TSN: Lecture 9
Non-blocking Switch

Topics Covered

- Non-blocking Switch Performance
- Switch Performance
- Hybrid solutions
- Multicasting
- Generating and distributing copies
- Header translation

Non-blocking Switch Performance

- Non-blocking Switch with no buffers
  - If output contention occurs, only one among *n* contending packets transmitted, all other dropped
  - Throughput = 63.2%; But remaining is all packet loss!!
- Non-blocking Switch FIFO input buffers
  - Throughput = 58.6%
  - Packet loss is a function of buffer size
  - For a Bernoulli packet arrival process (with a probability *p*)

$$P_{loss} < \frac{p(2-p)}{2(1-p)}\left[\frac{p^2}{2(1-p)^2}\right]^B$$

Switch Performance (contd ..)

- Non-blocking switch with non-FIFO buffers
  - packets are selected from a window (*w*) of buffer to minimize contention

Size	FIFO	Window Size (*w*)							
N		1	2	3	4	5	6	7	8
2	75.0%	75%	84%	89%	92%	93%	94%	95%	96%
4	65.5%	66%	76%	81%	85%	87%	89%	94%	92%
8	61.8%	62%	72%	78%	82%	85%	87%	88%	89%
16		60%	71%	77%	81%	84%	86%	87%	88%
32		59%	70%	76%	80%	83%	85%	87%	88%
64		59%	70%	76%	80%	83%	85%	86%	88%
∞	58.6%								

Switch Performance (contd ..)

- Non-blocking Switch with Output buffers
  - Best performance (100% Throughput) as there is no HOL blocking
  - Delay performance depends on the output queueing
- Non-blocking Switch with Shared buffers
  - Packets lost in contention are stored in a separate buffer that feeds as direct input (depending upon the number of extra inputs)
  - Performance can be close to 100% with large shared buffer
  - Switch size grows

Hybrid solutions

- Buffers at more than one point
- Becomes hard to analyze and manage
- But common in practice

Outline

- Circuit switching
- Packet switching
  - Switch generations
  - Switch fabrics
  - Buffer placement
  - Multicast switches

Multicasting

- Useful to do this in hardware
- Assume port-mapper knows list of outputs
- Incoming packet must be copied to these output ports
- Two subproblems
  - generating and distributing copies
  - VCI translation for the copies

Generating and distributing copies

- Either implicit or explicit
- Implicit
  - suitable for bus-based, ring-based, crossbar, or broadcast switches
  - multiple outputs enabled after placing packet on shared bus
  - used in Paris and Datapath switches
- Explicit
  - need to copy a packet at switch elements
  - use a *copy* network
  - place # of copies in tag
  - element copies to both outputs and decrements count on one of them
  - collect copies at outputs
- Both schemes increase blocking probability

Header translation

- Normally, in-VCI to out-VCI translation can be done either at input or output
- With multicasting, translation easier at output port (why?)
- Use separate port mapping and translation tables
- Input maps a VCI to a set of output ports
- Output port swaps VCI
- Need to do two lookups per packet

Packet Size Impacts

- Fixed Length Packets
- Variable Length Packets