Node architectures for optical packet and burst switching

Chris Develder,
Jan Cheyns, Erik Van Breusegem, Elise Baert,
Ann Ackaert, Mario Pickavet, Piet Demeester

Dept. of Information Technology (INTEC)
Ghent University, Belgium

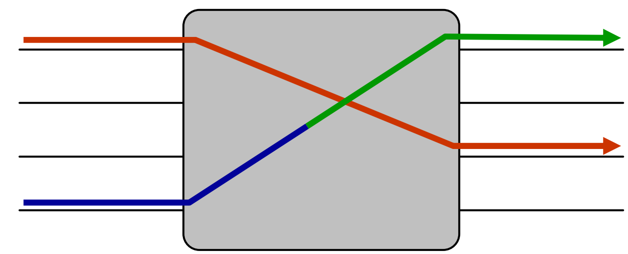
Outline

- Introduction
  - OPS and OBS concepts
  - packet & header format
- Node architecture
  - functionality
  - optics and/or electronics
- Switch matrix
  - alternatives
  - scalability
- Contention resolution
  - problem & solution
  - buffer architectures

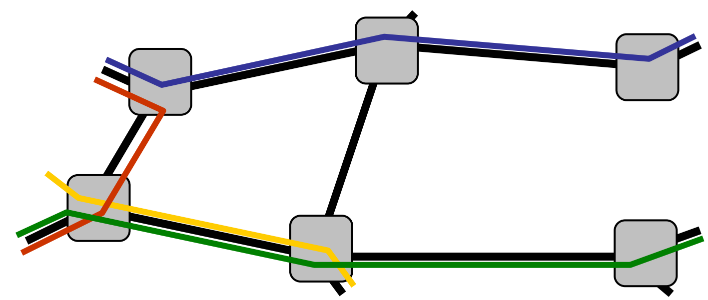
Introduction

Optical switches

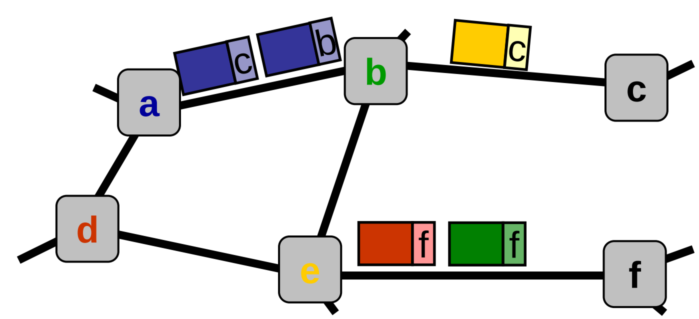
- Optical switching:
  - direct light from an input port to an output port
  - possibly wavelength conversion



- circuit-switching:
  - continuous bit-stream
  - pre-established light-paths
  - set-up: “manually” or dynamic



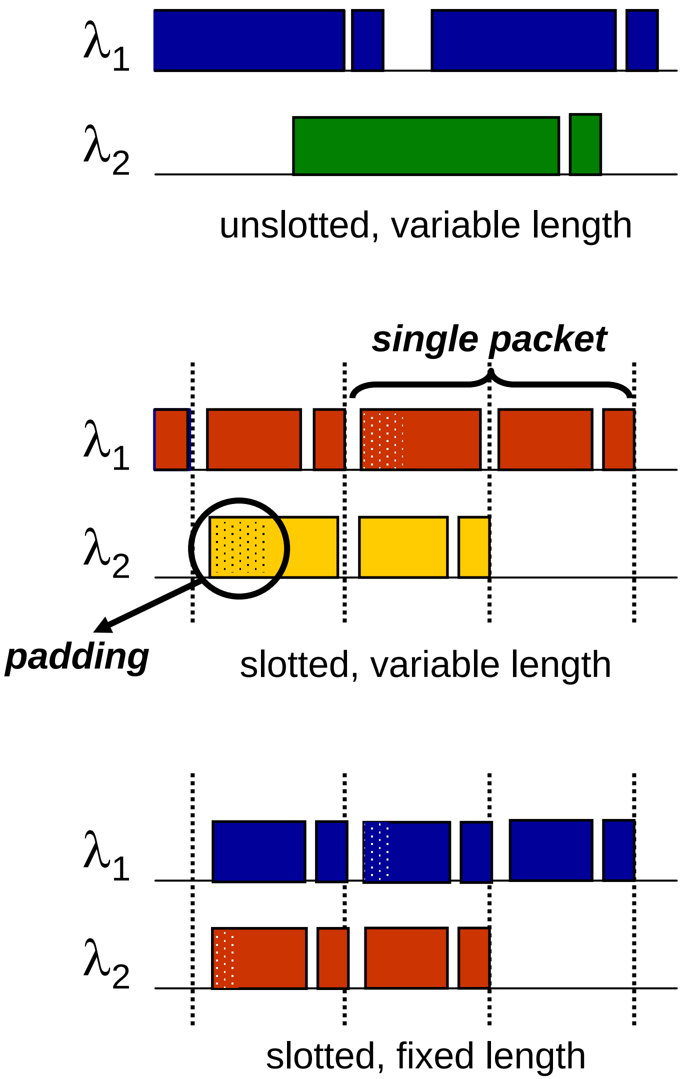
- packet/burst switching
  - chunks of bits, encapsulated in packets
  - packet header determines forwarding
  - e.g. label switching, GMPLS based



Packet format

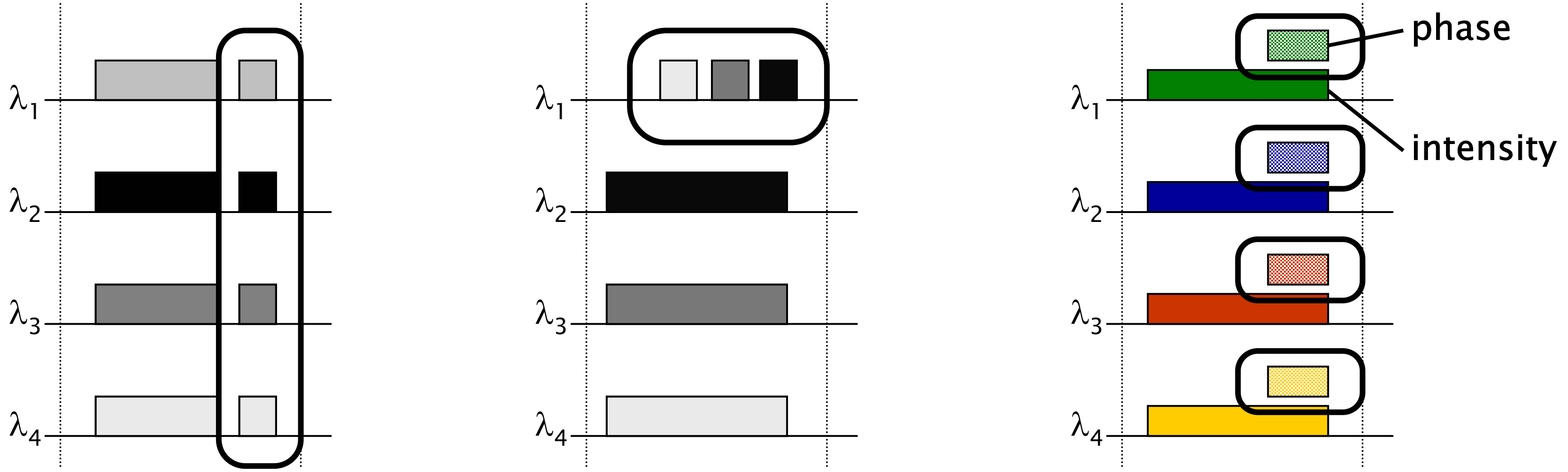
- fixed/variable duration:
  - pro variable = no fragmentation/ reassembly, no padding, less header overhead
  - contra = long packets can block many short ones
- slotted/unslotted operation:
  - pro slotted = easier packet scheduling (synchronous switching)
  - contra = cost of synchronisation components

- OPS = fixed, slotted packets
- OBS = variable, unslotted packets



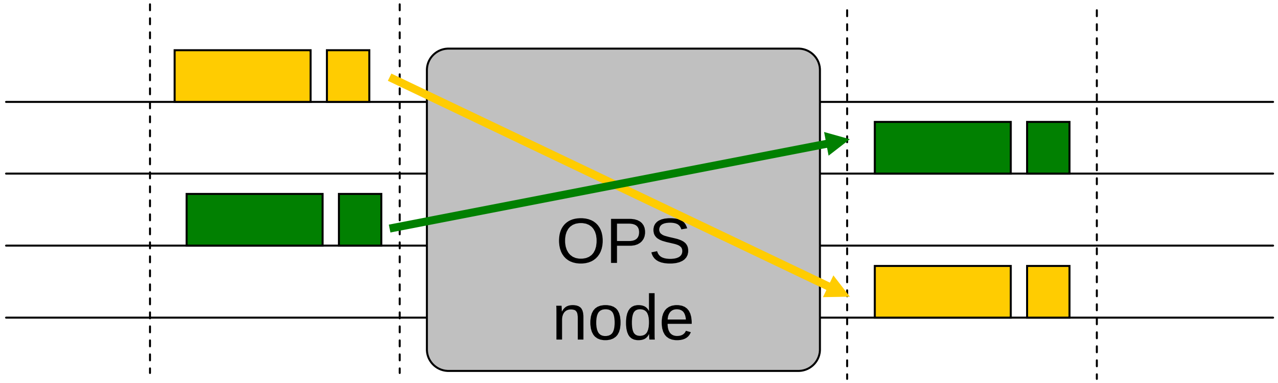
Header format

- position of header:
  - in band: header and payload are sent sequentially, separated in time
  - out of band: dedicated wavelength; also multi-wavelength headers have been proposed (see e.g. PS.TuC1)
  - out of band: orthogonal channel (e.g. DPSK)



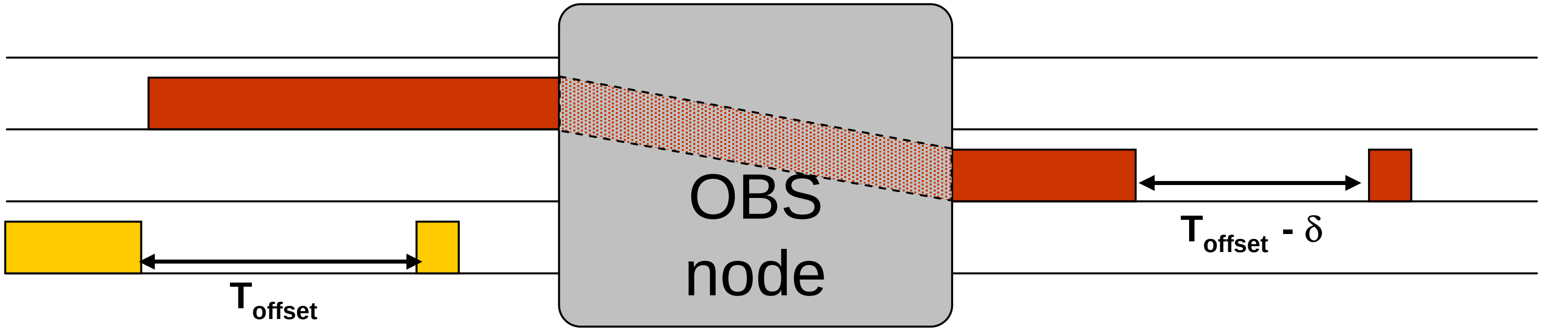
Operation of OPS

- <u>fixed-length</u> packets, <u>slotted</u> operation
- header accompanies payload
    - contains necessary information to make forwarding decision
- each timeslot:
    - inspect packets at input ports
    - decide which packets can be forwarded without collisions
- switch is “memory-less”
    - no knowledge of packets scheduled in past is necessary



Operation of OBS (1)

- variable packet lengths, unslotted operation
- header is sent T_{offset} before payload
  - contains necessary information to make forwarding decision
  - functions as one-way reservation (allows timely config. of switch fabric)
  - offset decreases by header processing time per hop
- on arrival of header:
  - decide whether burst can be forwarded without collisions
  - make necessary resource reservations if burst is accepted
- switch needs “memory”:
  - keep track of reservations made in past



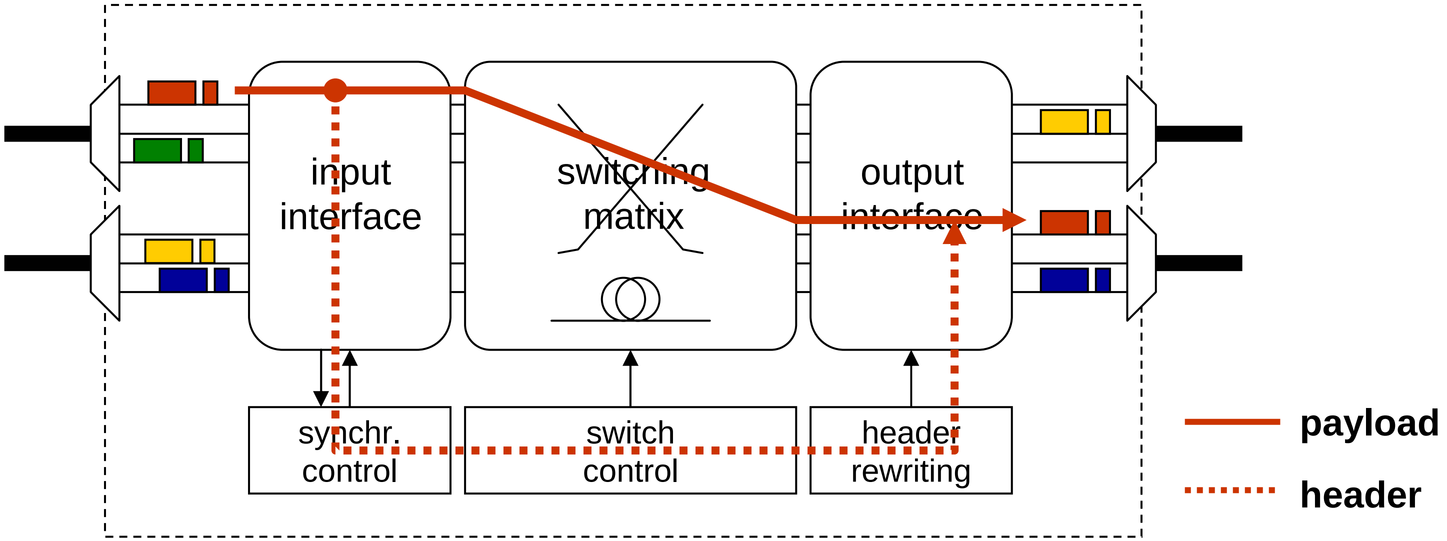
Operation of OBS (2)

- Note: WR-OBS = wavelength-routed OBS
  - two-way reservation
    - “header” is sent from source to destination within OBS network
    - if all goes well, acknowledgement is sent back to source
  - this is more like wavelength switching at very short timescales
  - proposed by P. Bayvel et al. (Univ. College London, UK)

Node architectures

Functionality of an OPS / OBS node

- input interface:
  - header extraction (straightforward if out-of-band)
  - synchronisation: detect beginning of packet/burst
  - in OPS: align packets
- switching matrix (see further)
- output interface
  - e.g. regeneration of optical signal; header re-writing…

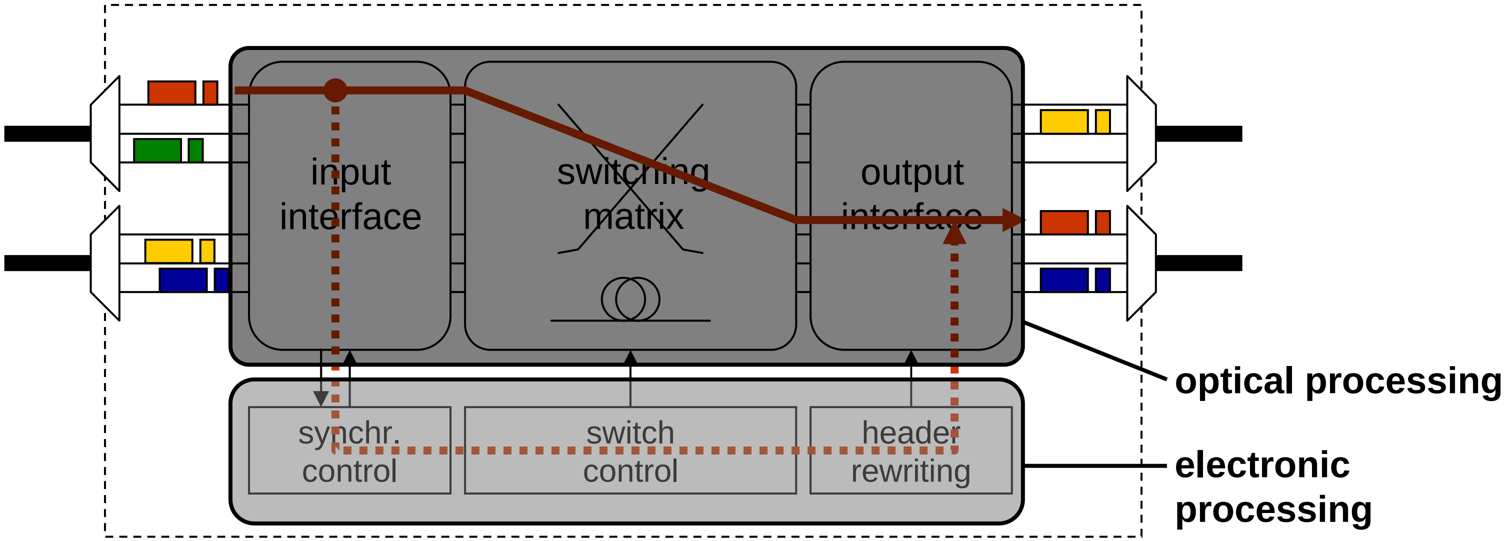


Combine the best of electronics and Optics

- optical packet switching “today”:
  - header is processed electronically
  - payload is switched optically

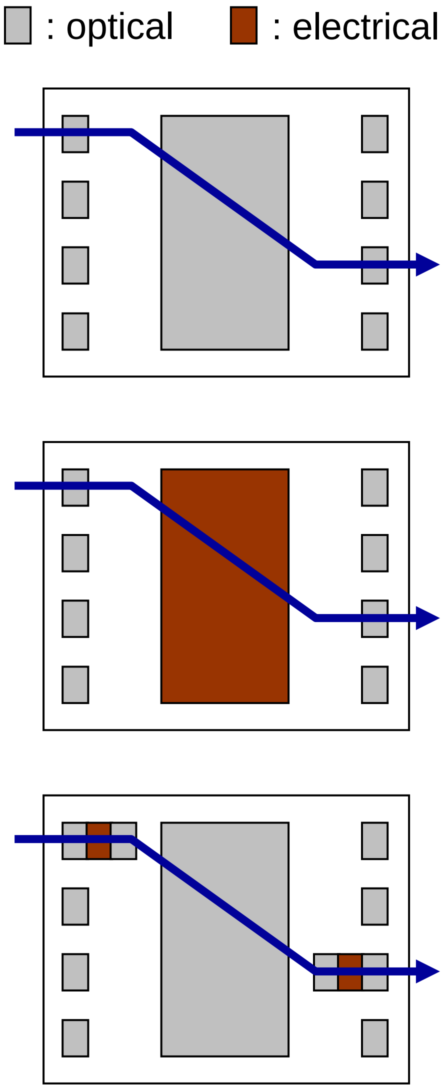
  ⇒ <u>optics</u> for capacity & switching,
  <u>electronics</u> for routing & forwarding

- note:
  - all-optical header processing is “under study” (e.g. PS2001, PThD5)



Oh-oh-oh*

- O-O-O:
  - optical input interface, optical switch fabric, optical output interface
  - payload is switched transparently, without leaving optical domain
  - **pro** = bitrate-transparent
  - **con** = still emerging technologies, BER can not be monitored
- O-E-O:
  - optical inputs, converted to el. for switching, back to optics at outputs
  - "opaque": no more all-optical;   but straightforward grooming
  - **pro** = well established techno, 3R regen. "for free"
  - **con** = no bit-rate transparency, scalability ~ Moore's law
- OEO-O-OEO
  - (some) inputs and outputs: electronic 3R regen.
  - **pro** = scalability of optical switch fabric with 3R regen.
  - **con** = bit more complex than O-O-O

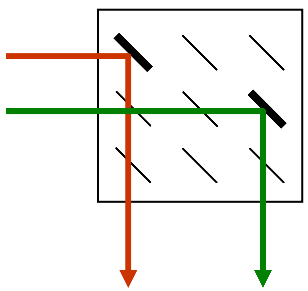


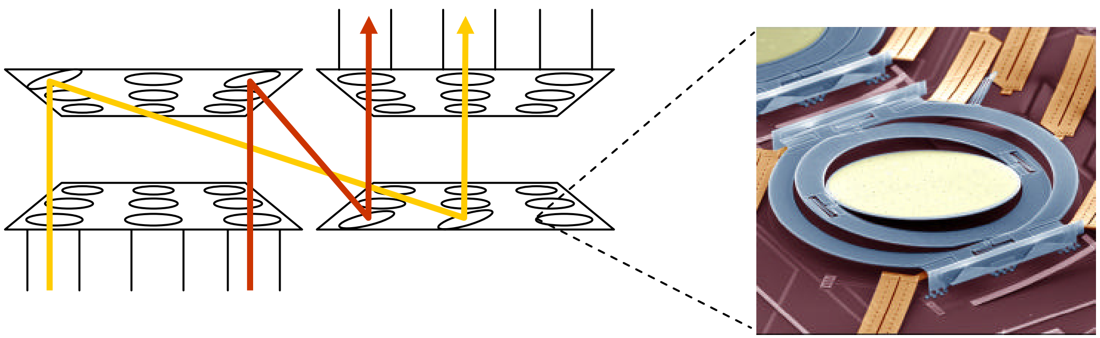
*: G.Bennet, at www.lightreading.com

Switch fabric

Overview of dominant architectures

- dominant approaches to optical switch fabrics:
  - MEMS (=micro-electro-mechanical systems)
  - broadcast-and-select (e.g. SOA-based)
  - AWG and tuneable lasers
- MEMS:
  - principle:



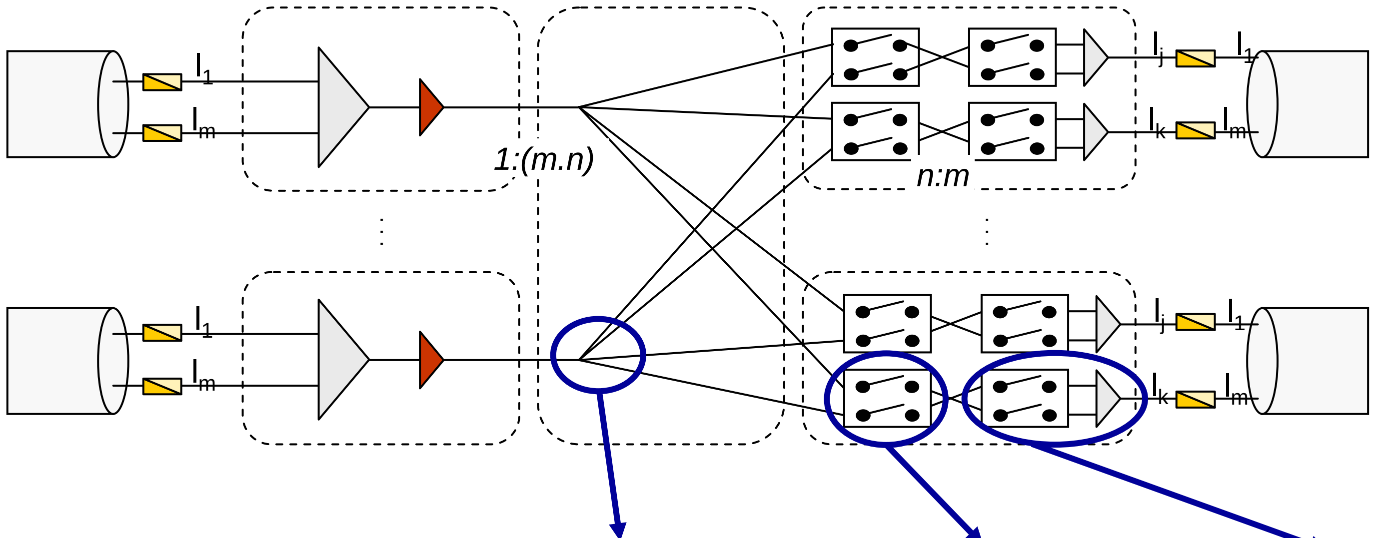


© Lucent, e.g. OFC2002

  - main components:
    - tiny mirrors (2D pop-up, or 3D tilting)
  - characteristics
    - low loss, good scalability (=high port counts)
    - but… too slow for packet switching (ns timescale not feasible)

Broadcast-and-select switch

- principle: (e.g. David, http://david.com.dtu.dk; ECOC'01)



- main components:
  - splitters
  - selectors
- characteristics:
  - split losses: calls for regeneration
  - inherent multicast capability

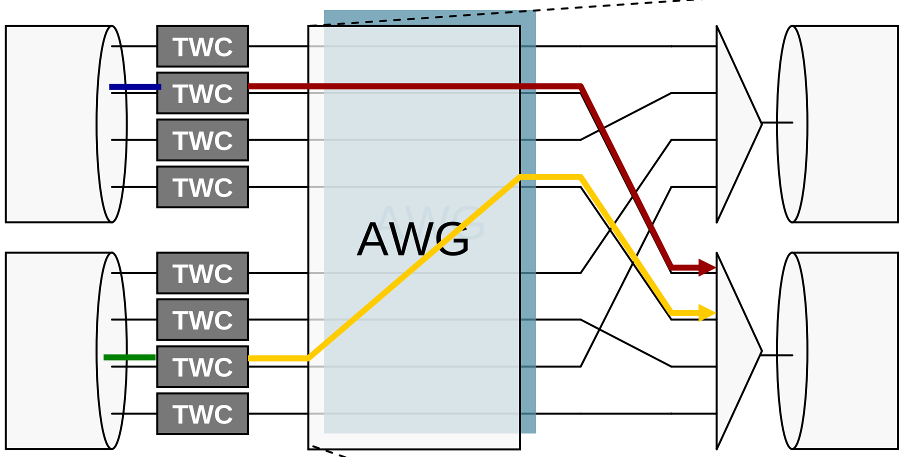
complete fibre is broadcast to all output ports

first array of SOAs selects 1 input fibre

second array of SOAs is used to select 1 wavelength

AWG based switch

- principle: (e.g. Stolas, http://www.ist-stolas.org)

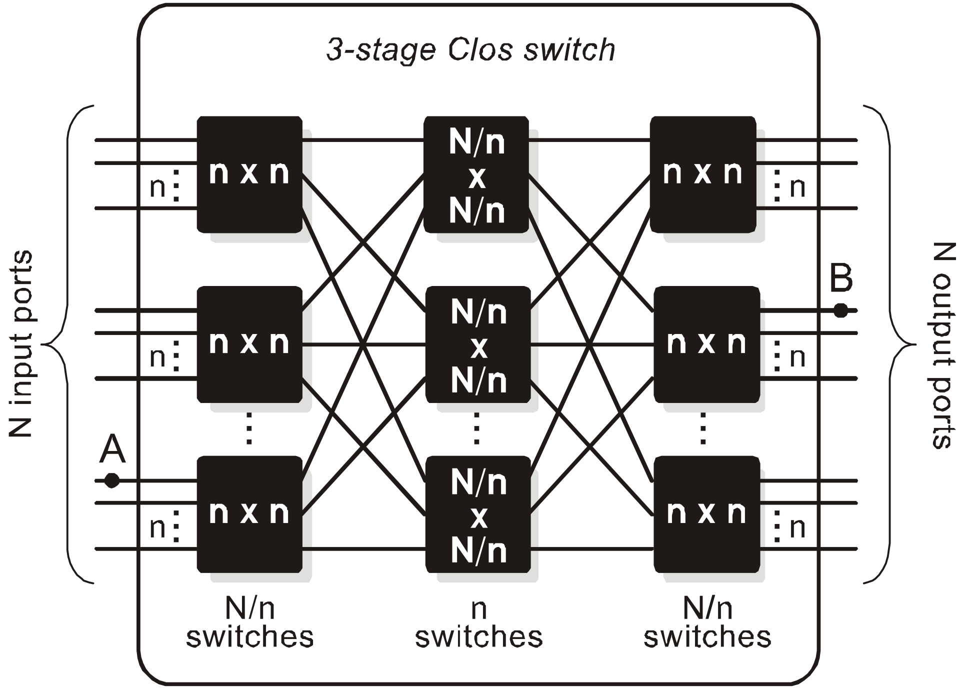


out-port \ in-port	1	2	3	4	5	6	7	8
1	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7	λ_8
2	λ_8	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7
3	λ_7	λ_8	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6
4	λ_6	λ_7	λ_8	λ_1	λ_2	λ_3	λ_4	λ_5
5	λ_5	λ_6	λ_7	λ_8	λ_1	λ_2	λ_3	λ_4
6	λ_4	λ_5	λ_6	λ_7	λ_8	λ_1	λ_2	λ_3
7	λ_3	λ_4	λ_5	λ_6	λ_7	λ_8	λ_1	λ_2
8	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7	λ_8	λ_1

- main components:
  - tuneable wavelength converters
  - AWG
- characteristics:
  - passive component, no split losses
  - multicast quite complex

Scalability

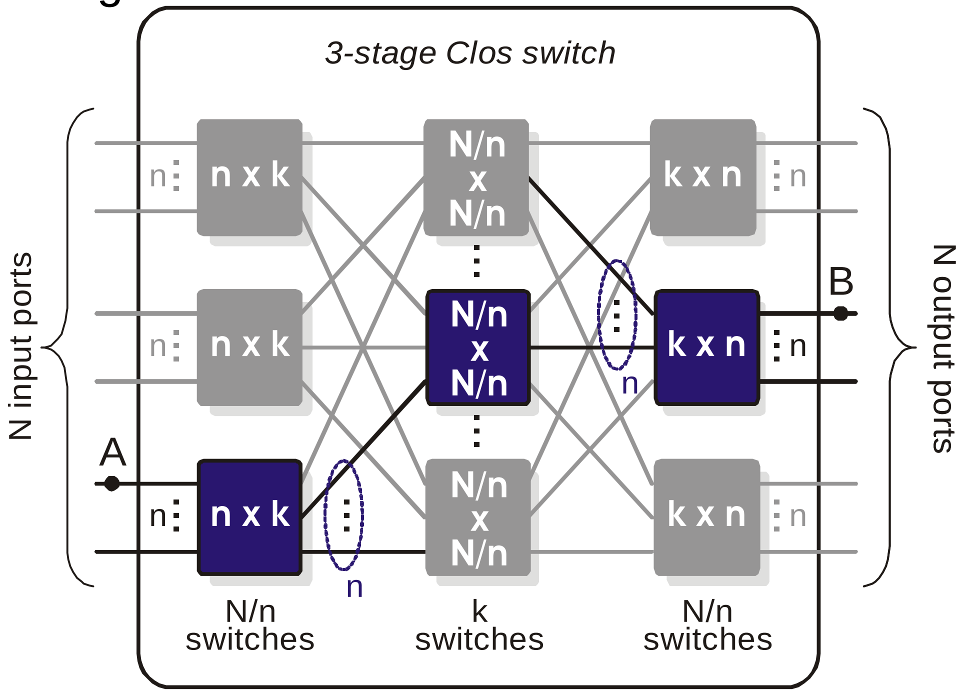
- switch dimensions limited by
  - B&S: split losses
  - AWG: tuneability range of TWCs
- Possible solution:
  - multi-stage switches, e.g. Clos-networks



A three-stage Clos-network (1)

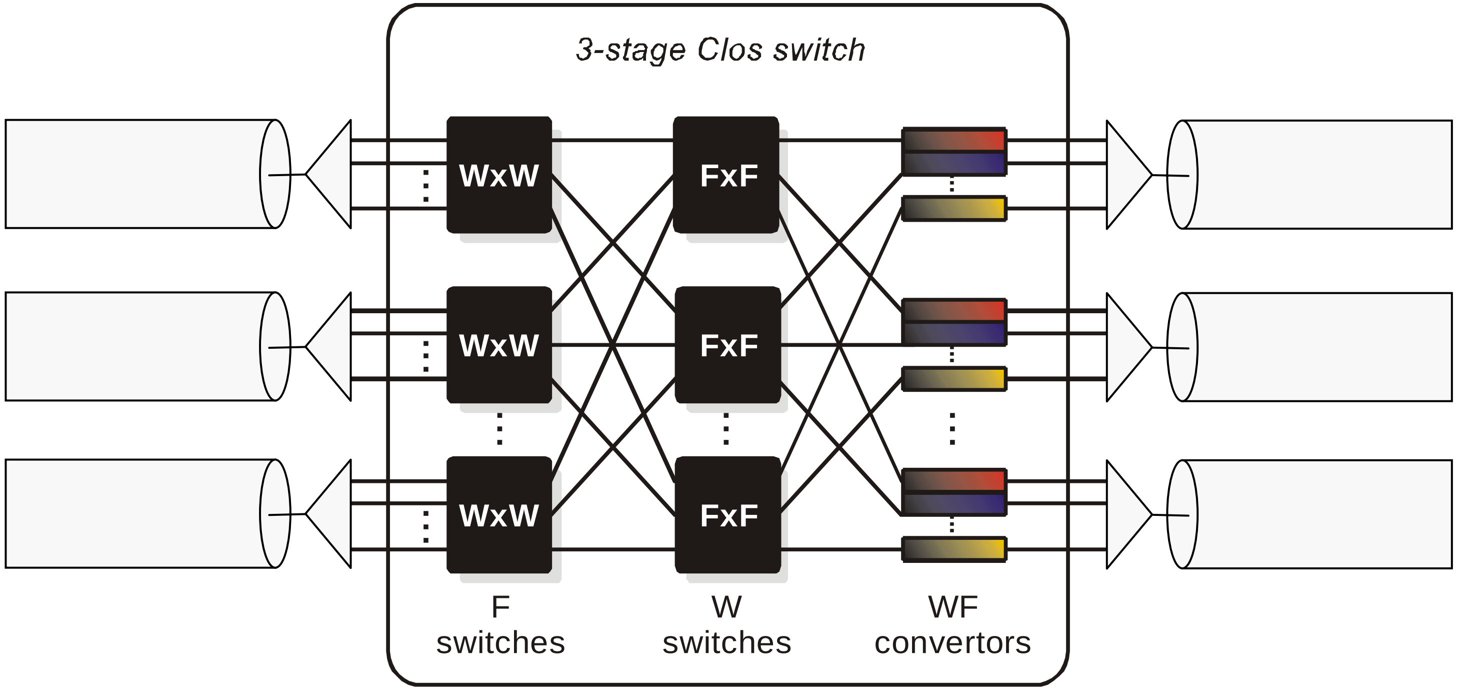
- slotted OPS
  - re-arrangeable non-blocking
  - second-stage switches: k ≥ n
- unslotted OBS
  - fully non-blocking
  - second-stage switches: k ≥ 2n-1

⇒ slotted OPS needs only half as much second stage switches



A three-stage Clos-network (2)

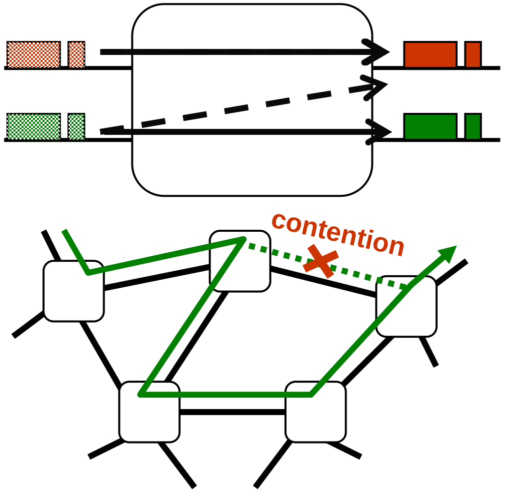
- <u>note</u>: if wavelength conversion is allowed, third switching stage can be eliminated
- e.g. slotted OPS:
  - F fibres, W wavelengths per fibre
  - make 1st and 3rd stage per input/output fibre

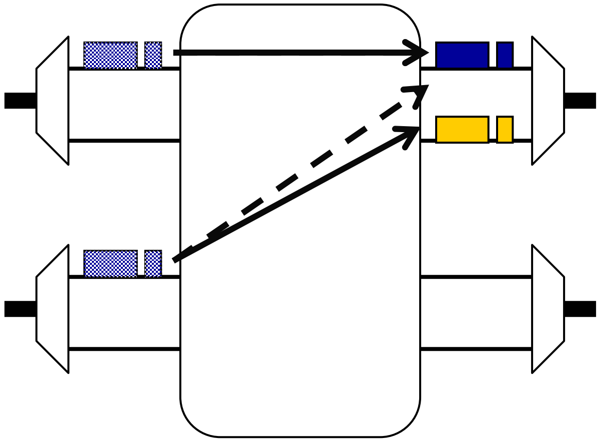


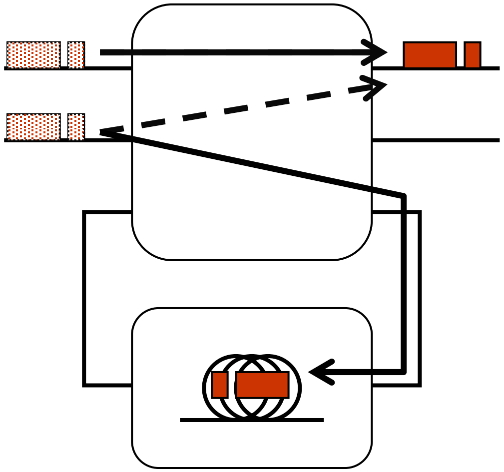
Contention resolution

Problem and possible solutions

- Problem:
  - two or more packets contend for same resource: destined for same outgoing port at the same time
- Solutions:
  - deflection routing
  - wavelength conversion
  - buffering: optical buffer = Fibre Delay Lines (FDLs)





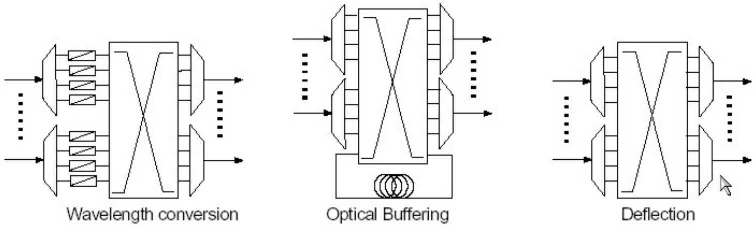


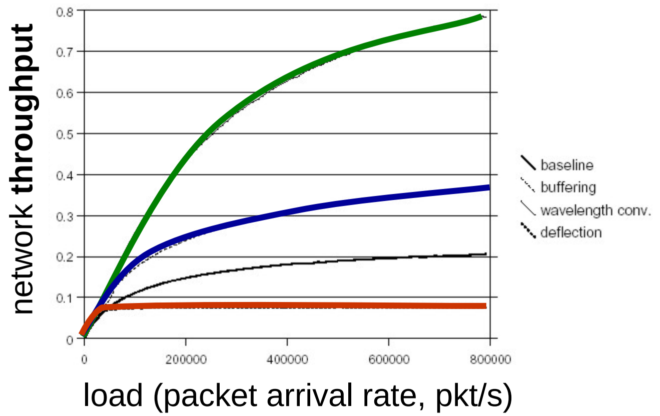
What solution to choose?

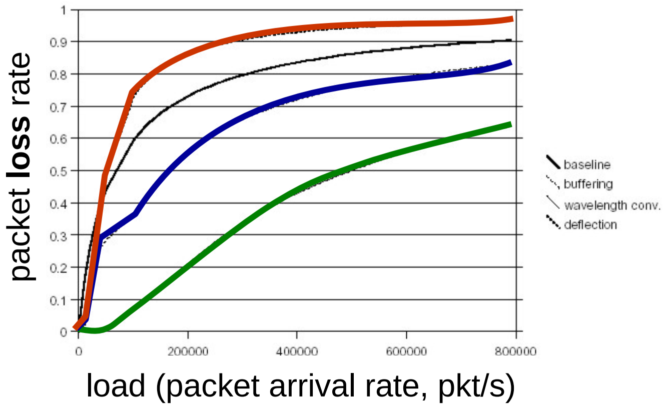
figures © Yao et al., Opticomm'00

- **Deflection:**
  - packets are "stored" in network:
  - increases load, increases delay
  - only works for low loads
- **Wavelength conversion:**
  - no packet storage
  - allows high network throughput, no increased delay
- **Buffering:**
  - local packet storage at nodes
  - small delay penalty

⇒ **Use combination of wavelength conversion and buffers**



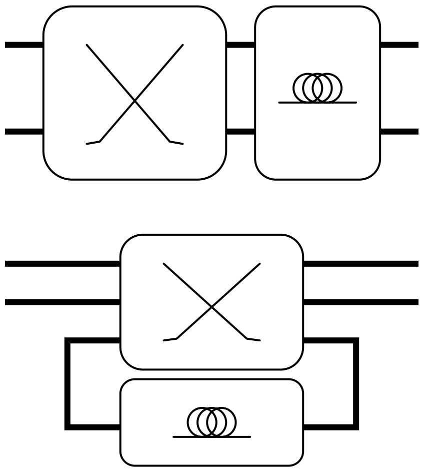




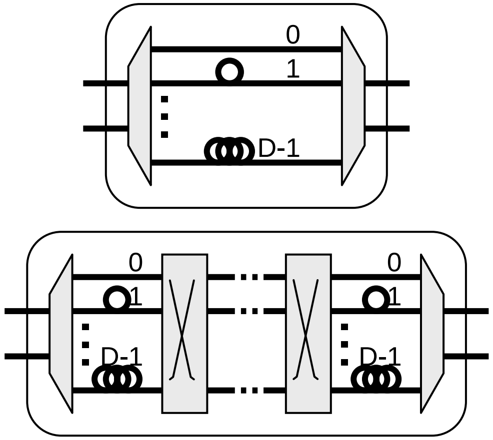
Buffer architectures

- feed-forward vs feed-back
  - feed-forward: input or output buffering
  - feed-back: shared, recirculating FDLs
- single stage vs multiple stage
  - multiple stages separated by switching elements
  - e.g.: each stage different delay resolution (“units”, “tens”, “hundreds”…)
- choice of FDL lengths
  - multiple FDLs: multiples of “unit”, resolution D
  - uniform/non-uniform (D,2D,3D or something else)

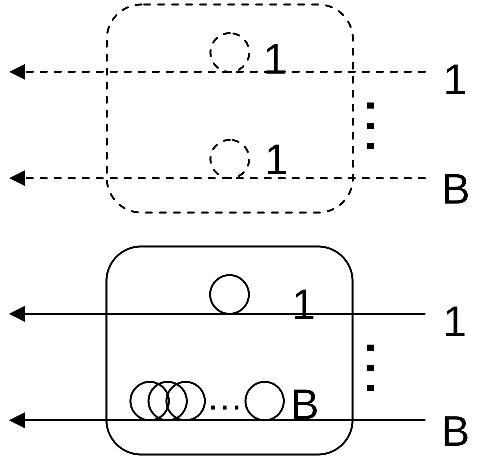
feed-forward vs feed-back



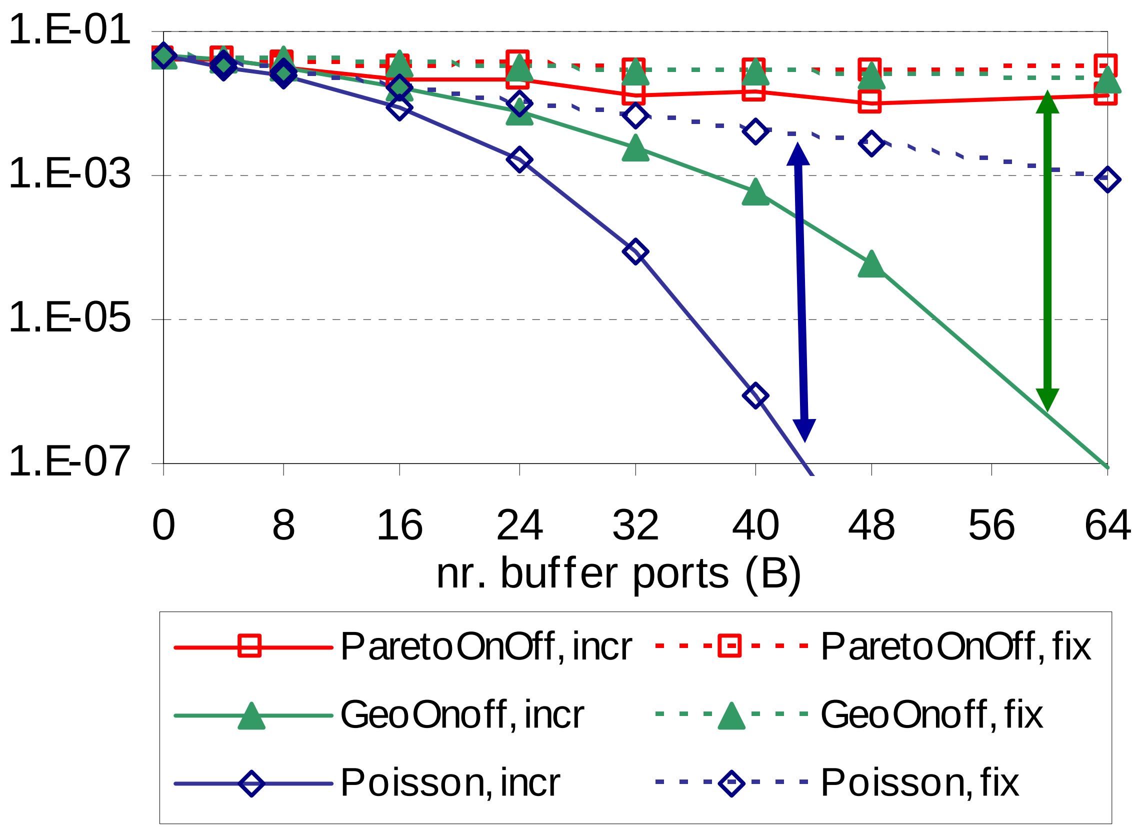
single vs multiple stages



OPS / OBS: fixed vs increasing FDLs



- sample results for fixed-length, slotted OPS
- **Increasing** FDL lengths give far lower PLRs (order of magnitude or more)
- “penalty”: reordering of packets, higher delays



see COIN.TuD1 for details

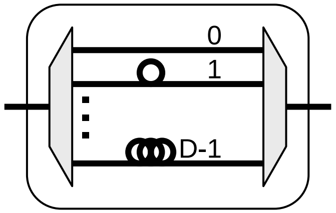
OBS: choice of granularity

- Optimal granularity D:

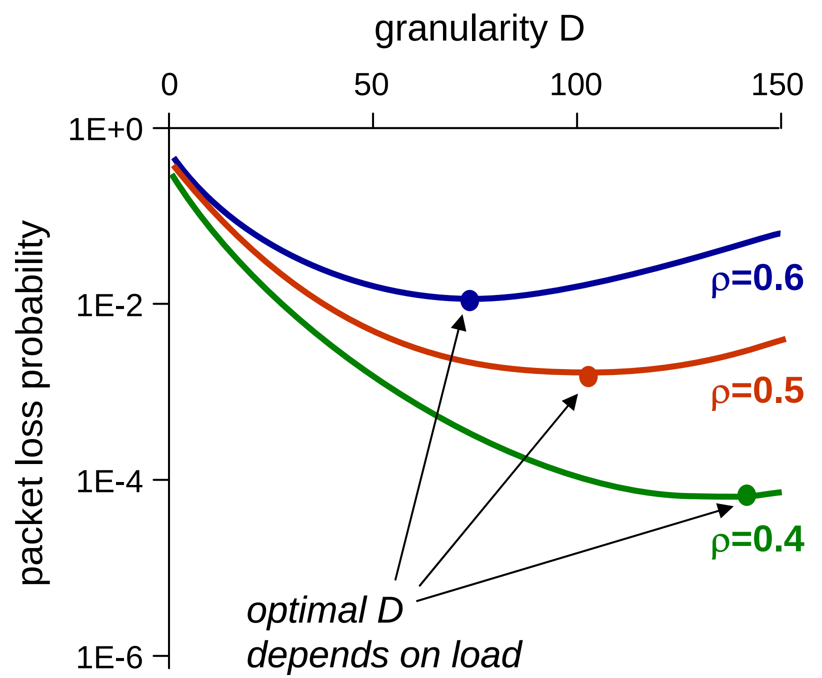
⇒ Non-trivial choice!

- Tradeoff between resolution and buffering capacity
  - small D: small gaps, but limited buffer depth
  - large D: large buffer depth, but large gaps between packets

- Function of
  - load
  - traffic profile (packet size distribution, burstiness…)



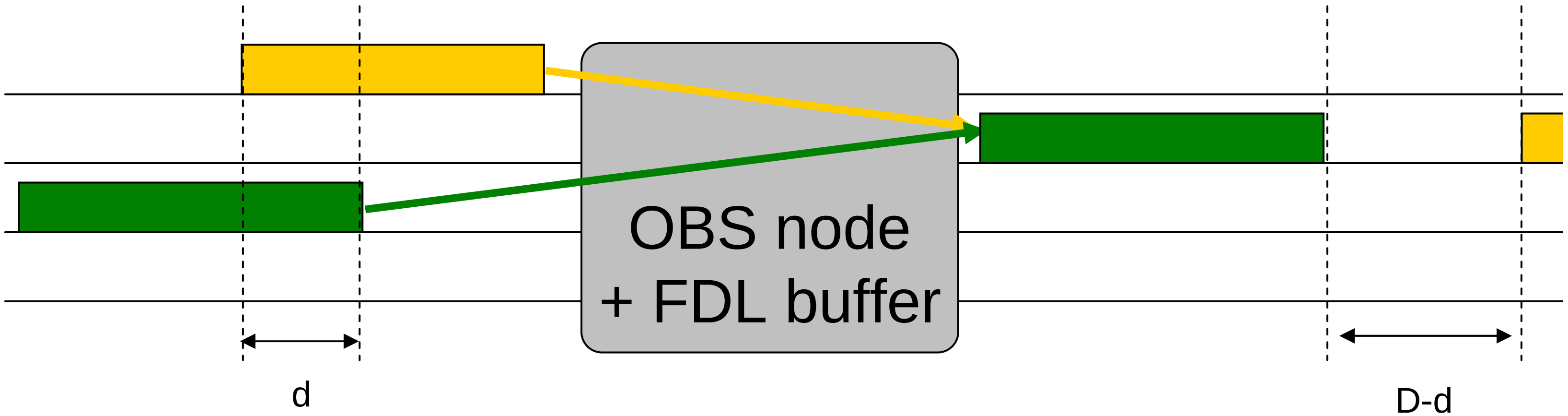
sample results for slotted switch with output buffering, single wavelength per fibre, geometric distr. packet lengths, bernouilli arrivals, 20 FDL lengths



OBS: void filling

- Void creation in case of variable packet-lengths (OBS):
  - FDL buffer offers discrete set of delays (in contrast to e.g. electronic RAM)
  - gaps between successive packets: “voids”

$\Rightarrow$ Need for intelligent buffer scheduling if we want to improve link throughput (i.e. void filling)



example: we need delay d, but FDL only offers D $\Rightarrow$ gap of D-d is created; void filling will attempt to insert packet in this gap

Contention resolution

- wavelength conversion greatly reduces need for buffering
- feed-back architecture allows sharing of FDL resources among all output ports
- when using different delay lengths, this calls for more intelligent buffer scheduling; for variable-length packets (OBS) the issue of void creation arises
- OBS: choice of delay lengths, i.e. the FDL granularity, is non-trivial issue

Conclusions

Summary

- reviewed OPS/OBS concepts
  - OBS as possible 1st step
  - OPS: fully exploit fast switching technologies
- switch architectures:
  - MEMS: too slow for OPS
  - broadcast & select: allows multicast, but splitting losses
  - AWG: passive core component, but no multicast
  - multi-stage architecture to increase scalability (OPS fewer switching elements than OBS)
- buffering:
  - push buffers to network edges
  - use FDLs to lower PLR in core
  - non-trivial choice of FDL granularity for OBS
  - quite complex scheduling for OBS

That’s all, folks!

… thanks for your attention