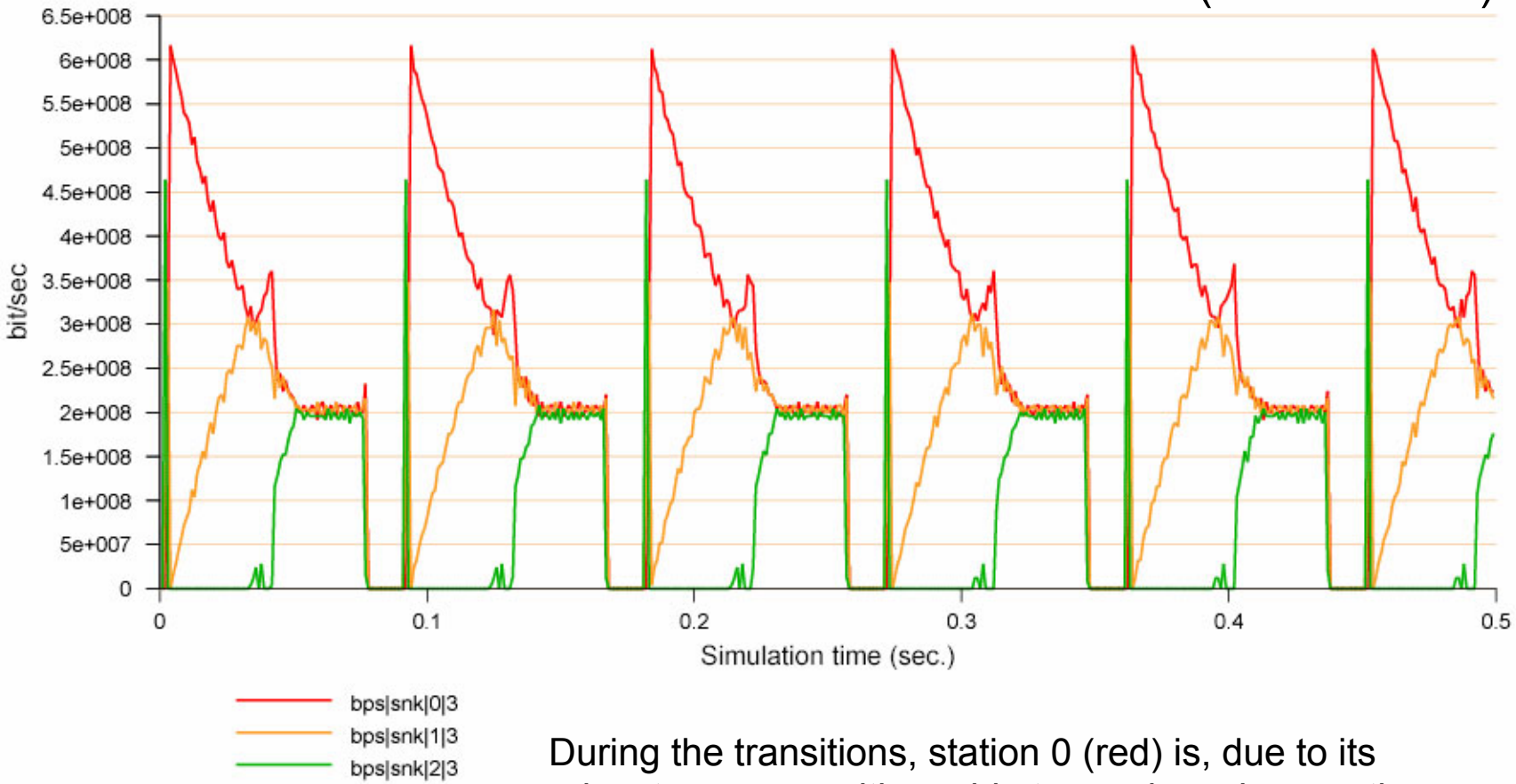
On-Off Sources (1)

- Three identical on-off flows
  - Same duration, start and end times
  - All stations want to send at 100% of the bandwidth, when in “on phase”
- During the on phase, all flows should get 33% of the available bandwidth



On-Off Sources (2)

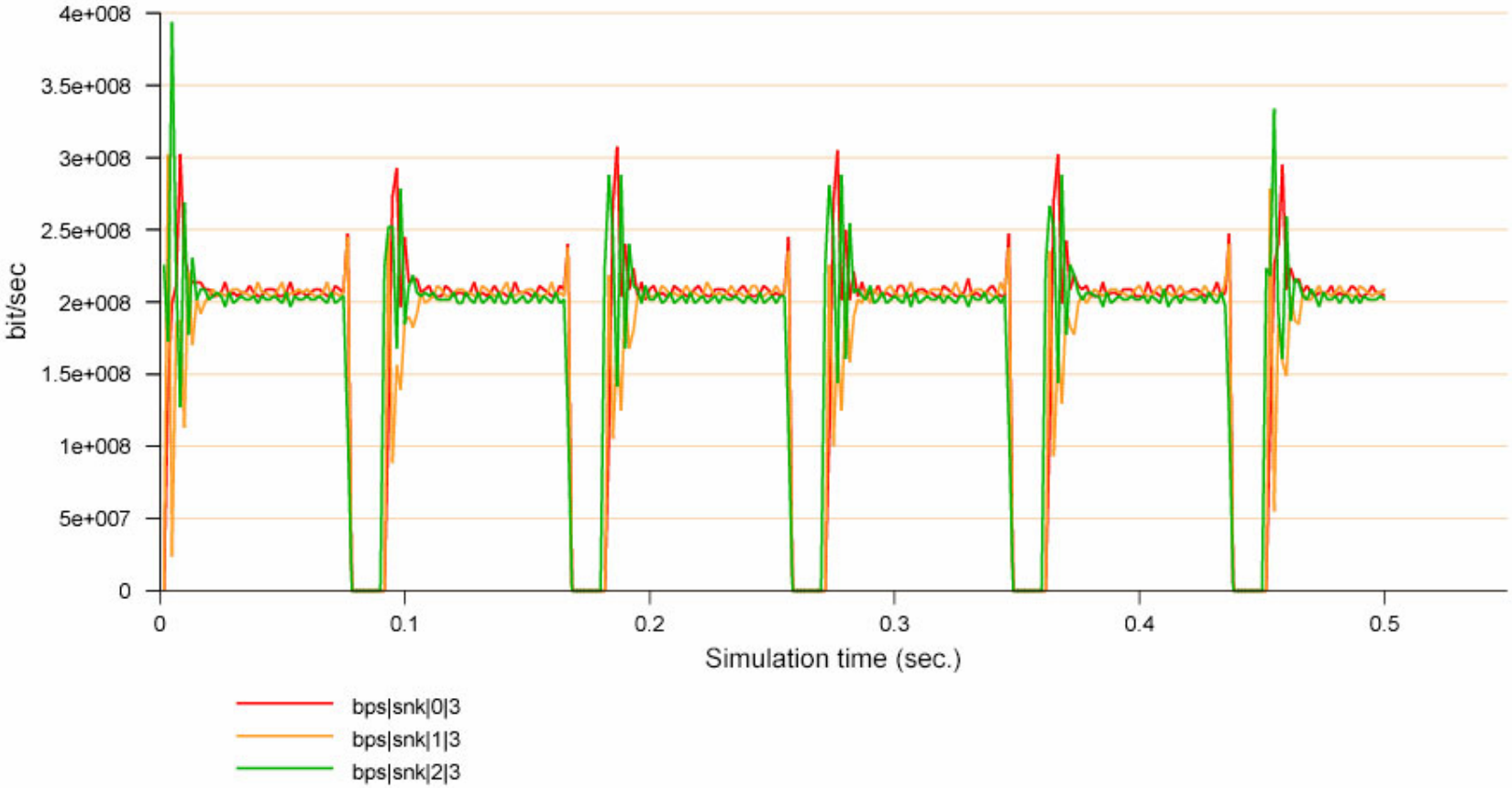
Darwin (conservative)



During the transitions, station 0 (red) is, due to its advantageous position, able to send much more than the other stations, causing unfairness.

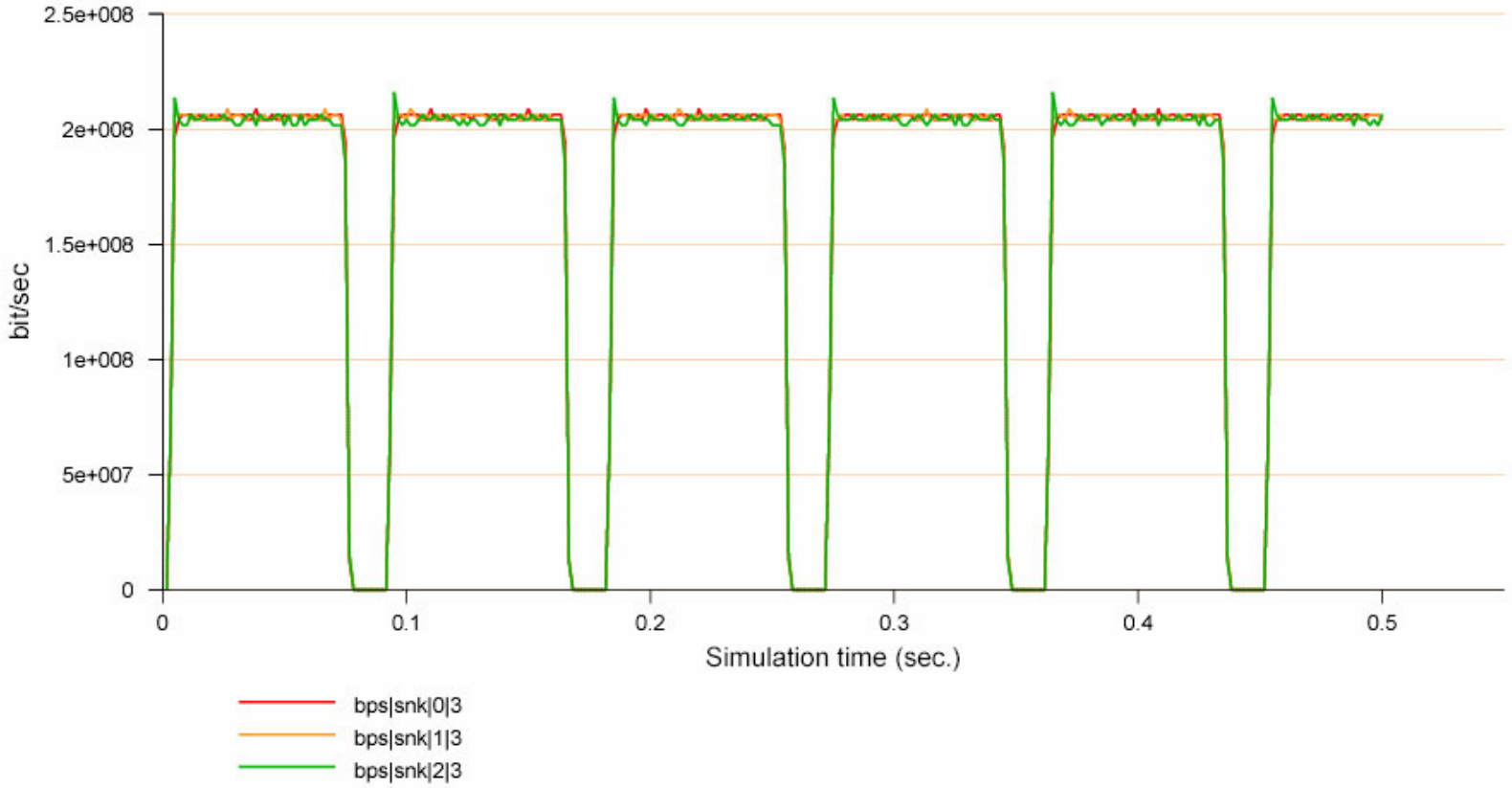
On-Off Sources (3)

Darwin (aggressive)



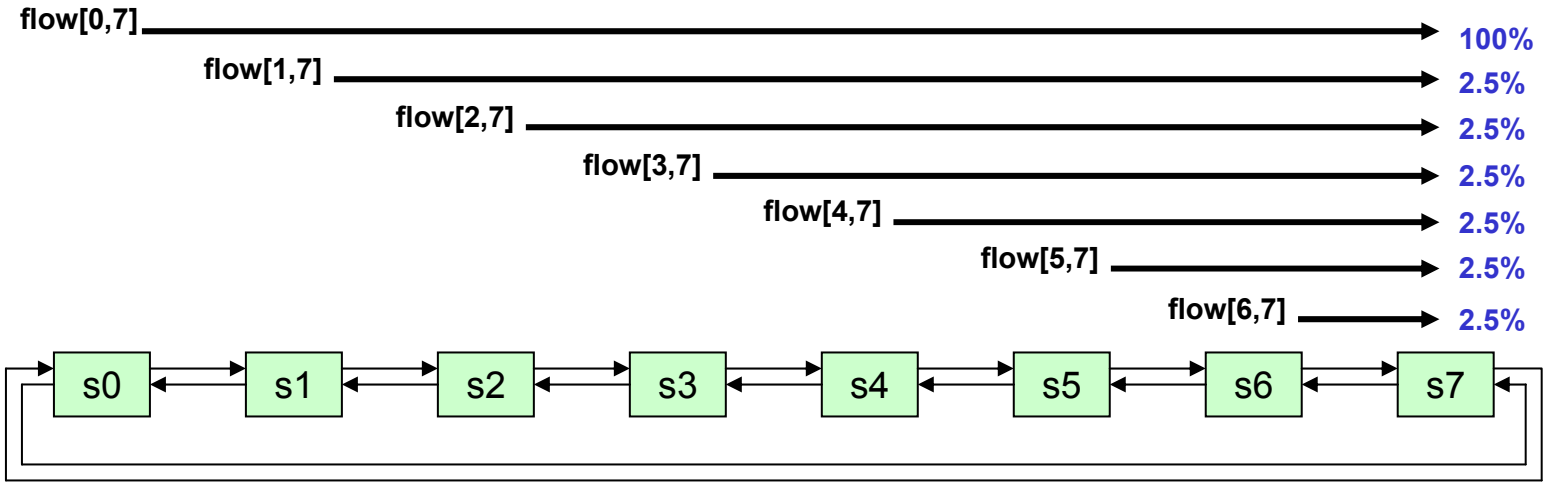
On-Off Sources (3)

CQMA



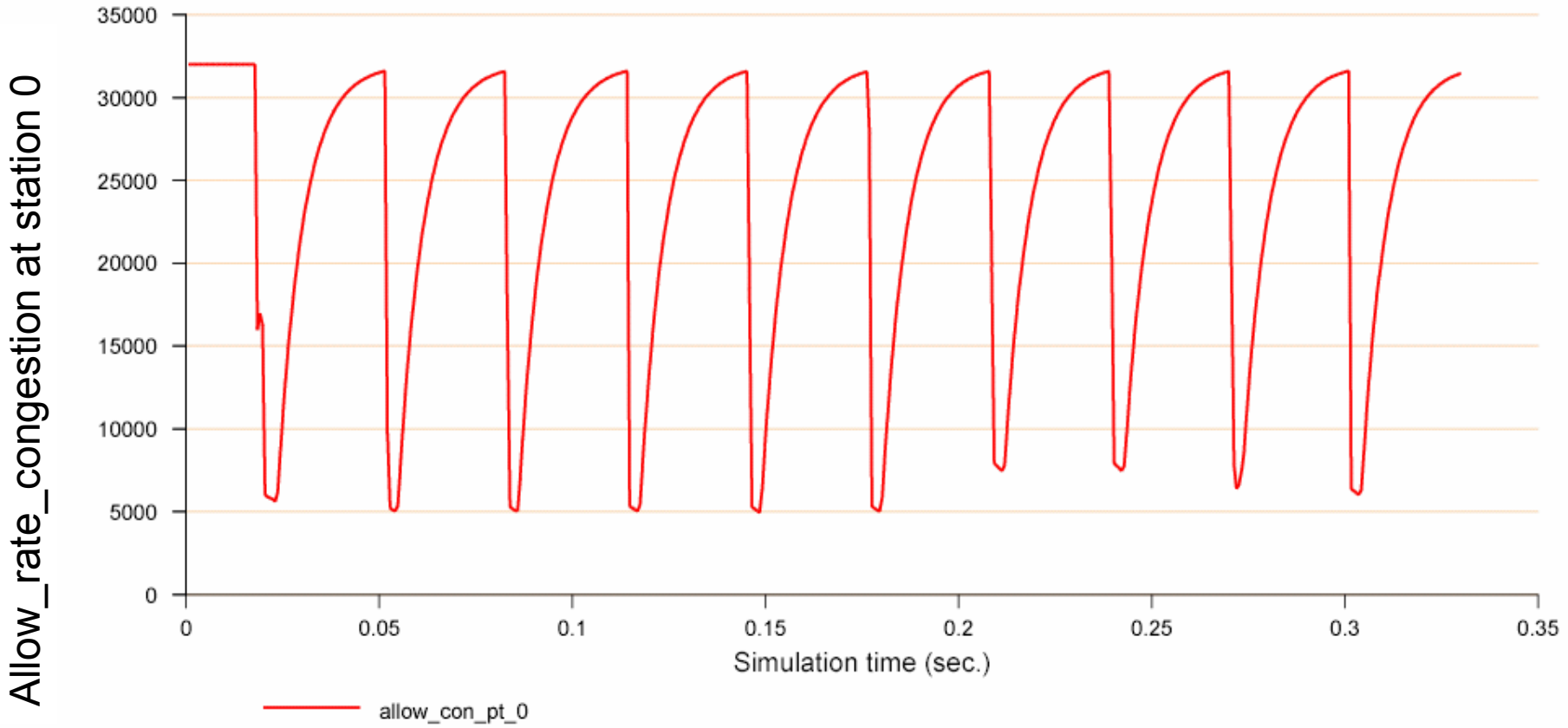
High- and low-bandwidth sources (1)

- Flow 0→7 wants full bandwidth, all other flows only 2.5%
- In theory, flow 0→7 should get 85%
- Problem:
  - Flow 0→7 is oscillating, causing throughput loss

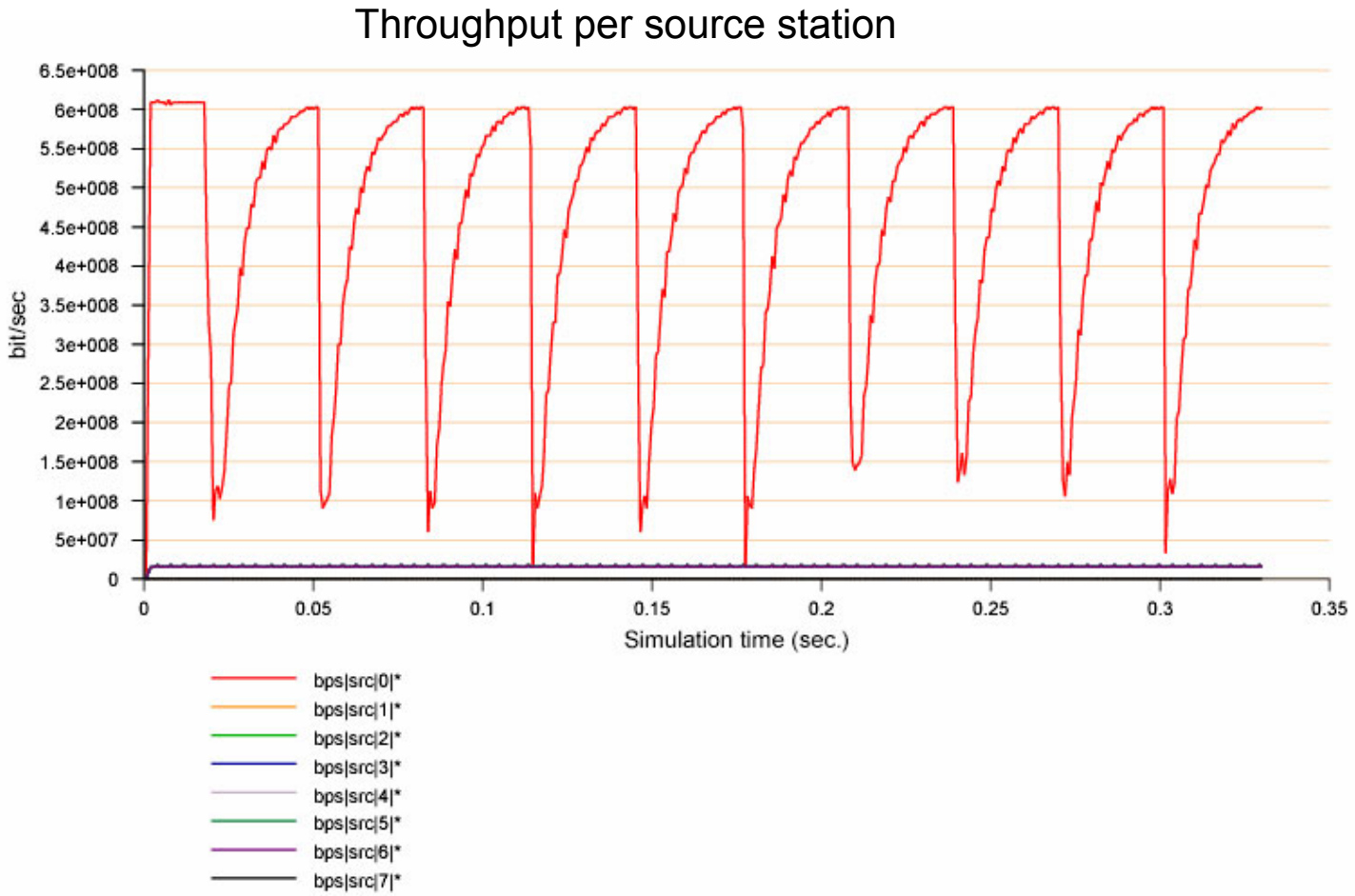


High- and low-bandwidth sources (2)

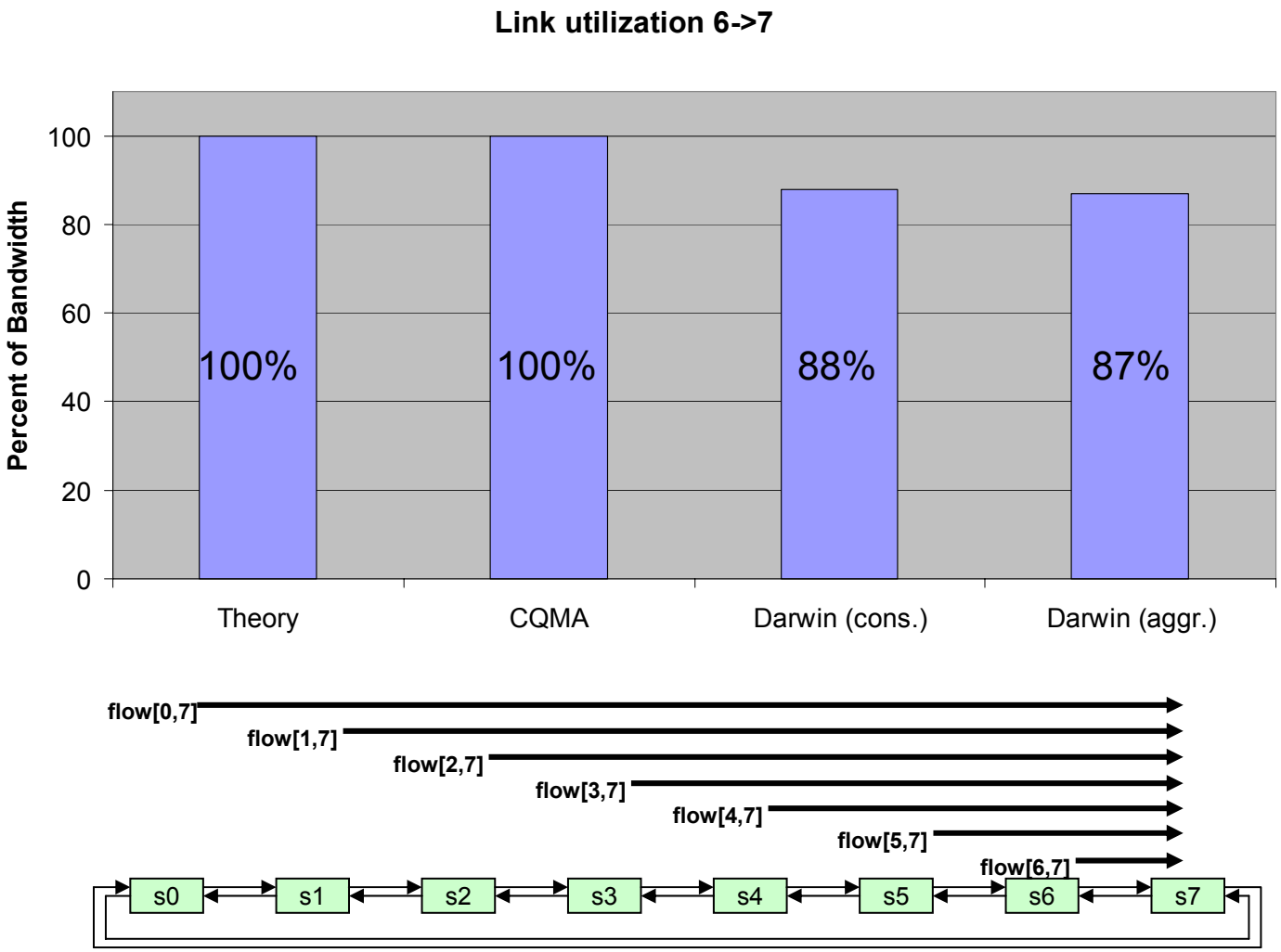
- Initial advertised rate is based on number of active stations



High- and low-bandwidth sources (3)



High- and low-bandwidth sources (4)



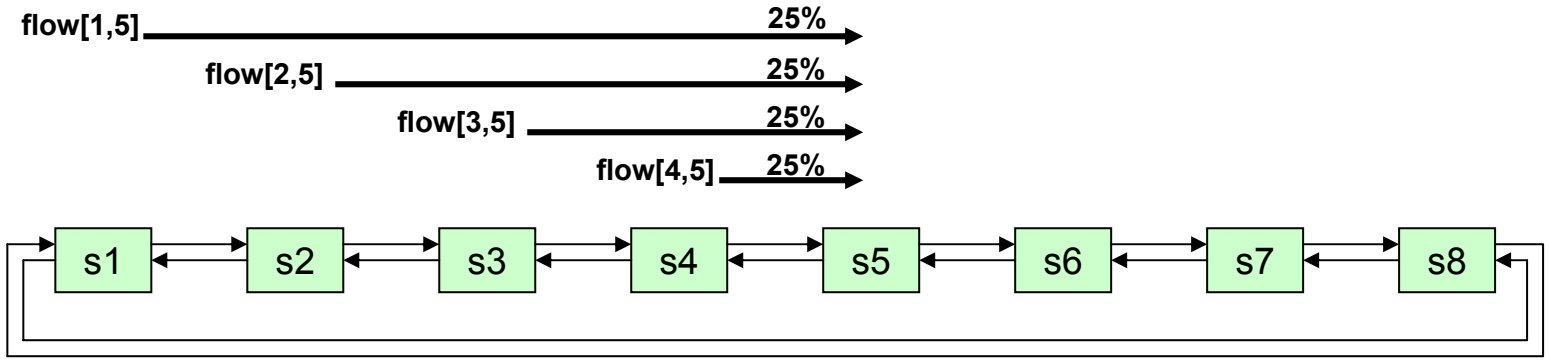
Conclusions

- Single- / Multi-choke
  - Single choke may result in unfairness
  - Single choke may achieve a lower ring utilization
  - Single choke does not work well in large rings (number of stations)
    - If multi choke solves these issues, why not make multi-choke mandatory (and move it from the MAC-client to the MAC)

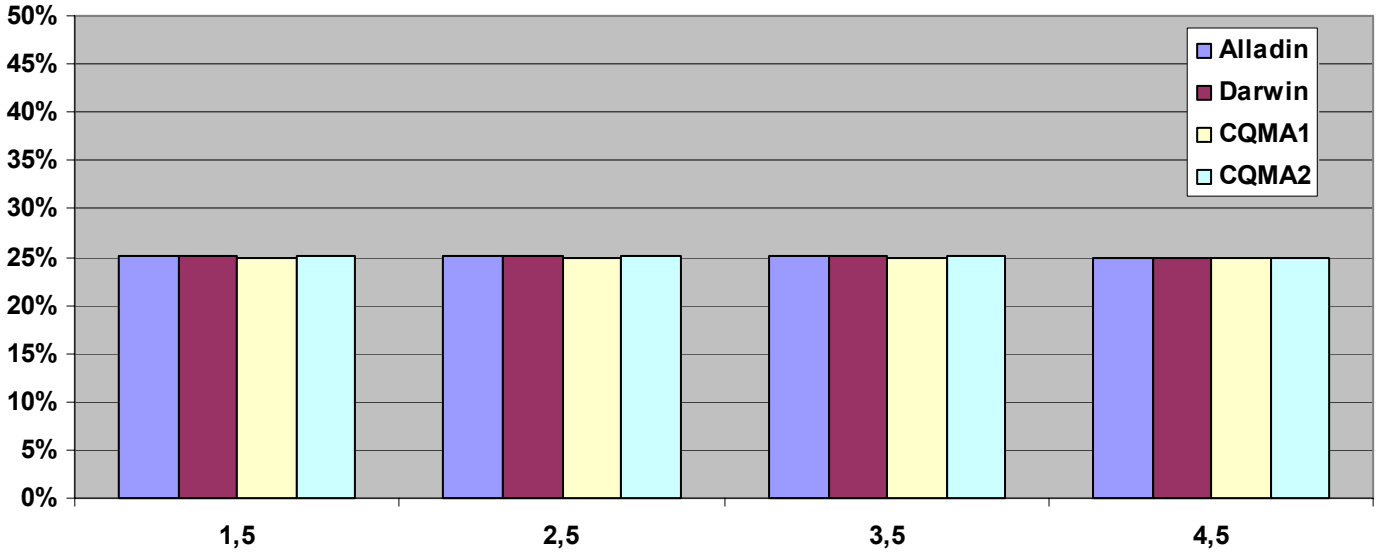
- Conservative / Aggressive mode
  - Scenarios exist where conservative is better and other where aggressive is better
    - How to decide which one when to use?

Other simulation results

Scenario 1: Parallel Parking Lot

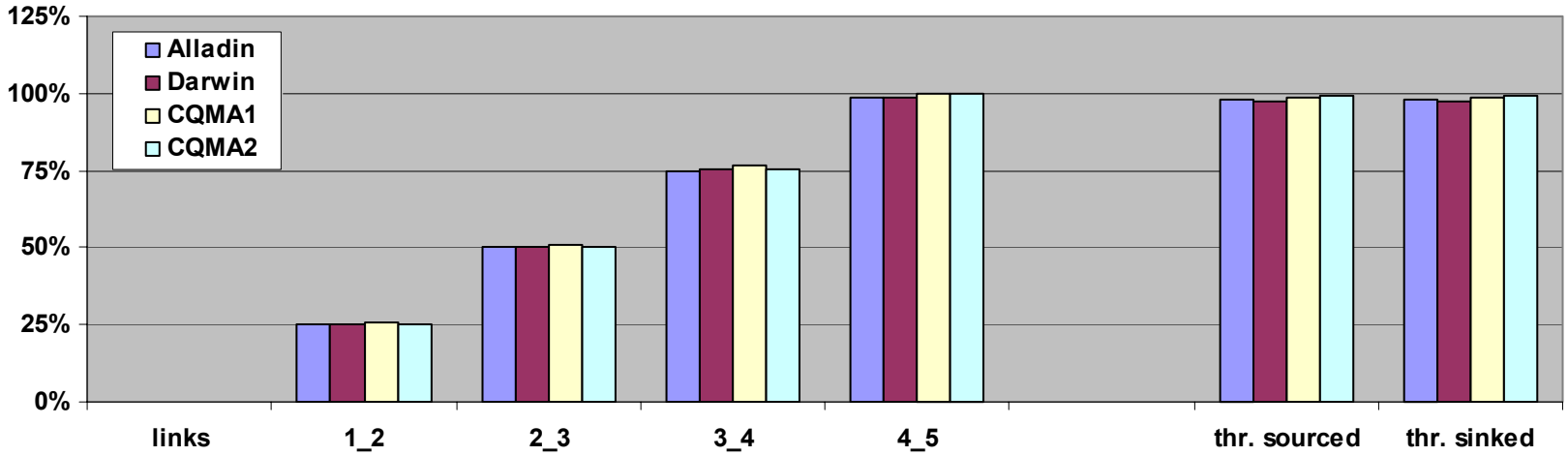


Flows

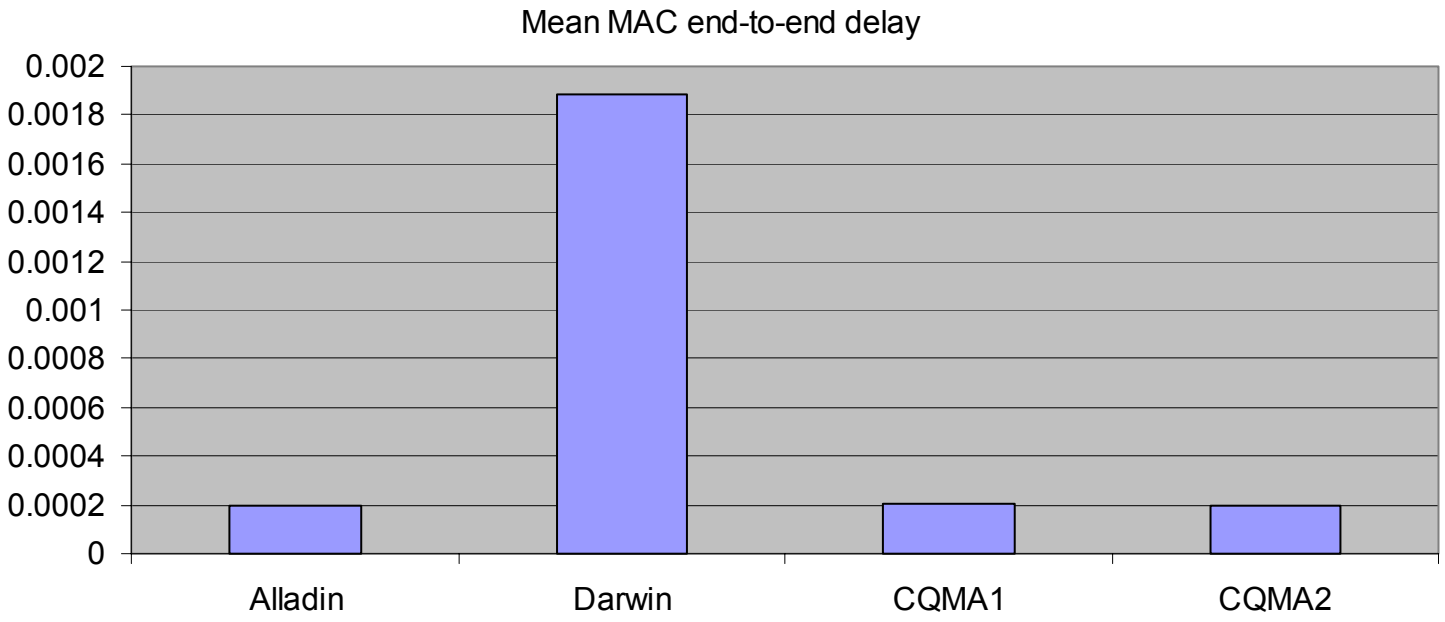


In this scenario we seem to have a fair use of bandwidth for each flow

Throughput

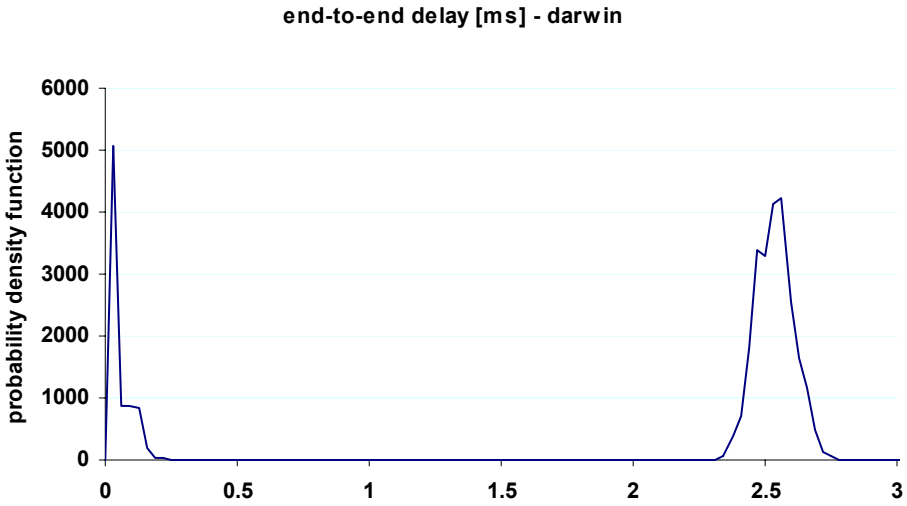

125%
100%
75%
50%
25%
0%
Alladin
Darwin
CQMA1
CQMA2
links
1_2
2_3
3_4
4_5
thr. sourced
thr. sinked

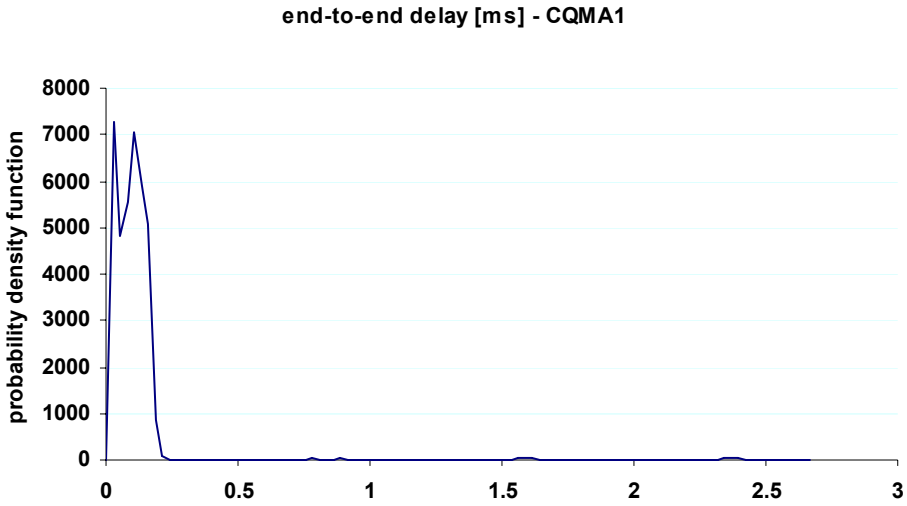
Delay

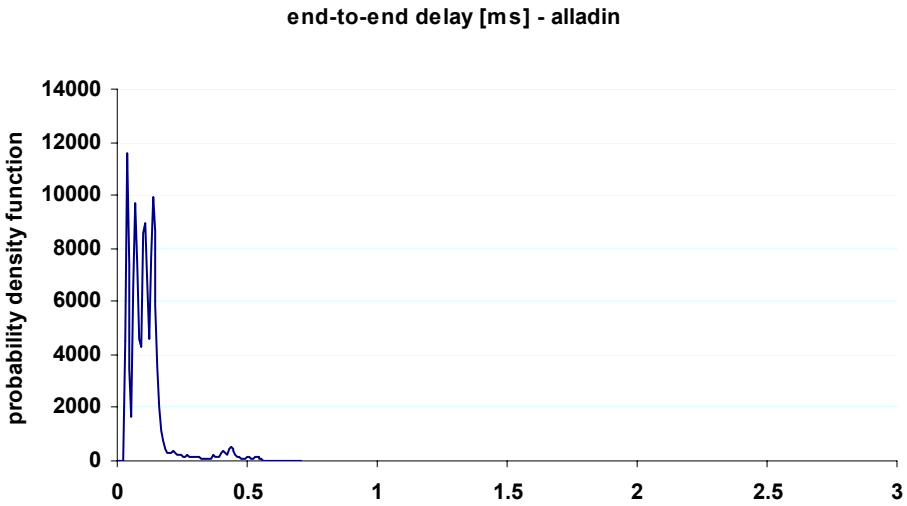


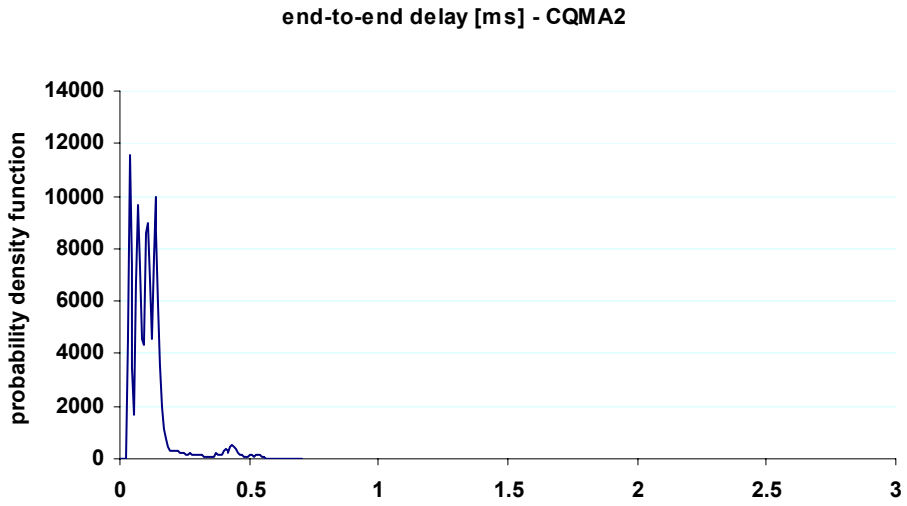
Delay for Darwin is higher than all other

Probability Density Function (PDF)
MAC end to end Delay

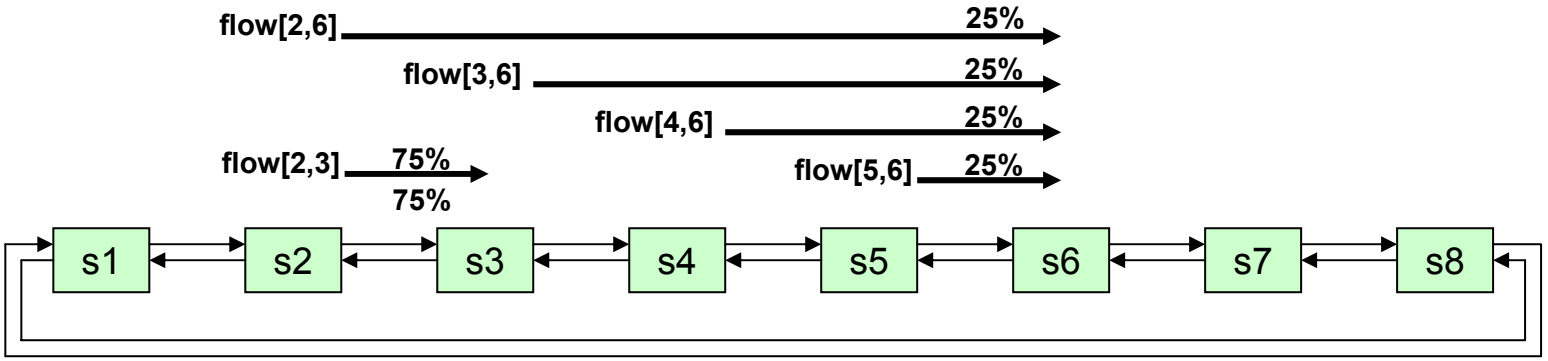




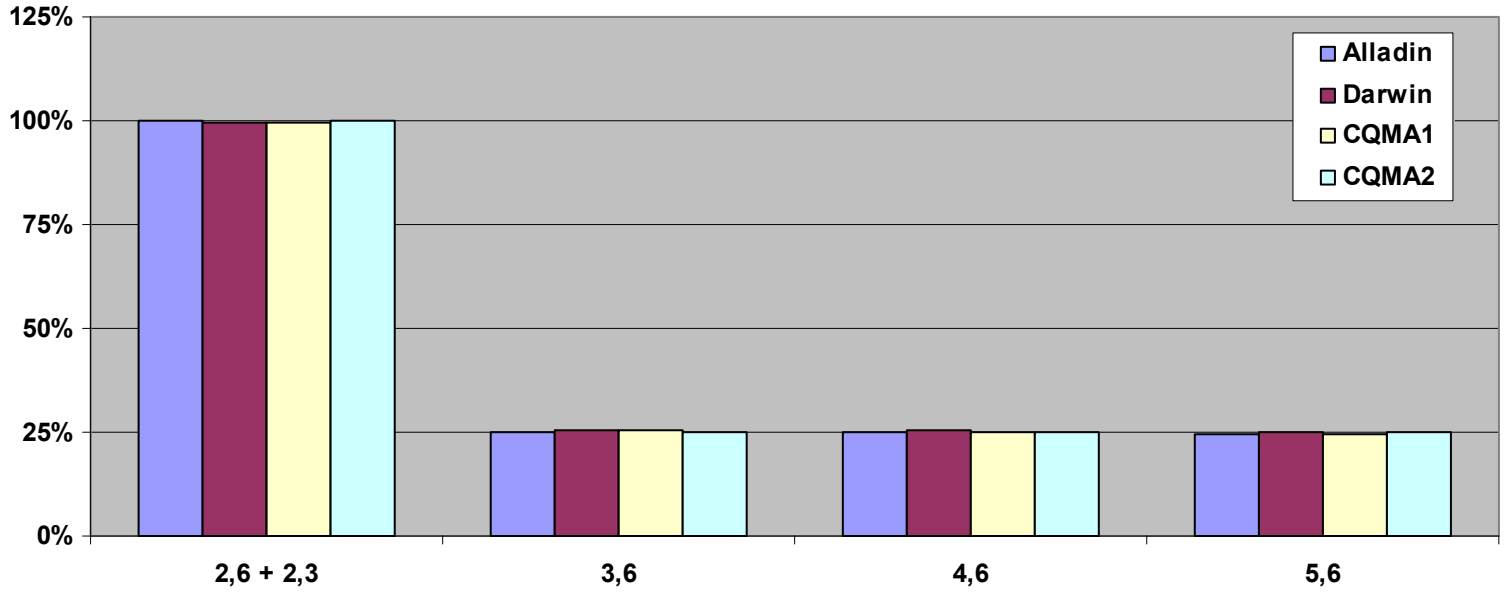




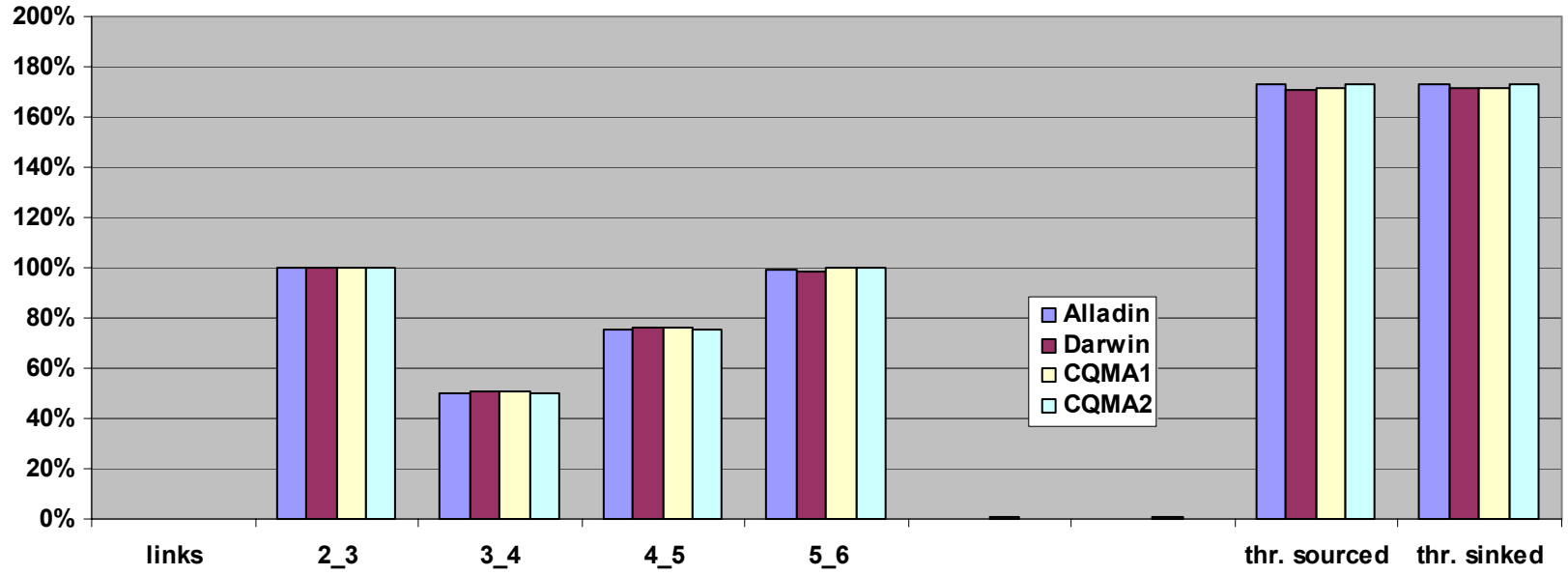
Scenario 2: Parallel Parking Lot



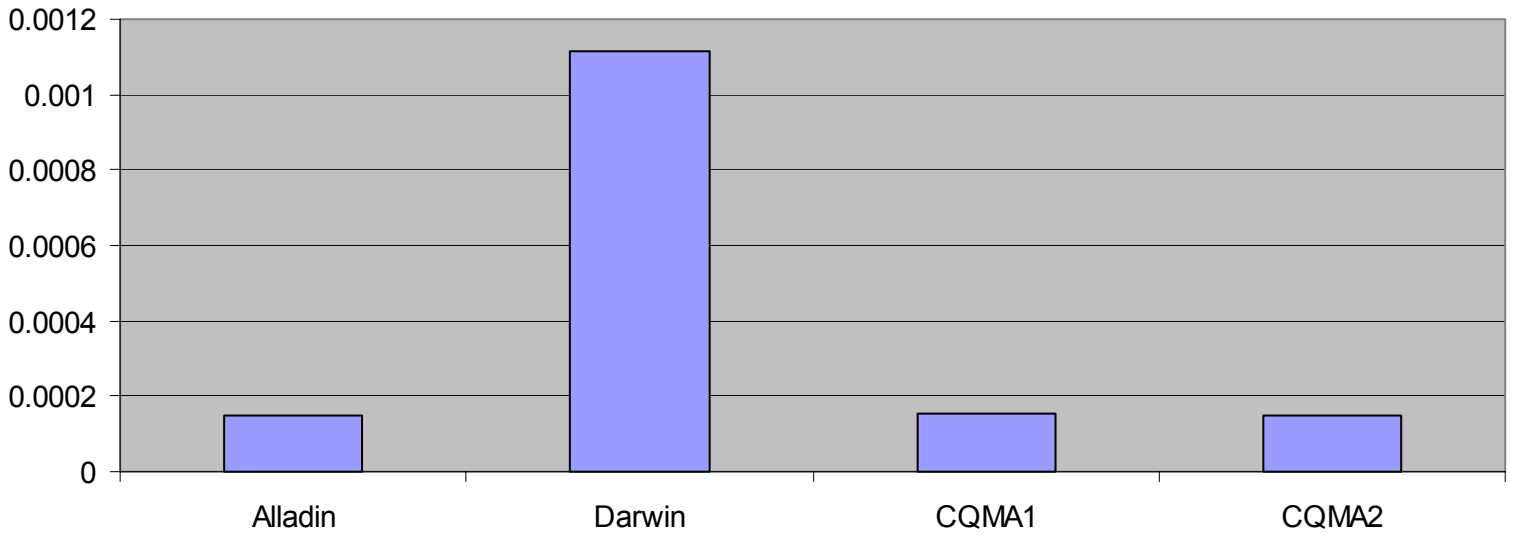
Flows



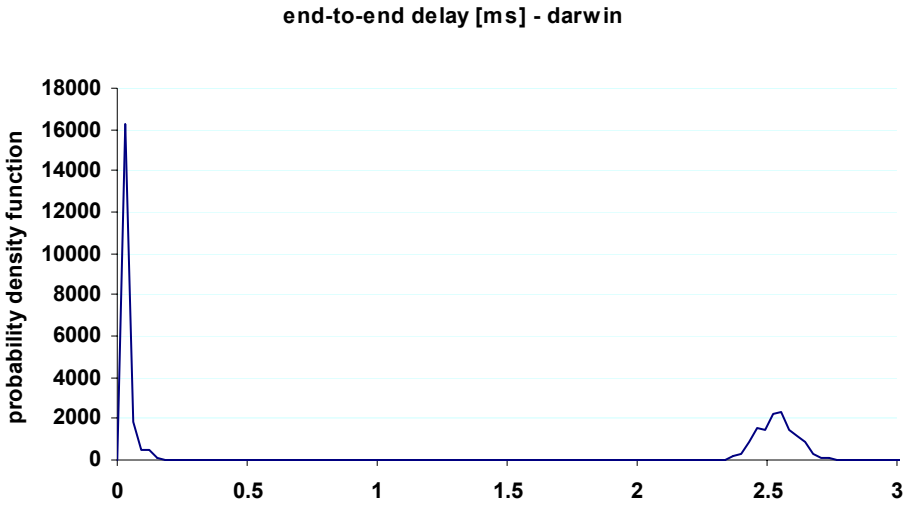
Throughput

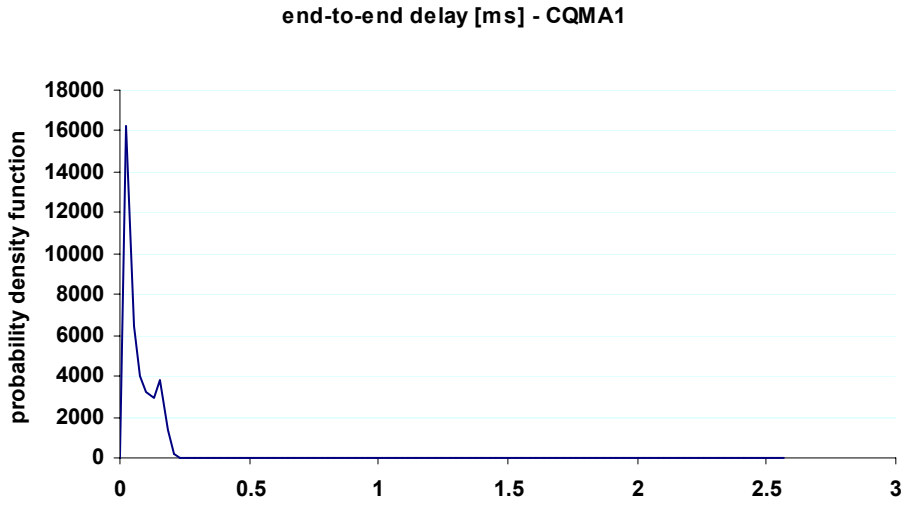

200%
180%
160%
140%
120%
100%
80%
60%
40%
20%
0%
Alladin
Darwin
CQMA1
CQMA2
links
2_3
3_4
4_5
5_6
thr. sourced
thr. sinked

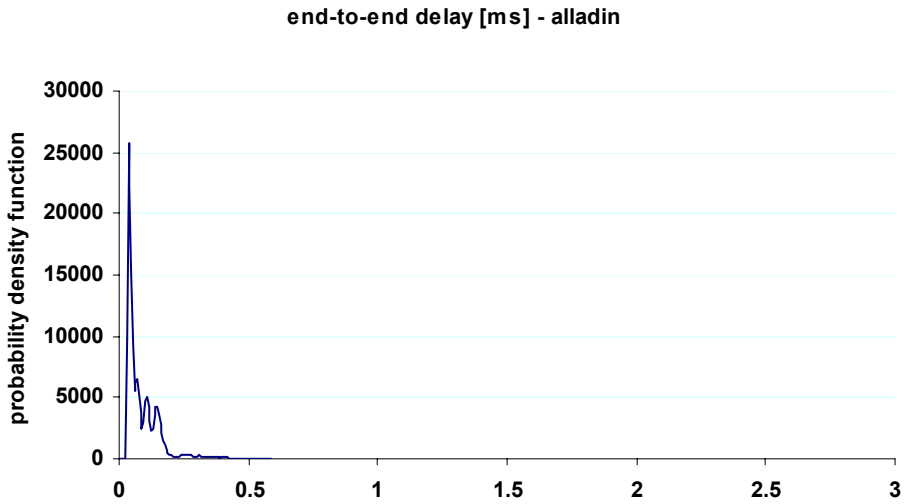
Delay

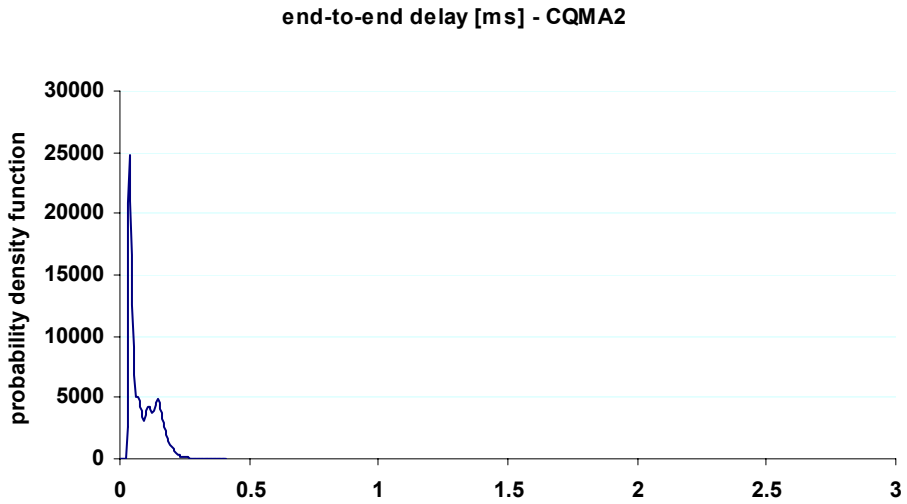

0.0012
0.001
0.0008
0.0006
0.0004
0.0002
0
Alladin
Darwin
CQMA1
CQMA2

Probability Density Function (PDF)
MAC end to end Delay

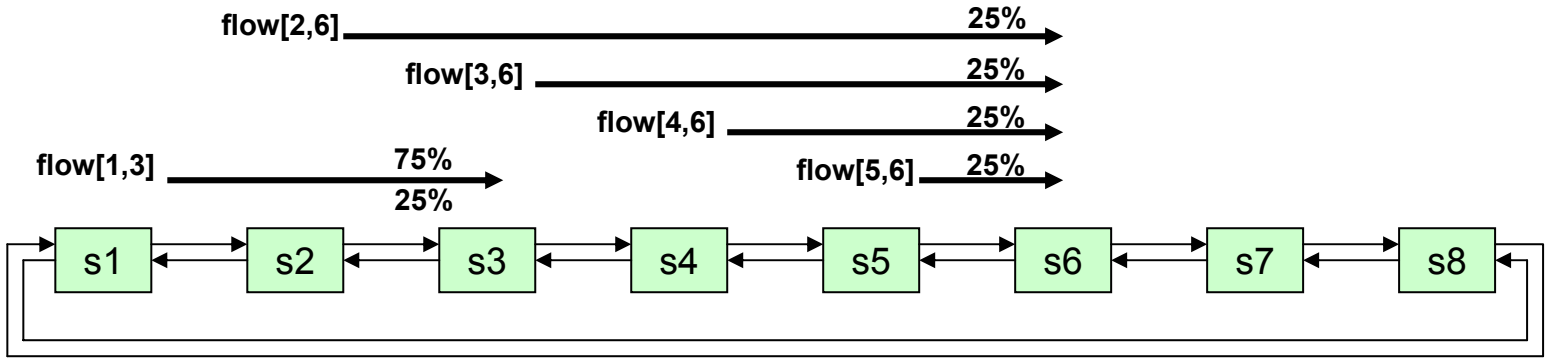




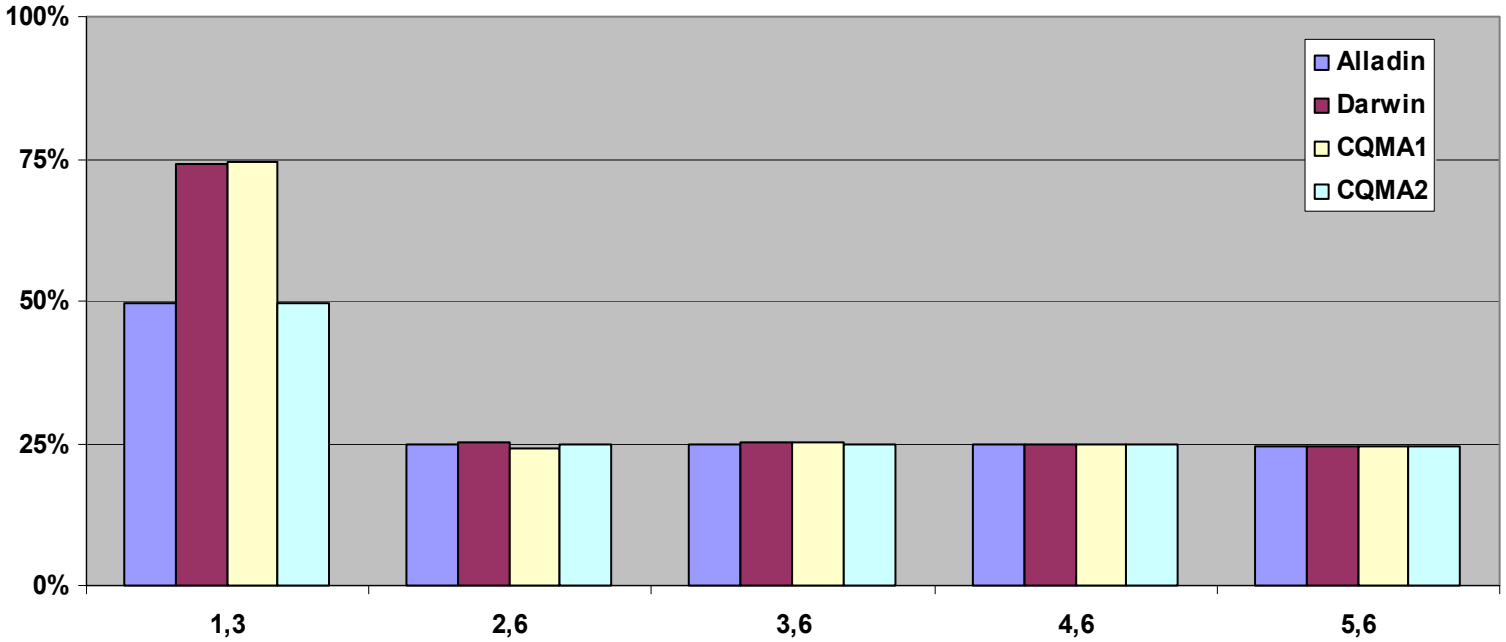




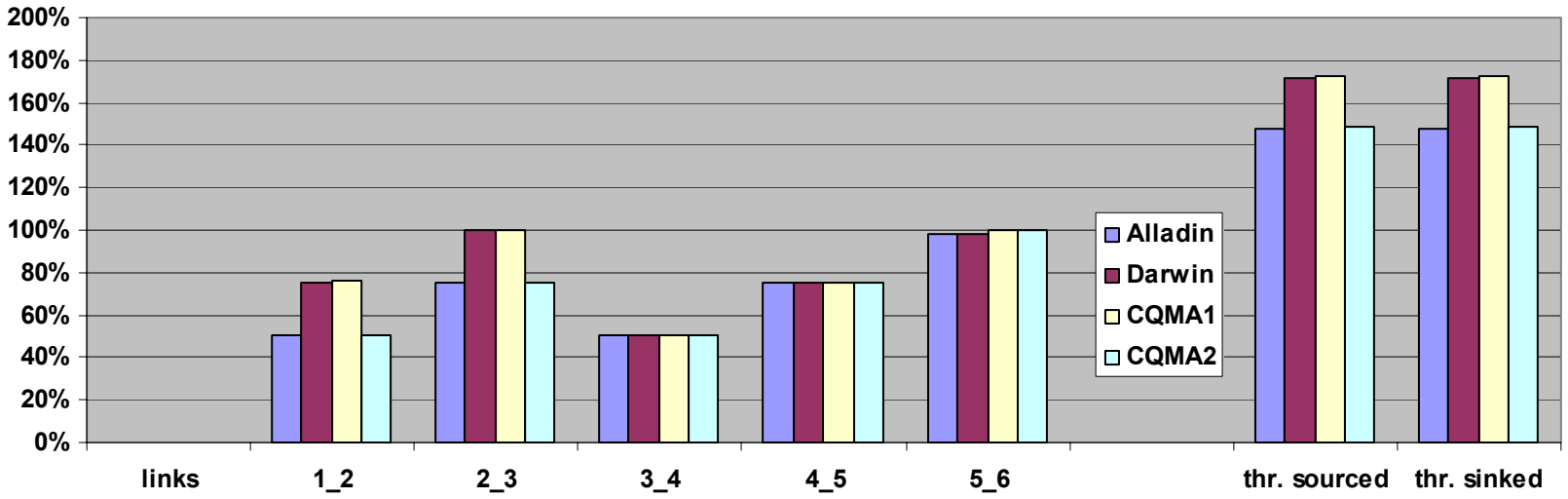
Scenario 3: Upstream Parallel Parking Lot



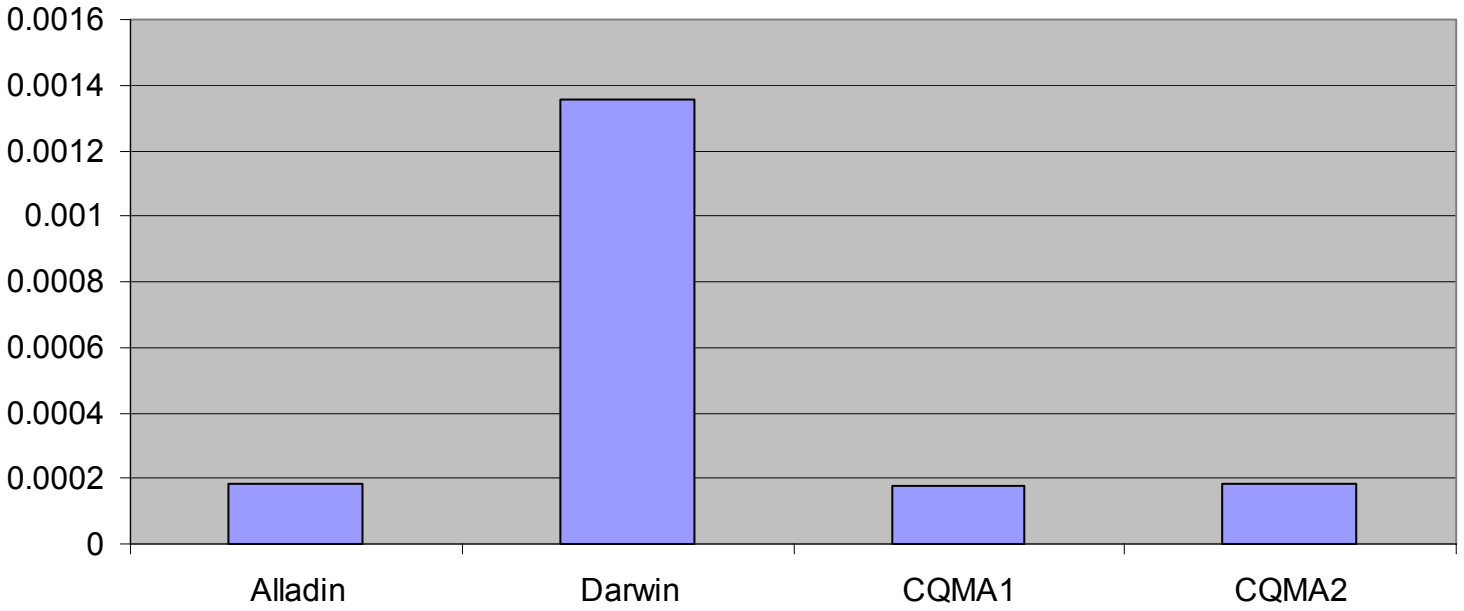
Flows



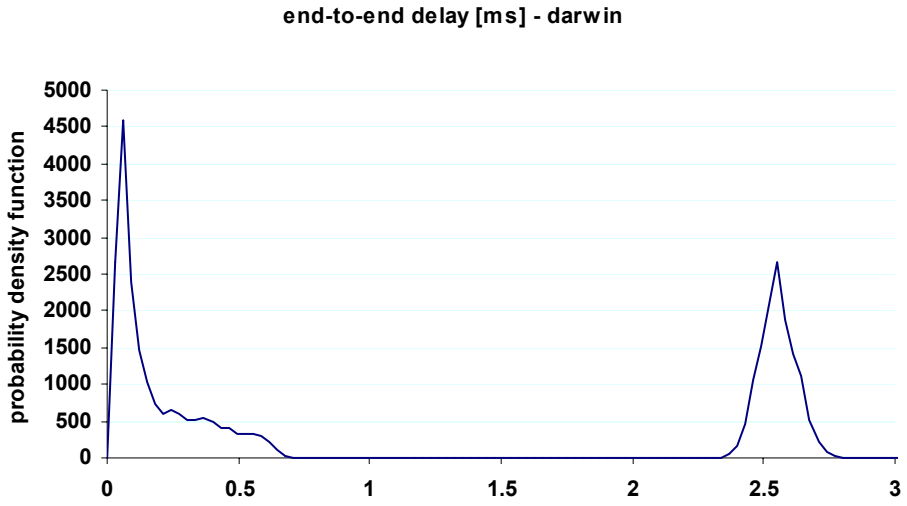
Throughput

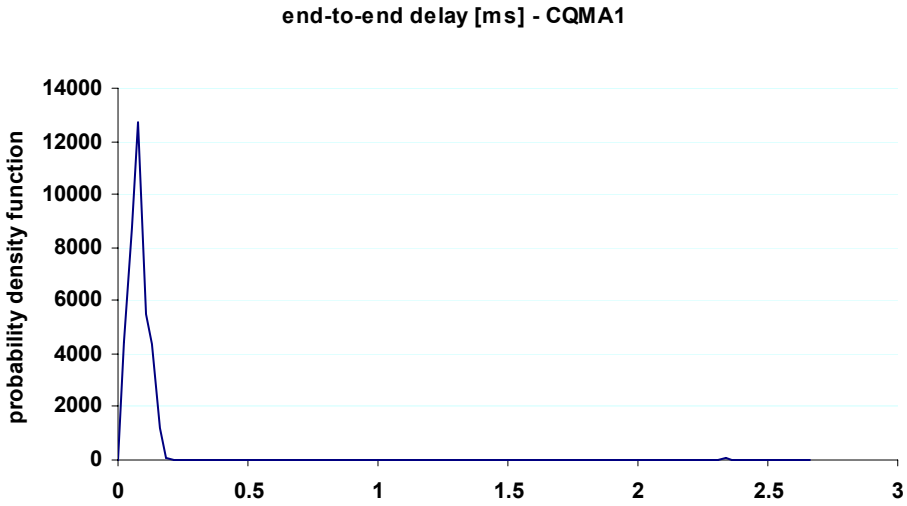

200%
180%
160%
140%
120%
100%
80%
60%
40%
20%
0%
links
1_2
2_3
3_4
4_5
5_6
thr. sourced
thr. sinked
Alladin
Darwin
CQMA1
CQMA2

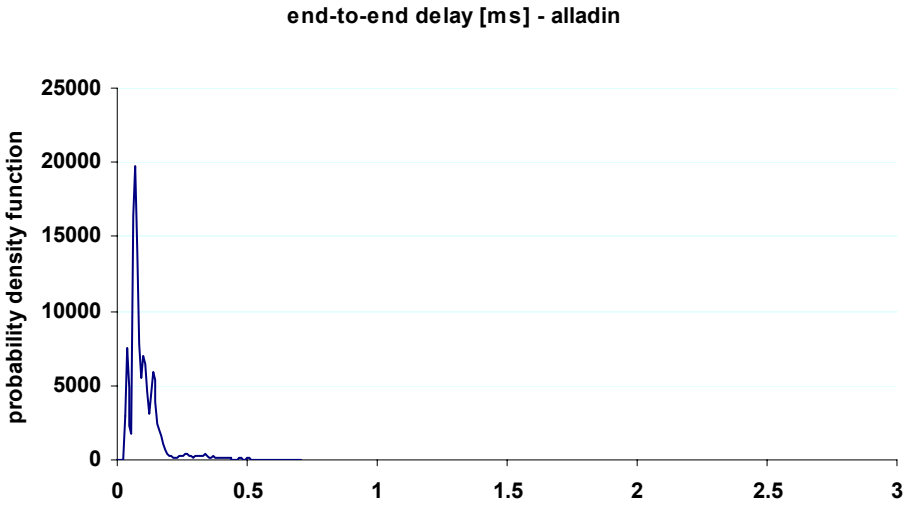
Delay

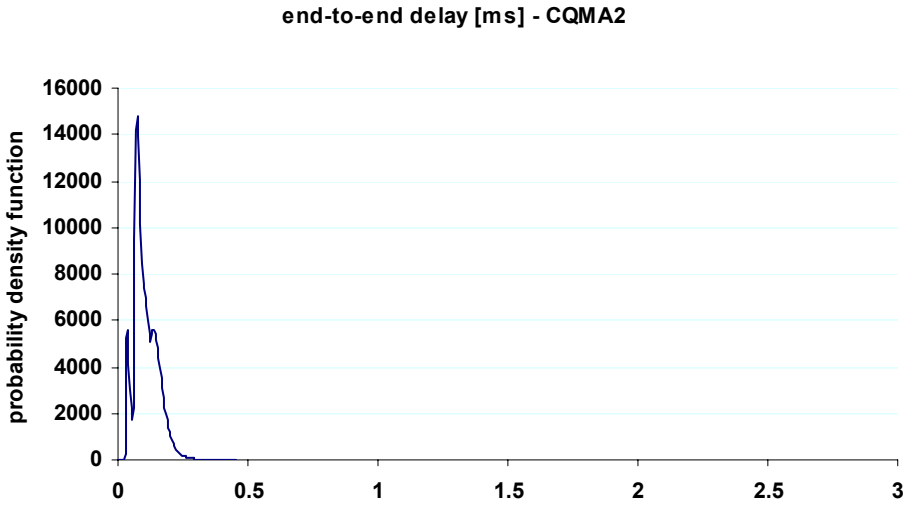

0.0016
0.0014
0.0012
0.001
0.0008
0.0006
0.0004
0.0002
0
Alladin
Darwin
CQMA1
CQMA2

Probability Density Function (PDF)
MAC end to end Delay

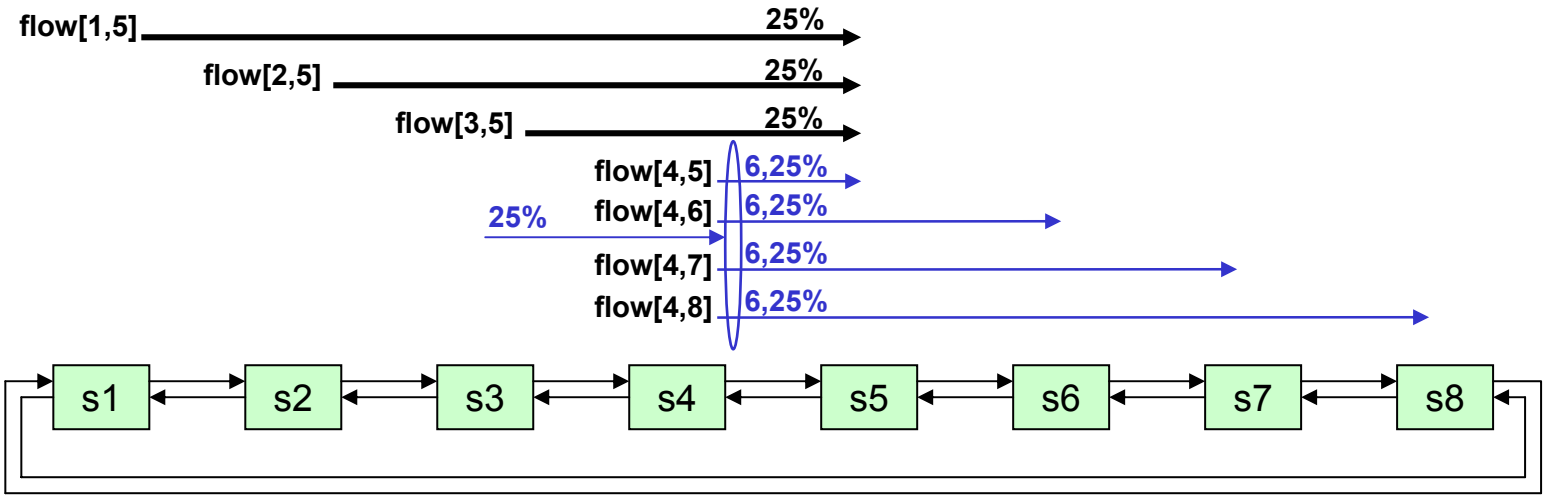




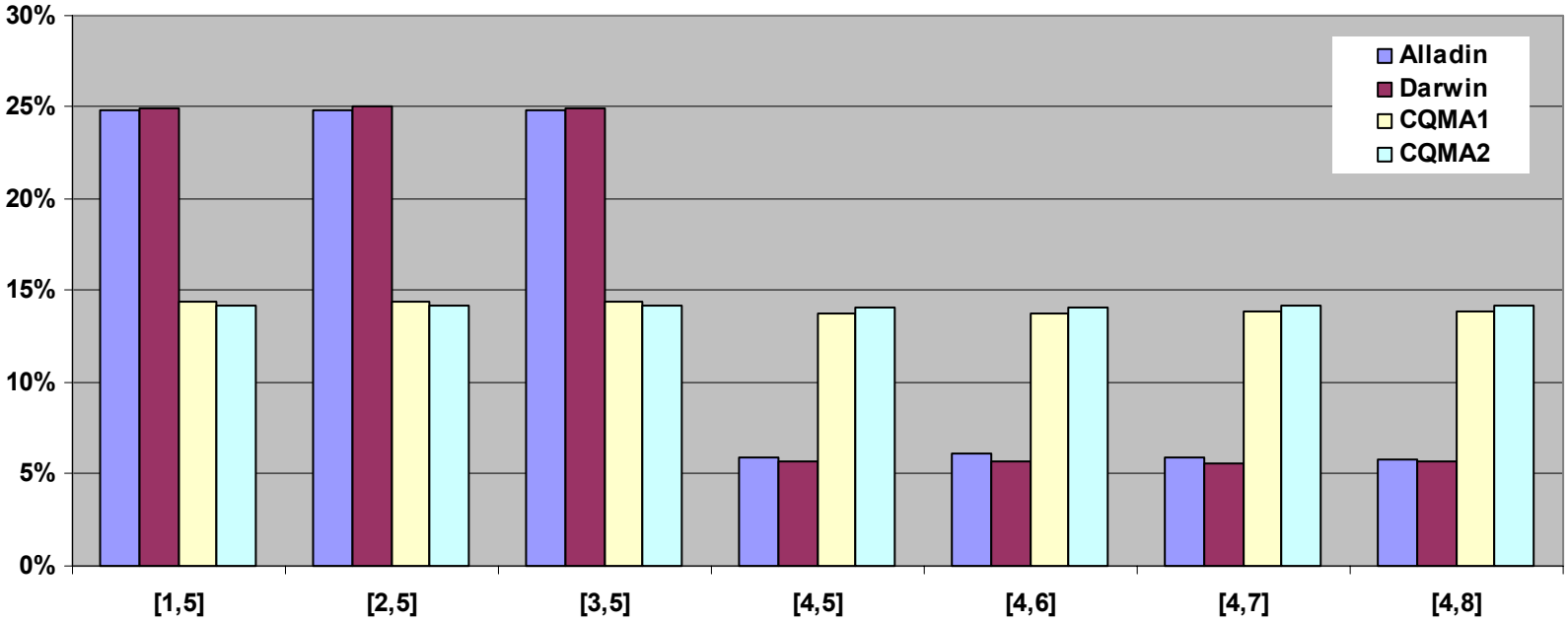




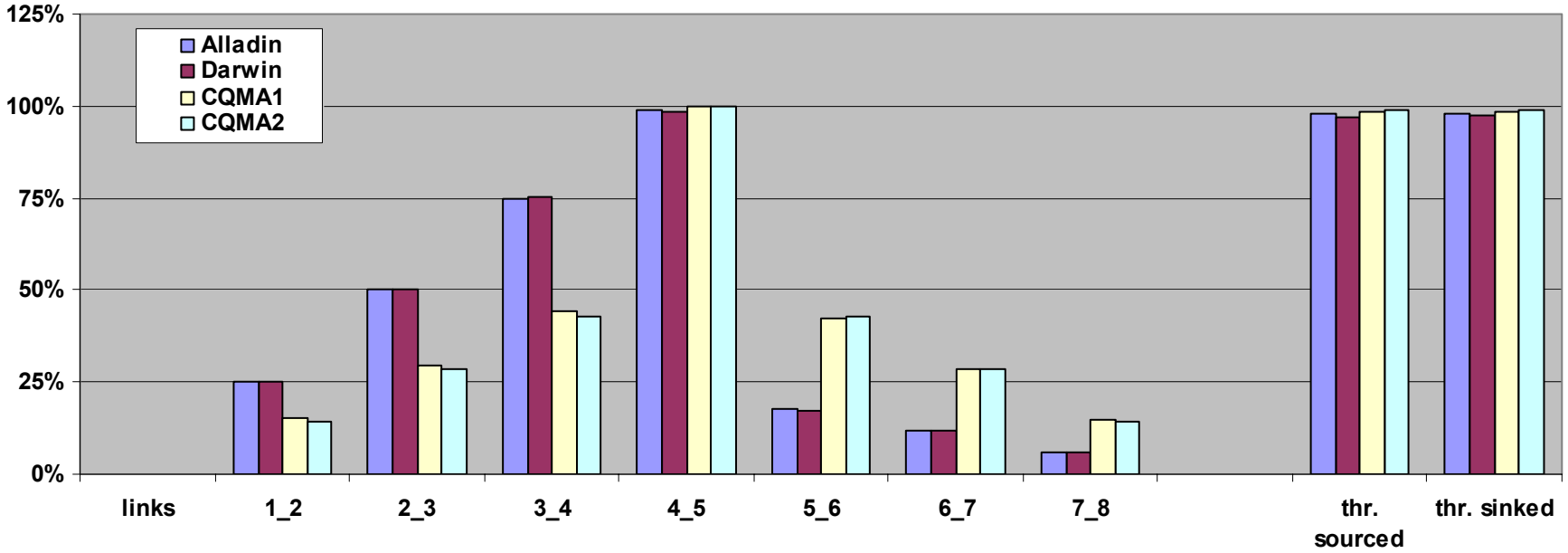
Scenario 4: Multi-Flow Parking Lot



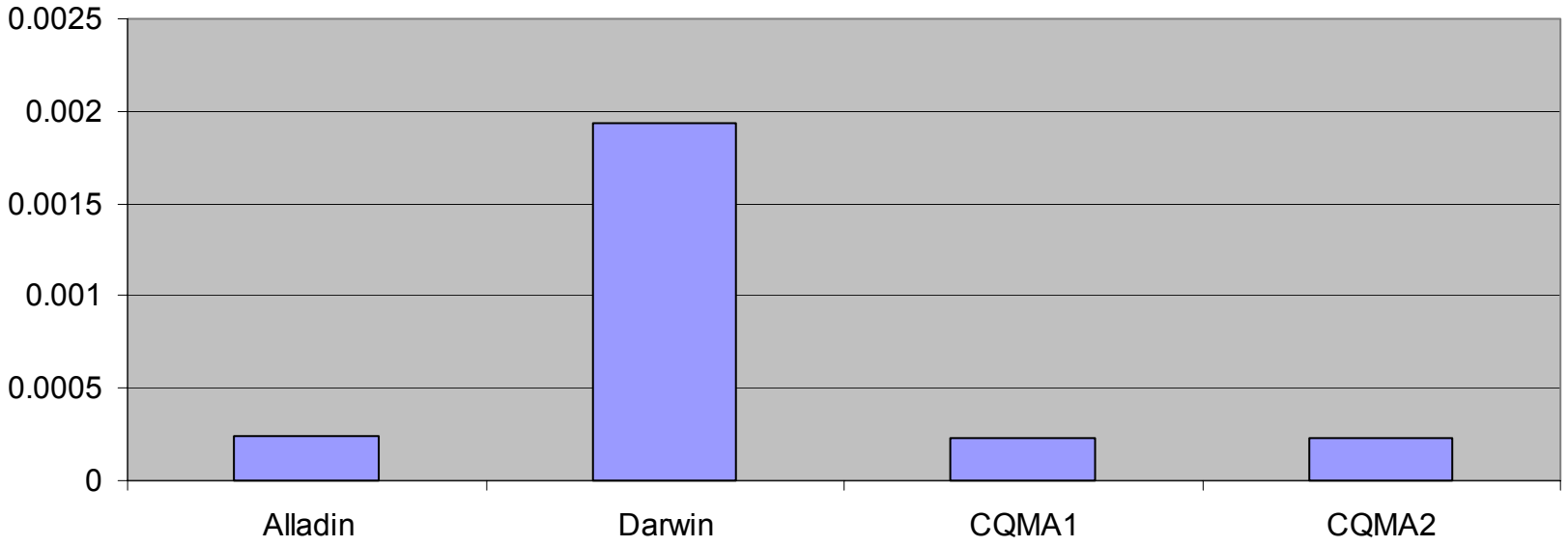
Flows



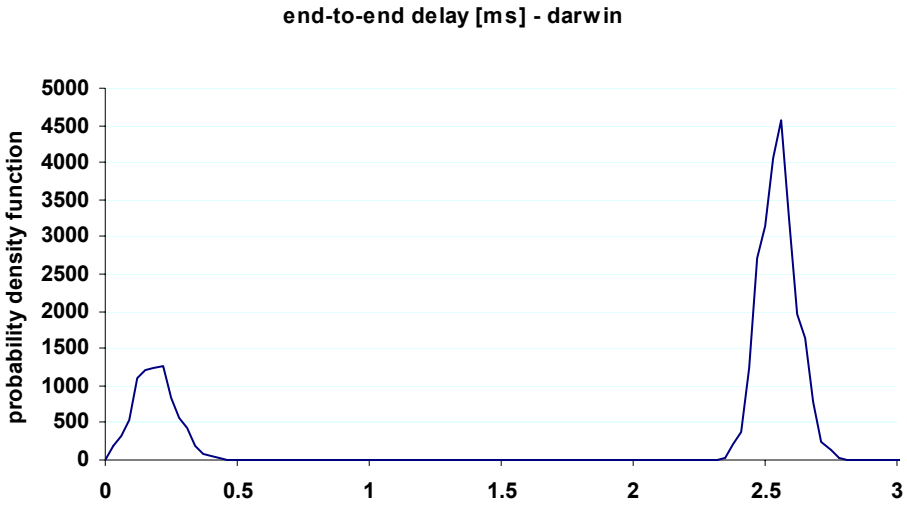
Throughput

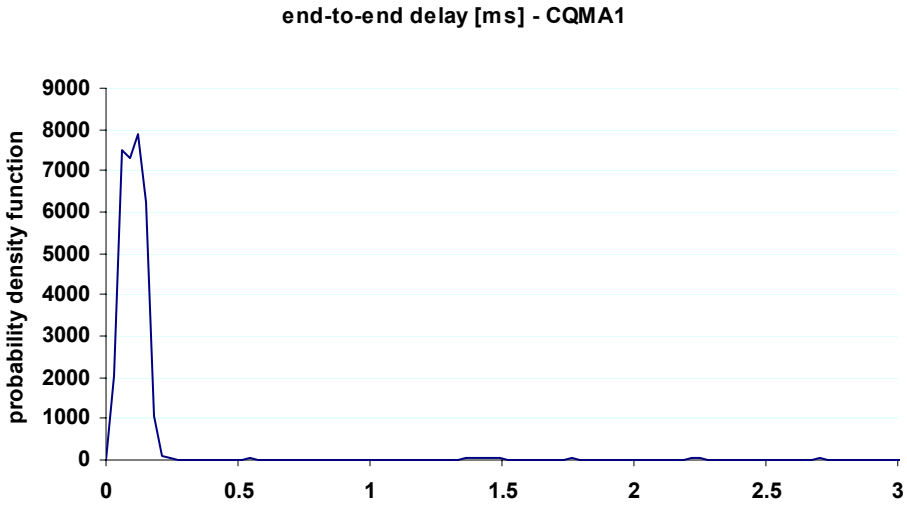

125%
100%
75%
50%
25%
0%
Alladin
Darwin
CQMA1
CQMA2
links
1_2
2_3
3_4
4_5
5_6
6_7
7_8
thr. sourced
thr. sinked

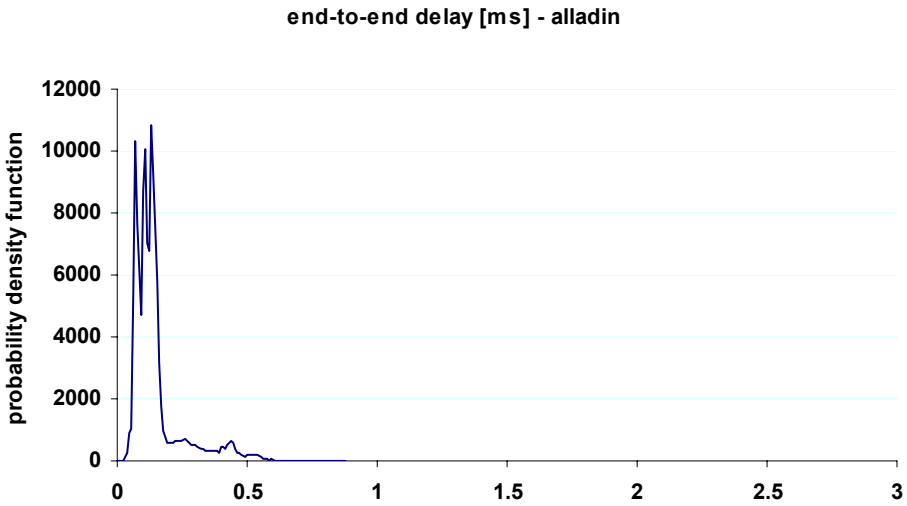
Delay

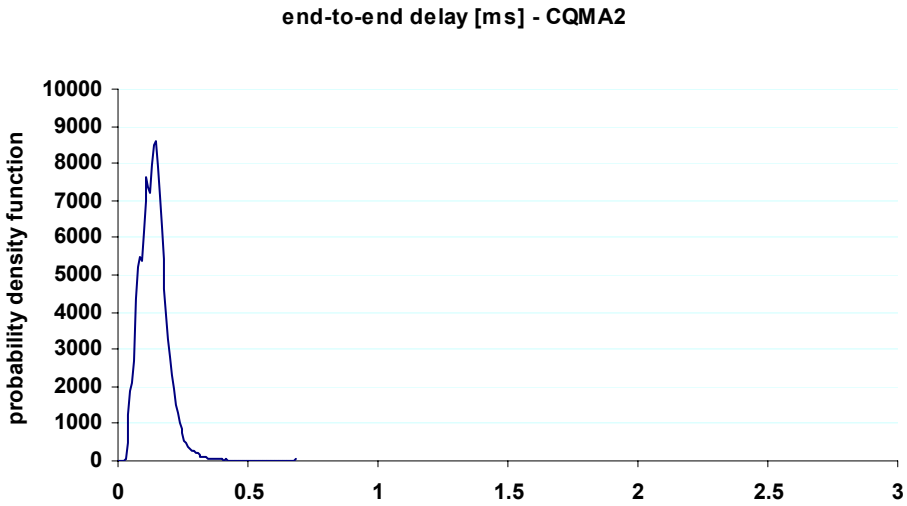


Probability Density Function (PDF)
MAC end to end Delay

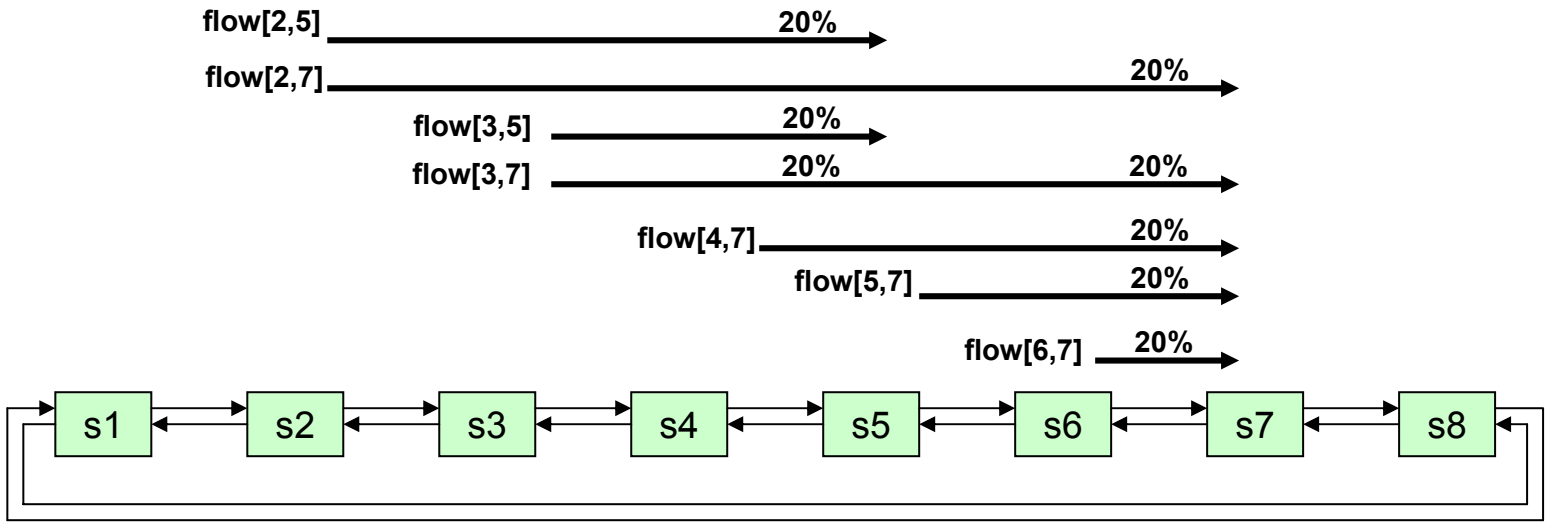




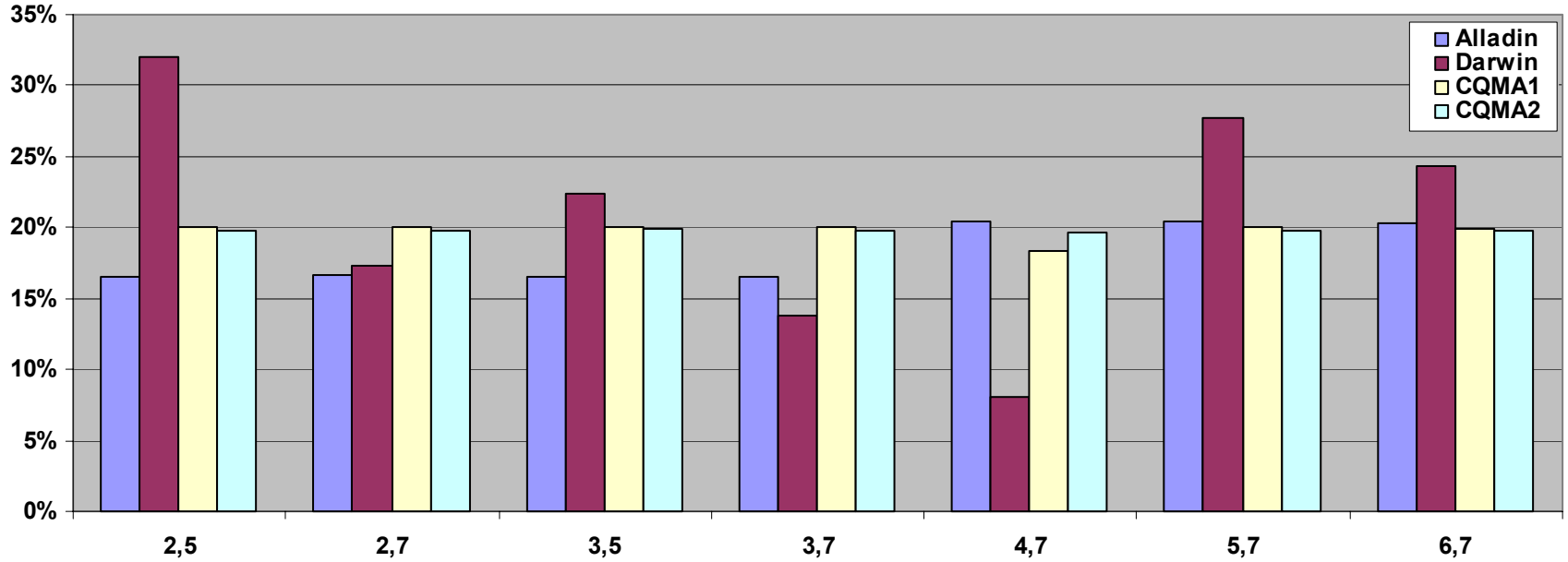




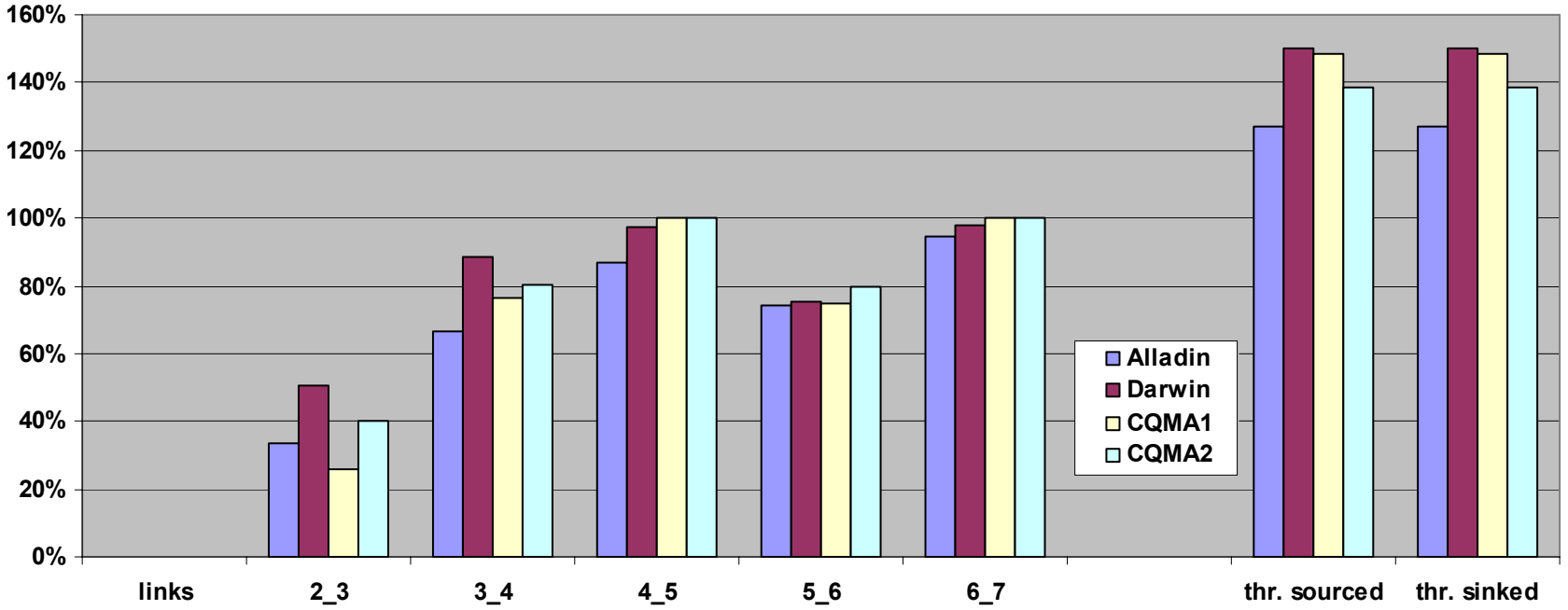
Scenario 5: Dual-exit parking lot



Flows


35%
30%
25%
20%
15%
10%
5%
0%
2,5
2,7
3,5
3,7
4,7
5,7
6,7
Alladin
Darwin
CQMA1
CQMA2

Throughput


160%
140%
120%
100%
80%
60%
40%
20%
0%
links
2_3
3_4
4_5
5_6
6_7
thr. sourced
thr. sinked
Alladin
Darwin
CQMA1
CQMA2

Probability Density Function (PDF)
MAC end to end Delay