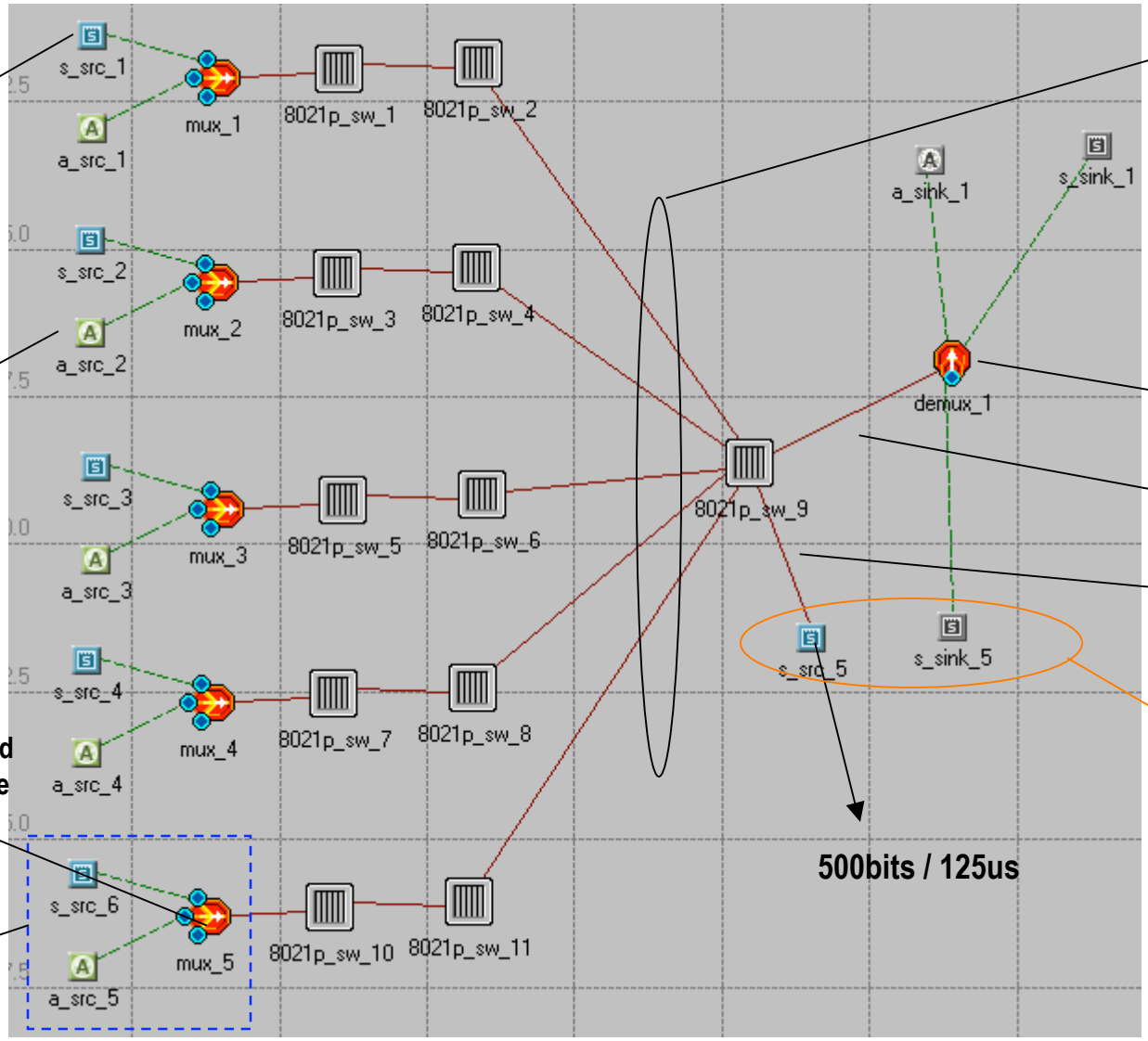
On the worst case, and pacing

Felix Feifei Feng
feng.fei@samsung.com

SAMSUNG Electronics

IEEE 802.3 RESG 2005 San Jose

Scenario 1: Using 802.1p switches



2Kbits / 125us

12Kbits / 1s

MUX: MAC interface of end-station. To multiplex and send out streams in one station

End-station

Conflicting Streams

DEMUX: MAC interface of end-station

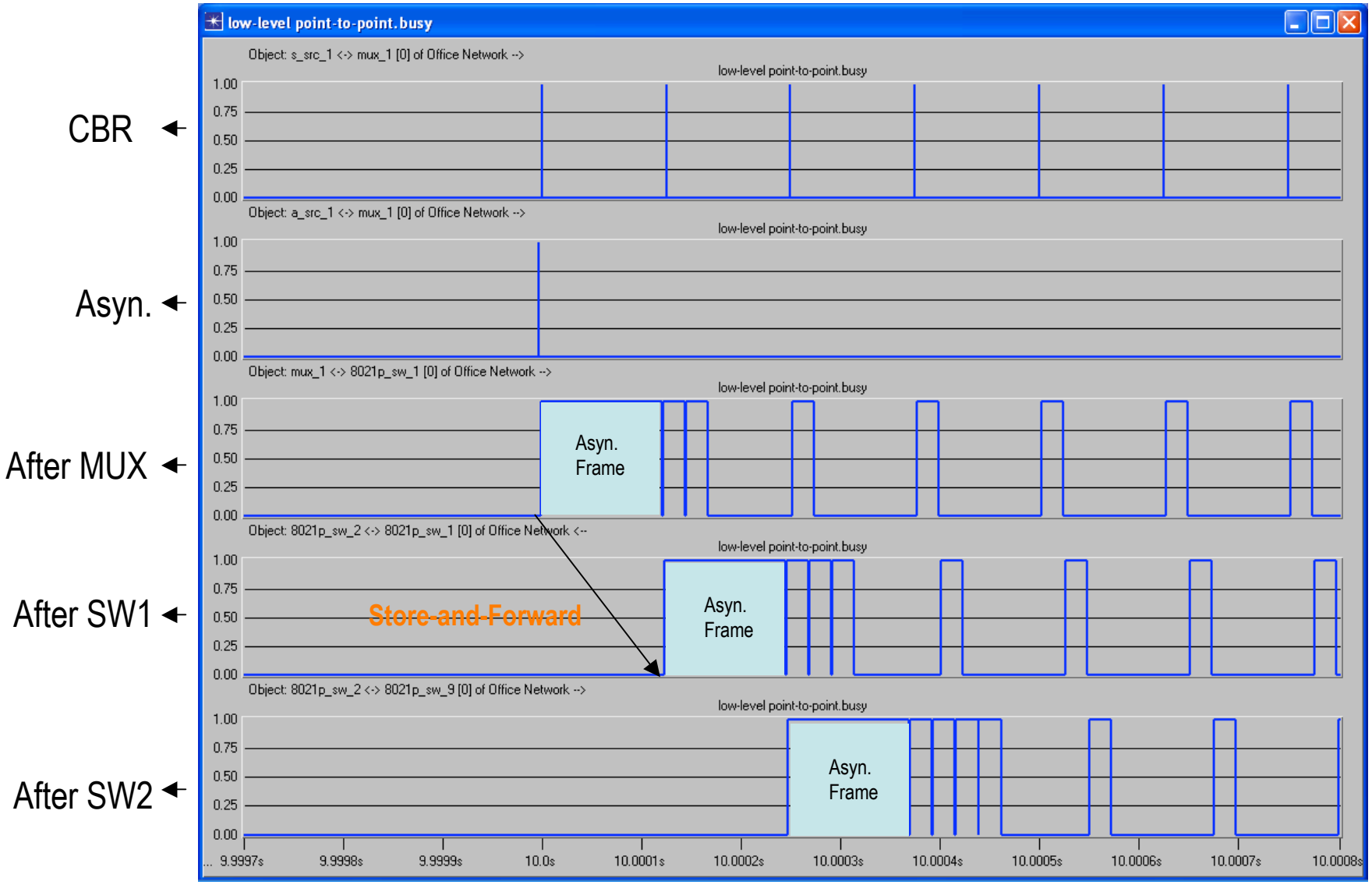
Output

Input

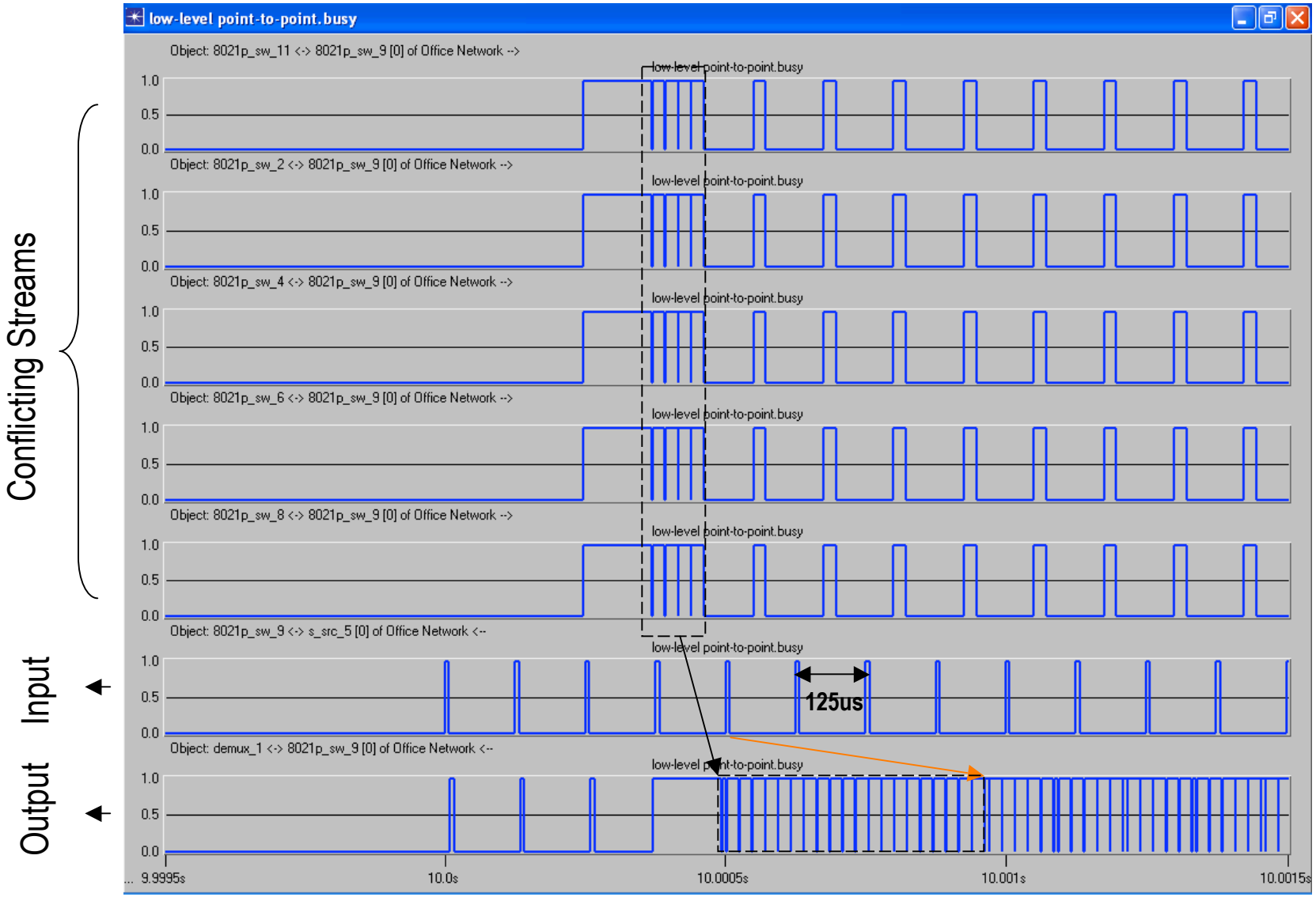
500bits / 125us

Our interested source/sink pair

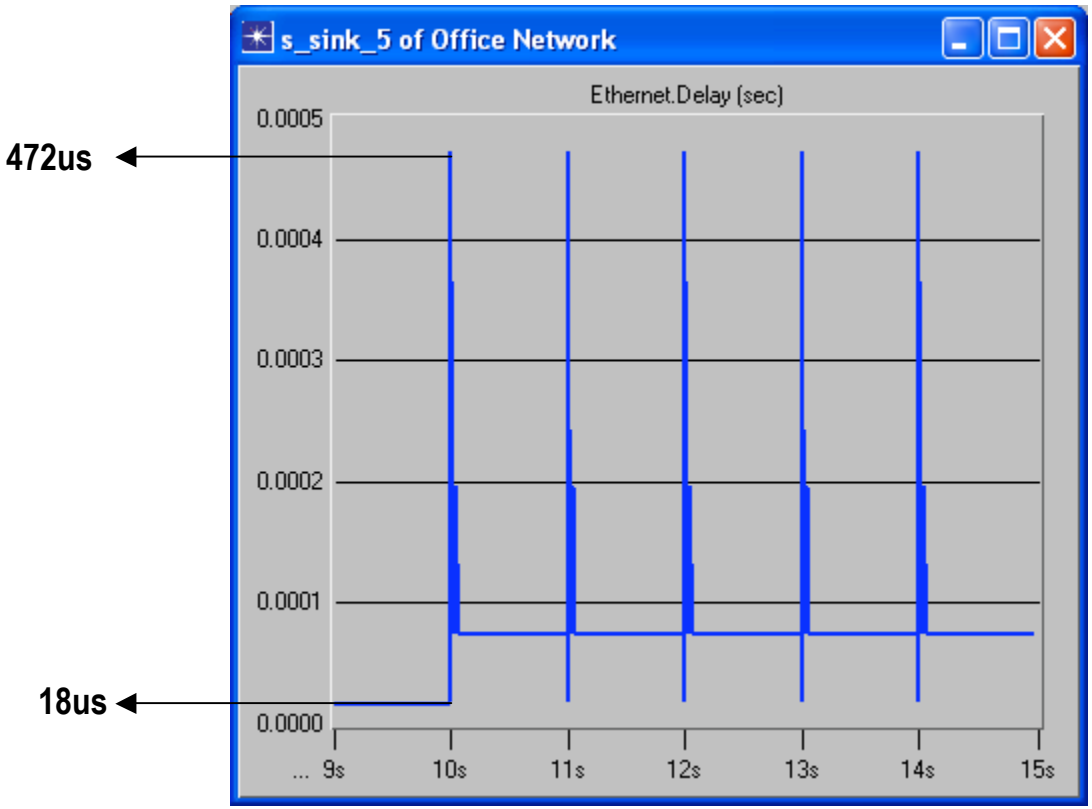
Distortion of CBR traffics



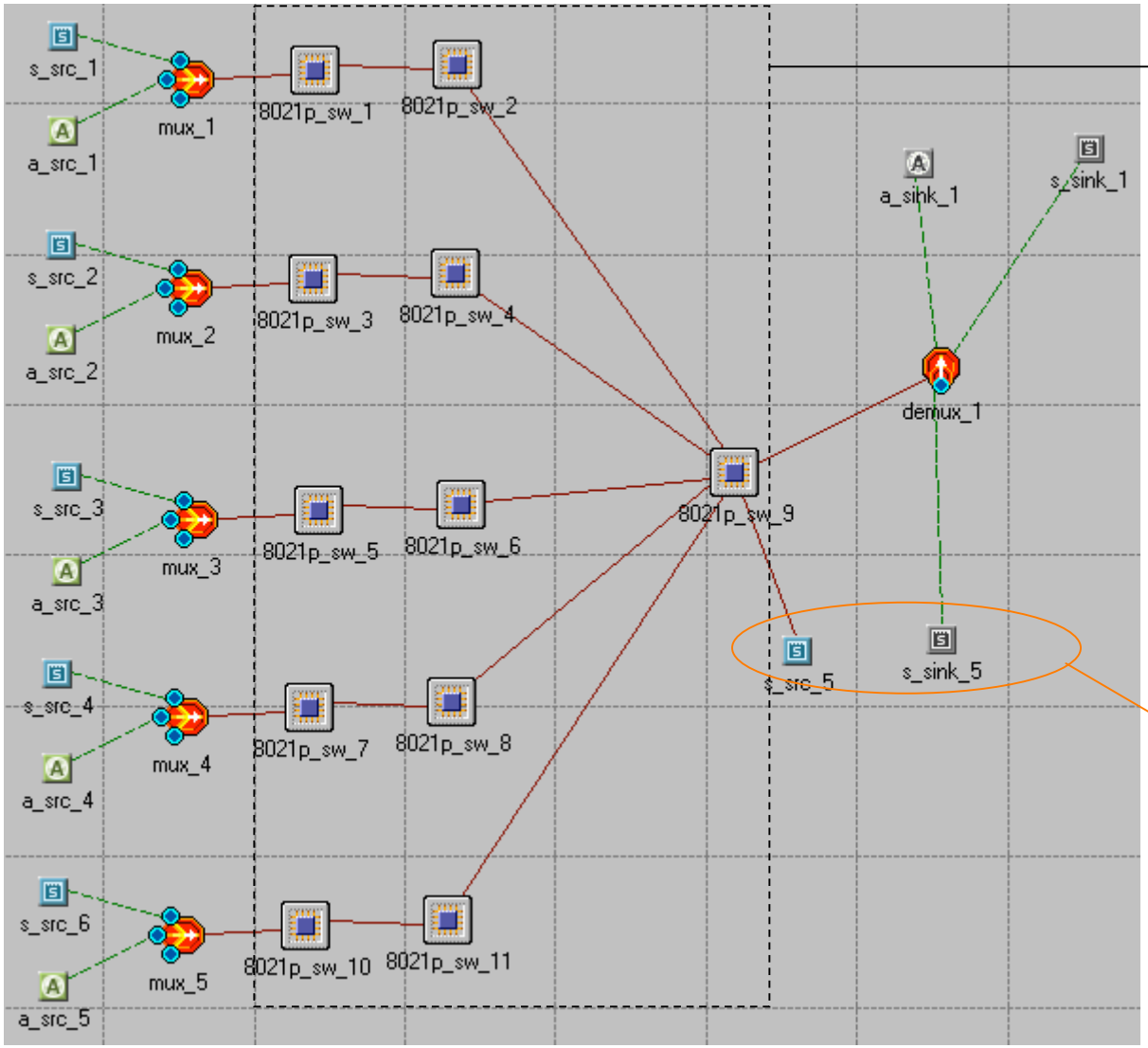
Contention


low-level point-to-point.busy
Object: 8021p_sw_11 <-> 8021p_sw_9 [0] of Office Network -->
Object: 8021p_sw_2 <-> 8021p_sw_9 [0] of Office Network -->
Object: 8021p_sw_4 <-> 8021p_sw_9 [0] of Office Network -->
Object: 8021p_sw_6 <-> 8021p_sw_9 [0] of Office Network -->
Object: 8021p_sw_8 <-> 8021p_sw_9 [0] of Office Network -->
Object: 8021p_sw_9 <-> s_src_5 [0] of Office Network <--
Object: demux_1 <-> 8021p_sw_9 [0] of Office Network <--
1.0
0.5
0.0
125us
9.9995s
10.0s
10.0005s
10.001s
10.0015s
Conflicting Streams
Input
Output

Delay Results



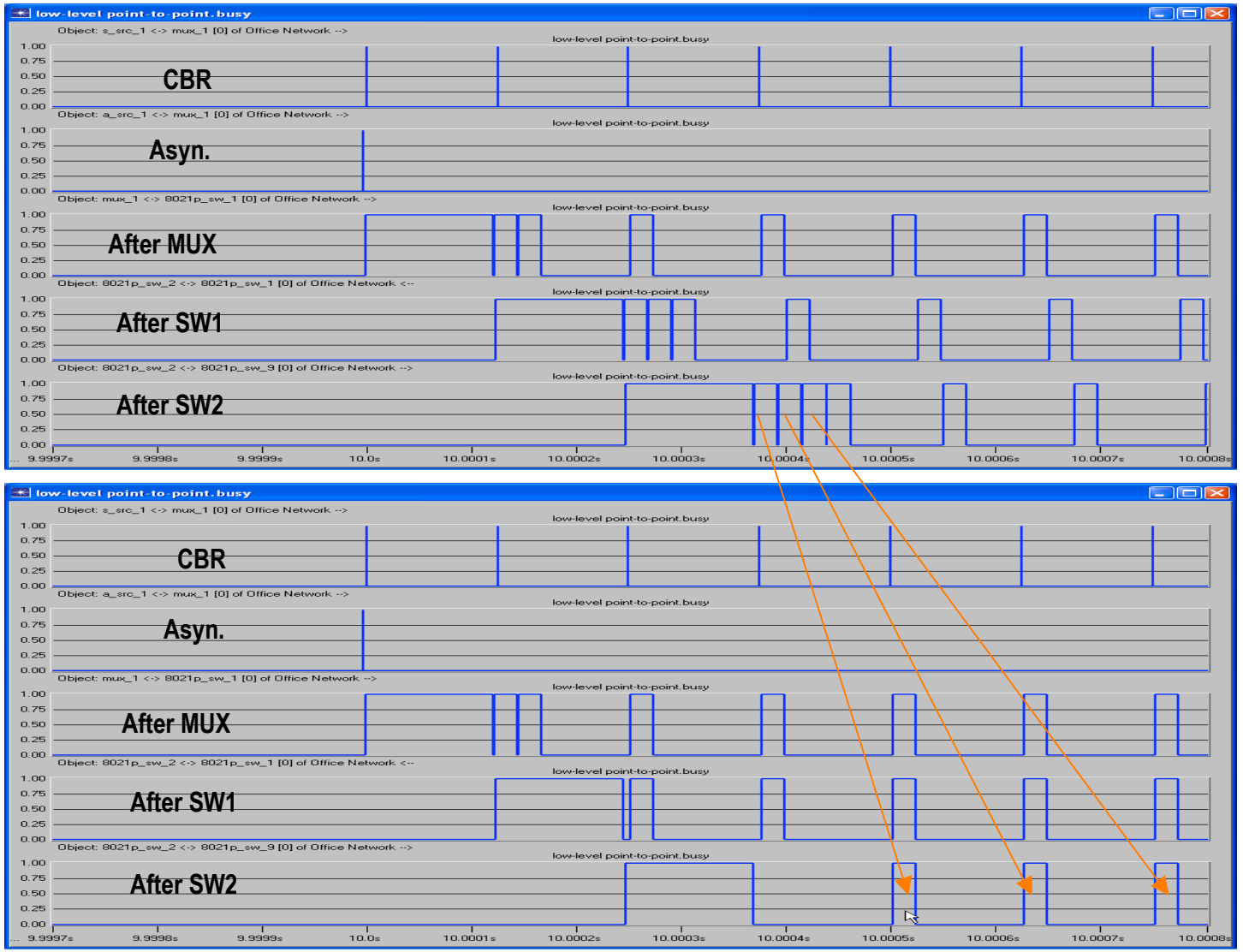
Scenario 2: Using Pacing-based Switches



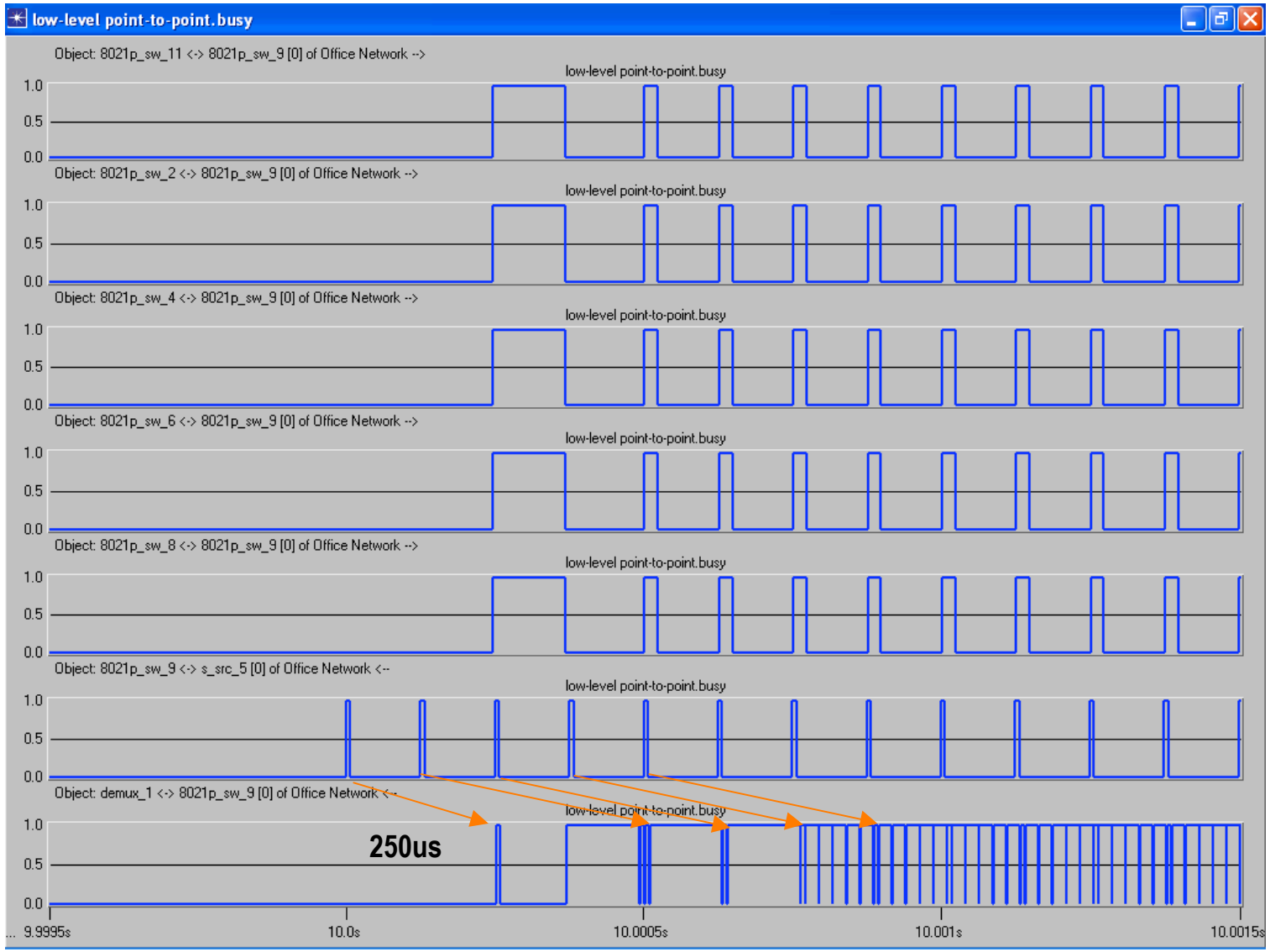
Pacing Avoids the Traffic Distortion

802.1p

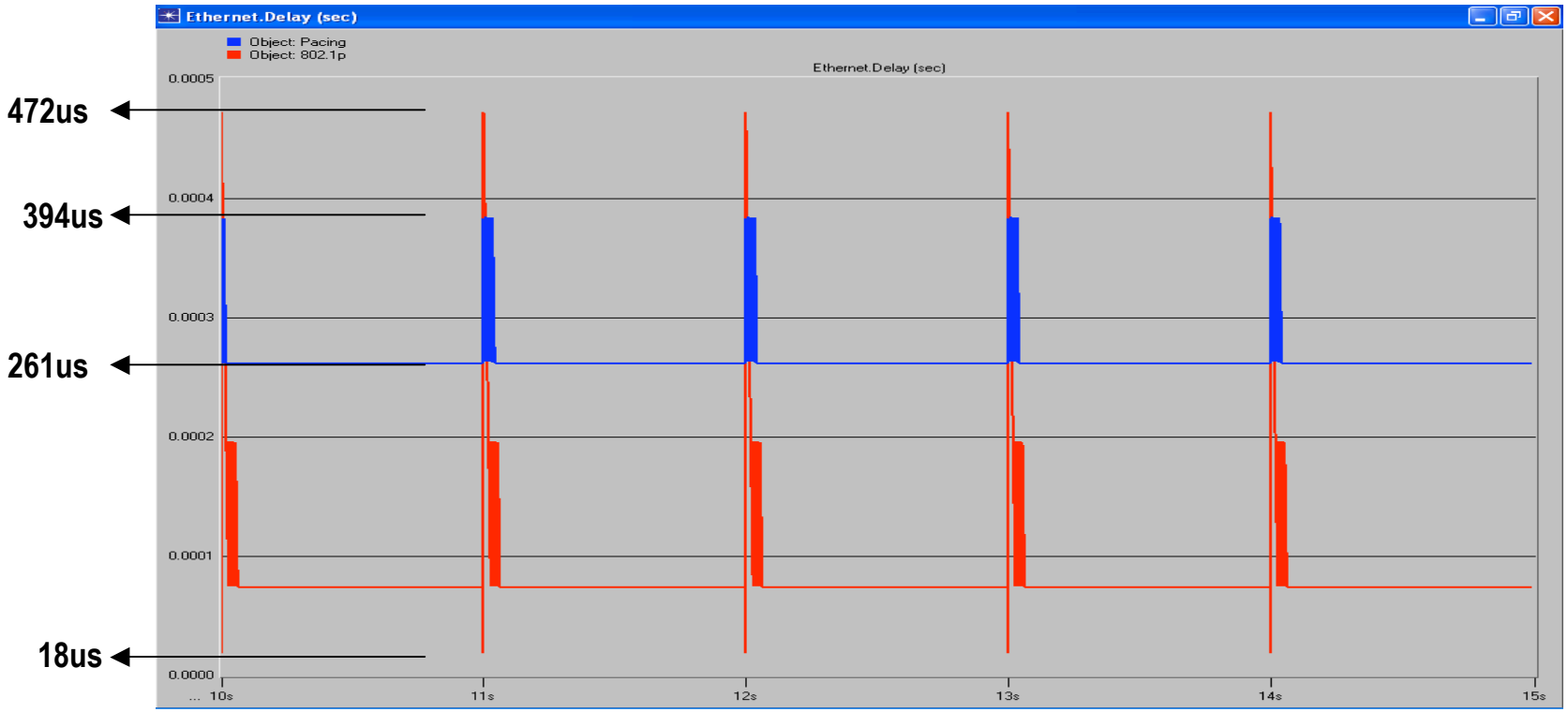
Pacing



Pacing


low-level point-to-point.busy
Object: 8021p_sw_11 <-> 8021p_sw_9 [0] of Office Network -->
low-level point-to-point.busy
Object: 8021p_sw_2 <-> 8021p_sw_9 [0] of Office Network -->
low-level point-to-point.busy
Object: 8021p_sw_4 <-> 8021p_sw_9 [0] of Office Network -->
low-level point-to-point.busy
Object: 8021p_sw_6 <-> 8021p_sw_9 [0] of Office Network -->
low-level point-to-point.busy
Object: 8021p_sw_8 <-> 8021p_sw_9 [0] of Office Network -->
low-level point-to-point.busy
Object: 8021p_sw_9 <-> s_src_5 [0] of Office Network <--
low-level point-to-point.busy
Object: demux_1 <-> 8021p_sw_9 [0] of Office Network <--
low-level point-to-point.busy
250us
1.0
0.5
0.0
9.9995s
10.0s
10.0005s
10.001s
10.0015s

Comparison of Delay Results



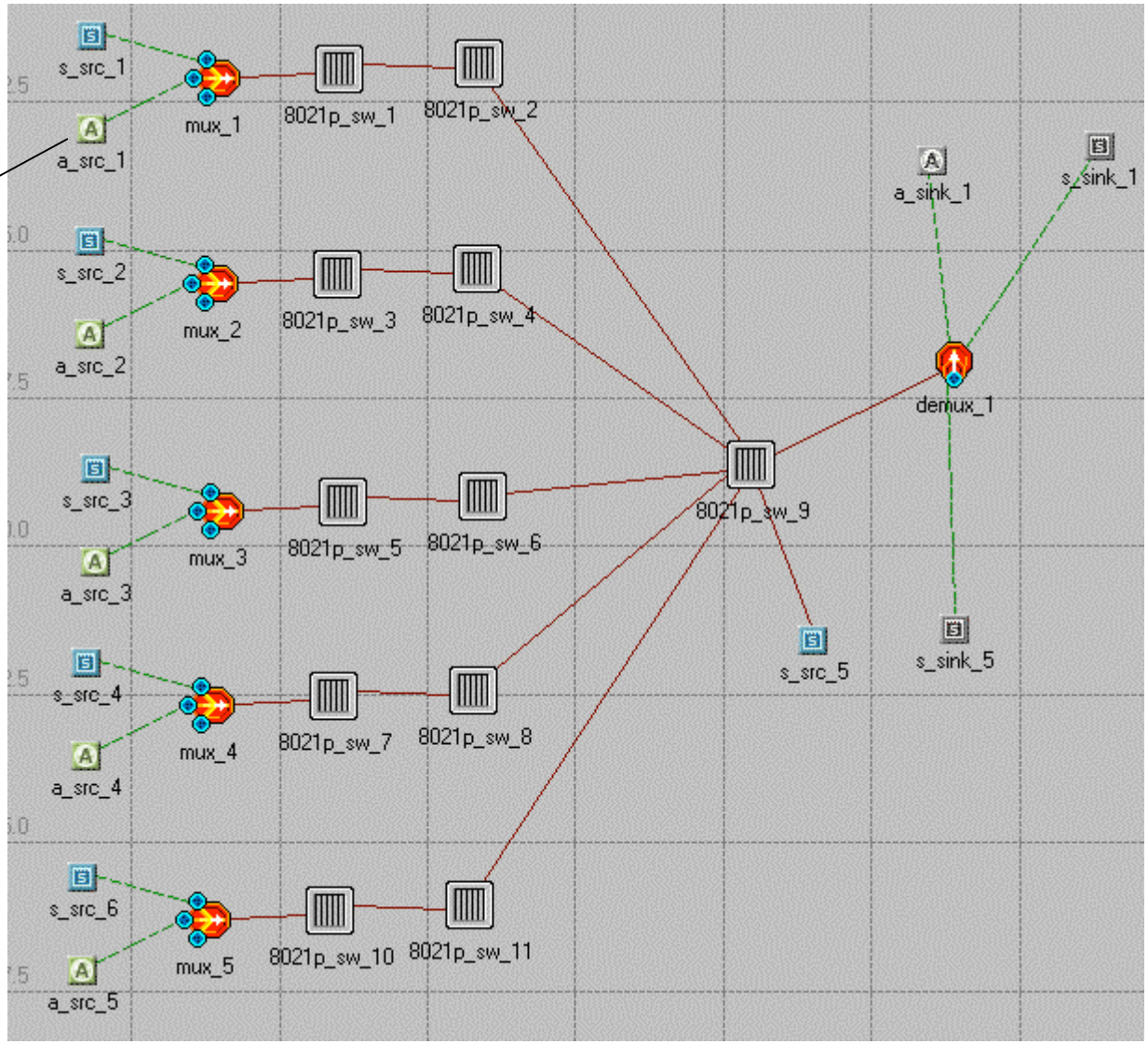
In this case, pacing can decrease both the delay and the delay variation. But the difference is margina l.

However…

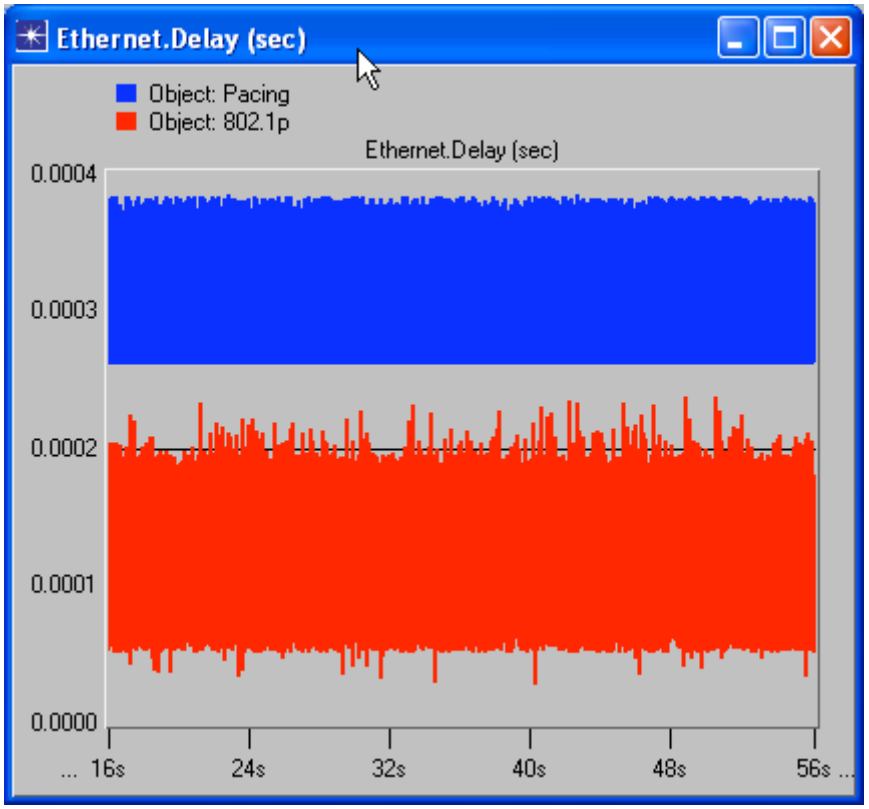
- This scenario is just an artificial case
  - We made all five conflicting CBR streams centralized. So when they come to switch-9, all five streams get bunched bursts and conflict with our interested stream at the same time. We made this case by:
    - Issuing a maximum size conflicting asynchronous packet to each conflicting CBR stream at the same time.
    - The conflicting CBR stream traverses through several store-and-forward switches, which makes the CBR packets bunched together
    - The link is almost fully loaded.
- In a realistic network
  - This kind of bunched conflicting may rarely occur

Scenario 3: A 'More Realistic' Scenario

Self-similar, H=0.7.
50packets/second.
Packet size =
uniform(1K,12K)



Delay Results

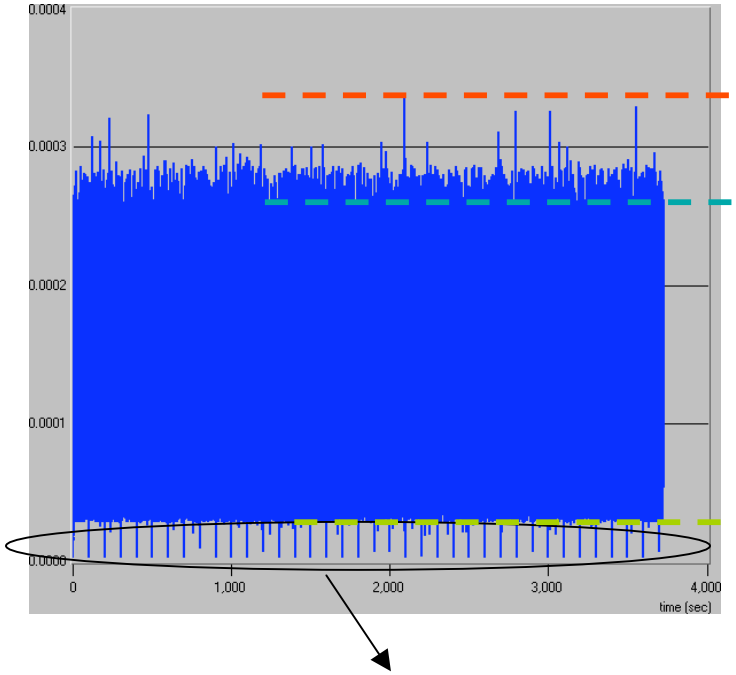


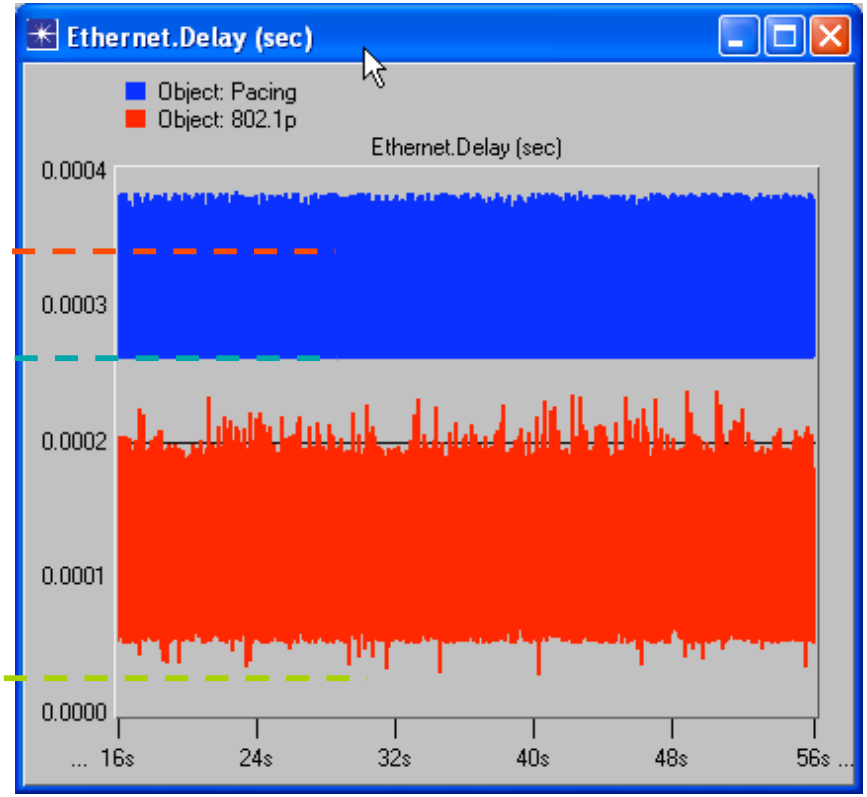
In this scenario, over 60 seconds, we didn't notice the occurrence of the worst case. Here pacing scheme has larger delay but smaller delay variation.

Additional tests: MTU Packets, Self-Similar Arrival

Self-similar, H=0.7.
50packets/second.
Packet size = 12K
With 802.1p switches

Self-similar, H=0.7.
50packets/second.
Packet size = uniform(1K,12K)



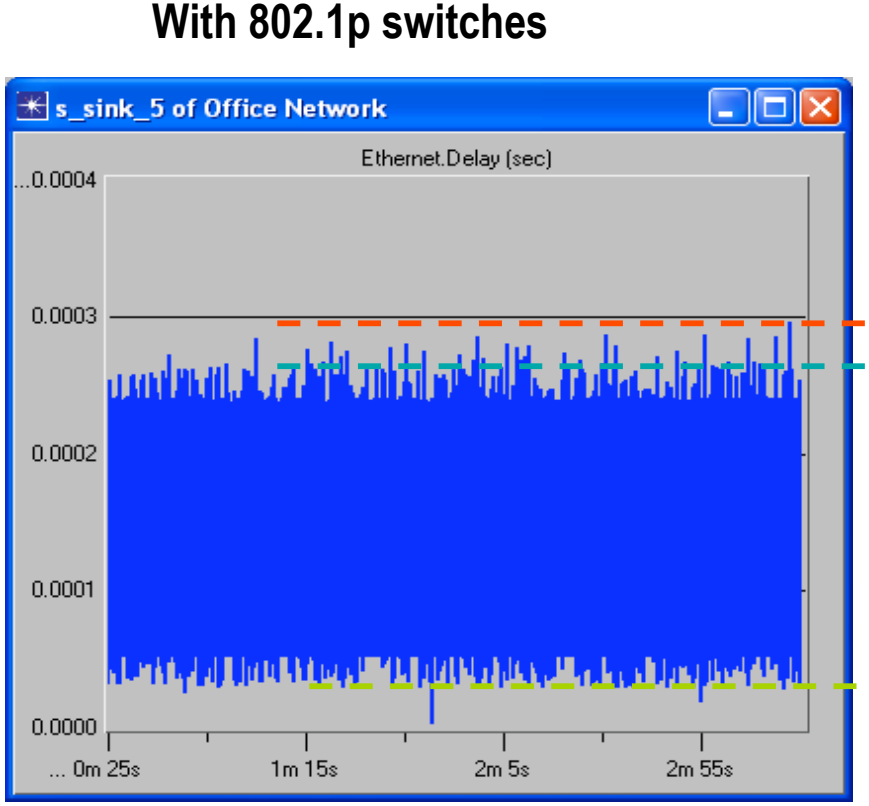


These small delays are because that I make sink nodes broadcast their addresses every 100 seconds.

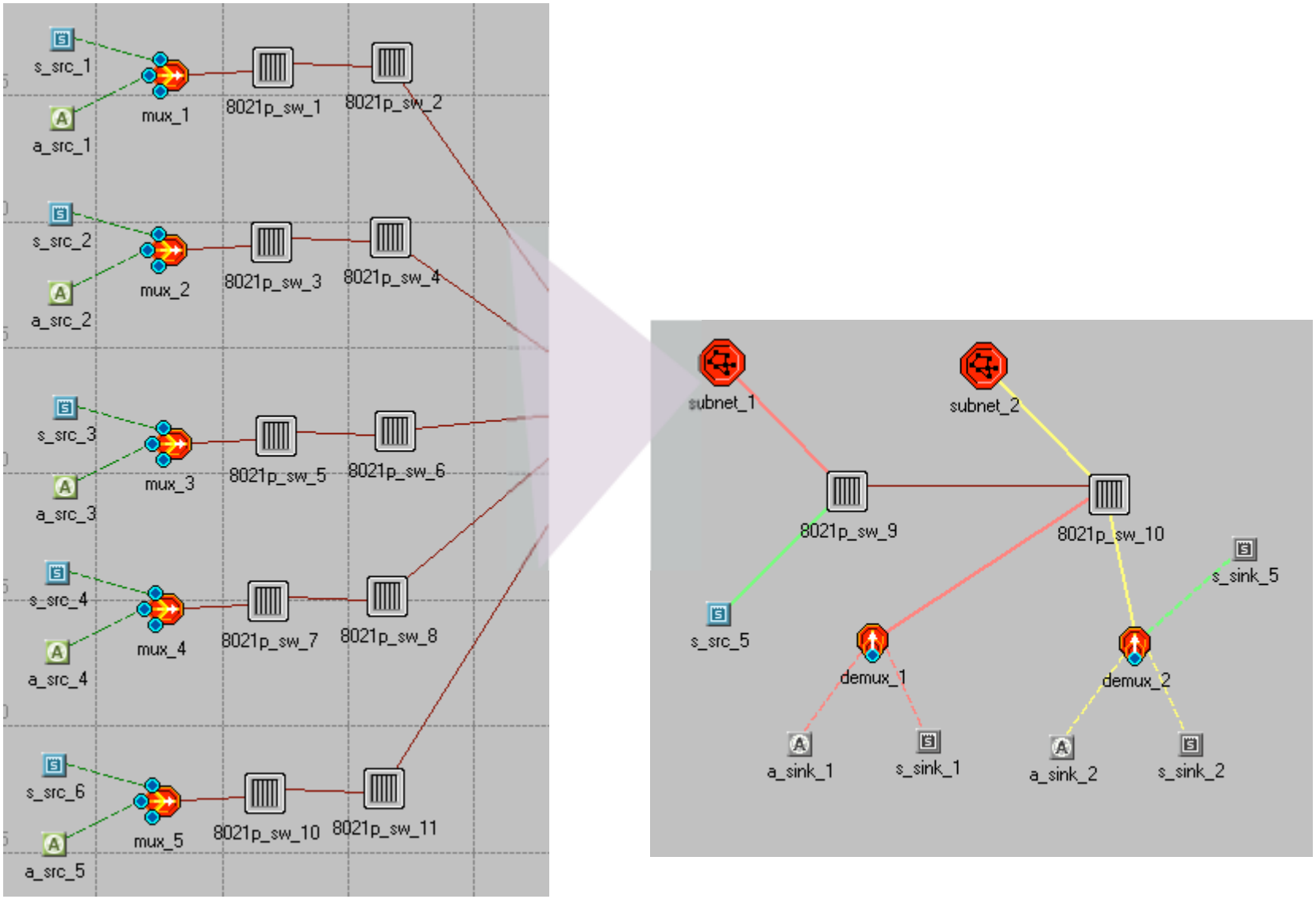
Additional tests: MTU Packets, Poisson Arrival

Poisson arrival,
50packets/second.
Packet size = 12K
With 802.1p switches

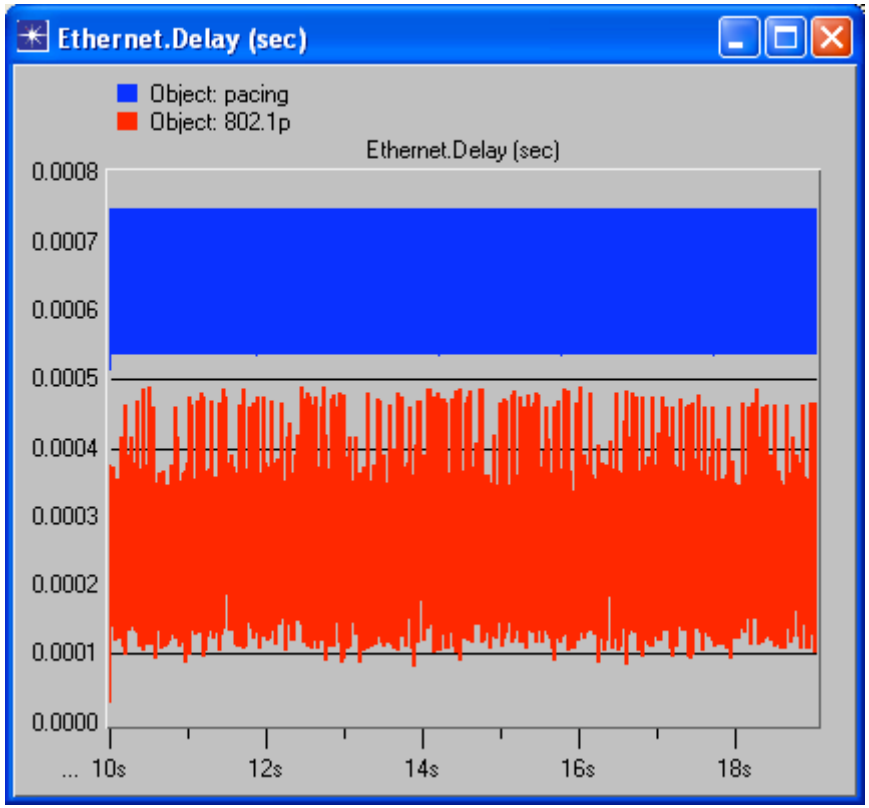
Self-similar, H=0.7.
50packets/second.
Packet size = uniform(1K,12K)



Scenario 4: Multi-hop Scenario



Delay Results



Within the observation period, pacing scheme still has larger delay but smaller delay variation.

- With pacing method, jitter is not accumulated along the multi-hop path.
- With 802.1p method, jitter could be accumulated along the multi-hop path. But as long as timing synchronization is implemented, this jitter can be removed using reasonable size of buffer.

Summary

- In a worst case, 802.1p has larger delay and delay variation than pacing scheme.
  - The worst case delay of 802.1p is related to the number of incoming conflicting streams, the extent of distortion of those conflicting streams, and the cycle size of CBR traffic
  - In our simulated scenarios, the performance differences are marginal
- In a realistic scenario, 802.1p shows a smaller delay, but larger delay variation (jitter).
  - As long as timing synchronization is implemented, this jitter can be removed using reasonable size of buffer. Note that pacing scheme do need some kinds of timing synchronization.