Using shared wavelength converters effectively in
OPTICAL SWITCHING

Kurt Van Hautegem

joint work with Wouter Rogiest and Herwig Bruneel

Faculty of Engineering and Architecture ♦ Department of Telecommunication and Information Processing ♦ SMACS research group ♦ Kurt.VanHautegem@Ugent.be

OPTICAL BACKBONE

technological developments &

internet-based business models

⬇

demand for bandwidth ↗

nowadays:

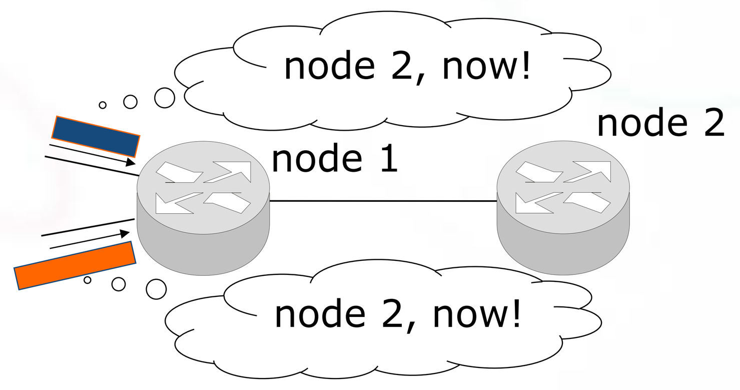
- unlimited fiber capacity
- switching bottleneck

optical burst/packet switching:

- no dedicated communication channel

✓ use of available capacity ↗

× contention ➡ packet loss



⬇

need for **contention resolution**

CONTENTION RESOLUTION

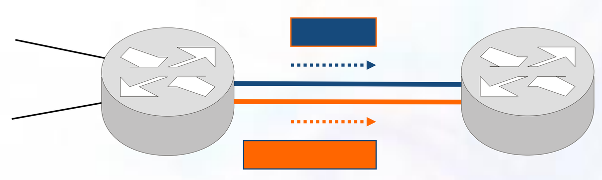
electronic buffering(RAM): too slow for optical speeds

⬇

optical contention resolution

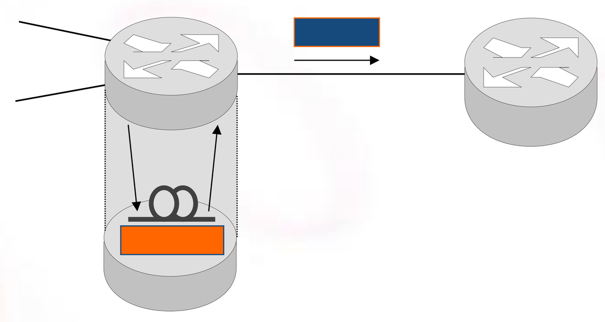
⬋ ⬊

wavelength converters (WCs)



- packets arrive on c ≠ wavelengths
- r WCs to schedule packets on same set of c wavelengths

Fiber Delay Lines (FDLs)

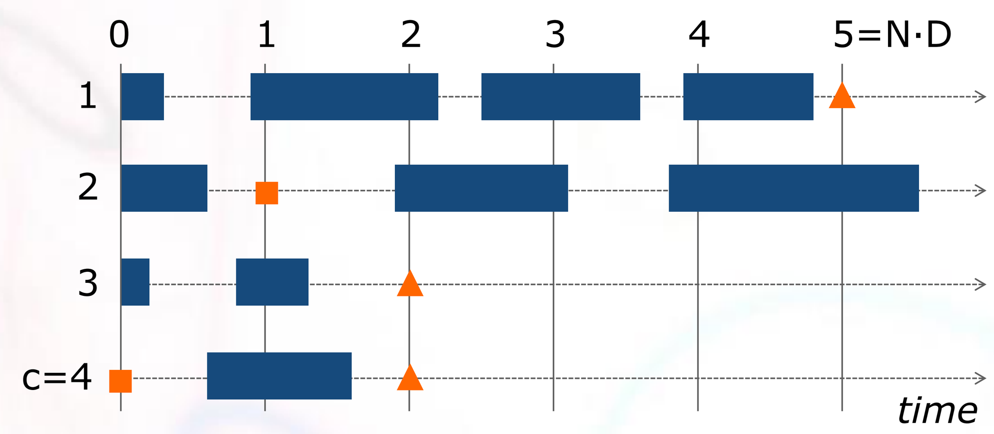


- set of fibers, #=N+1
- lengths j·*D*, j=0...N
- N = buffer size
- D = granularity

⬊ ⬋

provisional schedule

- updated at every arrival
- shows already scheduled packets
- horizontal lines: outgoing wavelengths (c=4)
- vertical lines: delays of FDLs (N=5, D=1)



SCHEDULING BASICS

choose:

- outgoing wavelength i (i=1...c)
- delay line j (j=0...N)

constraints:

- availability of wavelength converter
- no overlap
- type of algorithm: NVF ▲

  VF ▲ + ■

(N)VF = (non-)void-filling

satisfied: Scheduling Points (SPs)

goal: minimize loss probability (LP)

⬇

choose 1 SP

currently best algorithms:

- NVF: minimum gap

gap = unscheduled length before SP

- VF: minimum delay

delay = FDL of SP

4 NEW ALGORITHMS

- assign cost to each SP
- choose SP with lowest cost
- v: # available WCs (v=1...r)
- w: arriving wavelength (w=1...c)
- α, β & ε: algorithm parameters

→ optimised for minimal LP

2 cost functions:

C: cost of SP:

$$C = \alpha \cdot gap + (1-\alpha) \cdot delay$$

➡ algorithms **C-NVF and C-VF**

CW: cost of SP:

$$CW = \left(\frac{1}{1+\beta}\right)^{1-\delta_{wi}} \cdot [\alpha \cdot gap + (1-\alpha) \cdot delay] + \frac{\beta}{1+\beta} \cdot D \cdot [1-\delta_{wi}] \cdot \varepsilon^{v-1}$$

extra summound to penalise use of WC

➡ algorithms **CW-NVF and CW-VF**

RESULTS

- N=9, D=100, c=4, r=1...4
- inter-arrival time: Poisson (average E[T])
- packet size: exponential (E[B]=100)
- load= ρ = E[B] / (c·E[T]) = 80 %
- Monte Carlo simulation

LP reduction with respect to currently best NVF and VF algorithms:

r	1	2	3	4
C-NVF	2,5 %	3,5 %	4,1 %	5,2 %
C-VF	0,8 %	3,4 %	5,2 %	8,9 %
CW-NVF	8,8 %	12,4 %	12,7 %	10,4 %
CW-VF	15,0 %	25,3 %	29,3 %	28,9 %

CONCLUSIONS

- C-NVF & C-VF: weighted average delay & gap

➡ performance ↗

- CW-NVF & CW-VF: weighted average delay & gap + penalised use of WC (energy consumer)

➡ performance ↗↗ + energy consumption ↘