

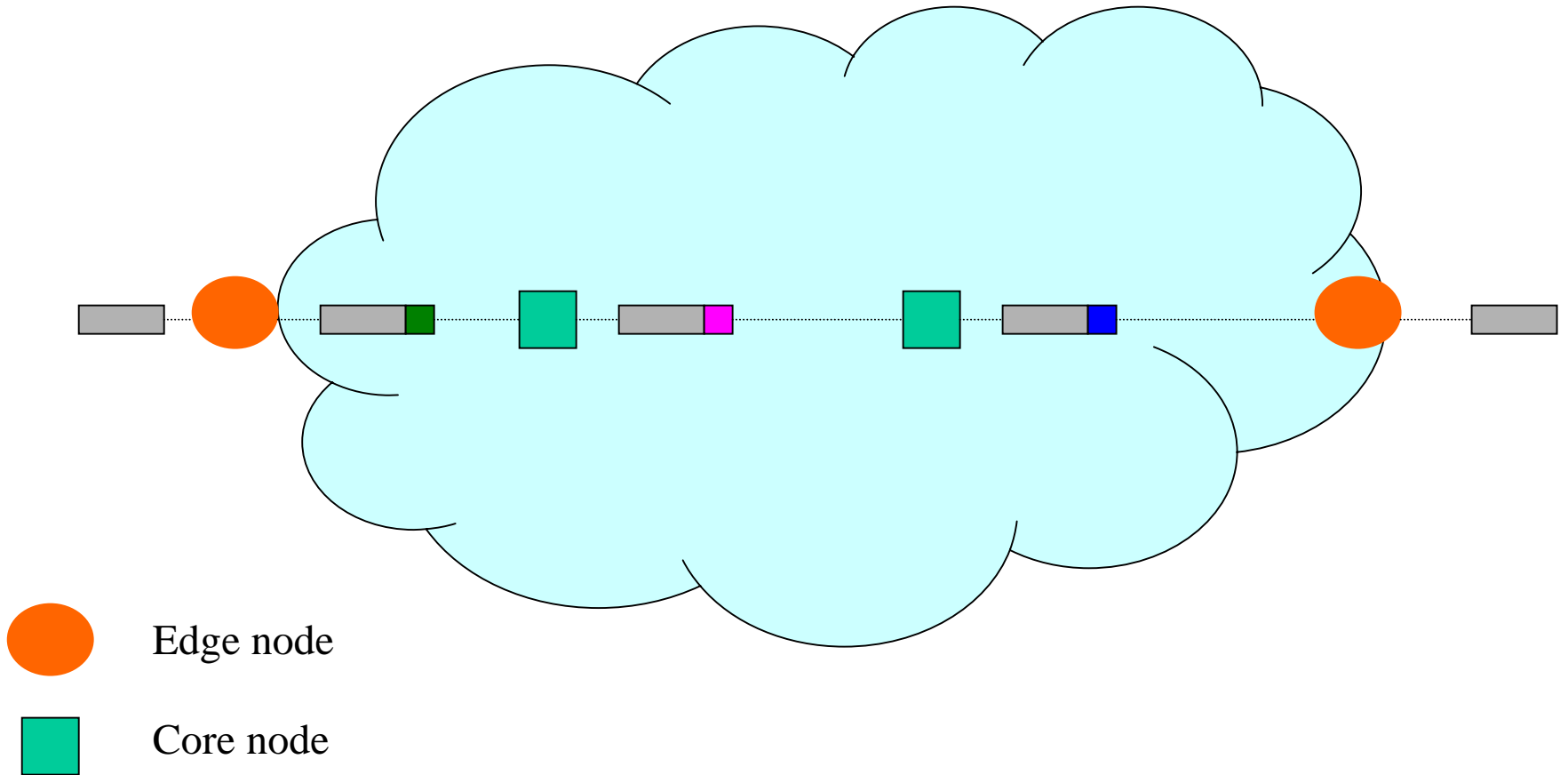
Predictable services in core-stateless networks

Avadora Dumitrescu

Dynamic Packet State (DPS)

- instead of having core nodes maintain per-flow state → packets carry this state (DPS)
- the core routers process each incoming packet based only on the packet header and its own internal state
- core nodes update their internal state and the state from the packet header

Core-Stateless (CS) network



Fair – Bandwidth Allocation Control

- some applications may need minimum bandwidth assurance
- adherence to the given throughput guarantees as long as traffic with reservation remains ‘in-profile’
- provide access to *a fair share* of the ‘excess’ (beyond the aggregate bandwidth of in-profile traffic) output link bandwidth, as expressed by the ‘max-min fairness’ principle
 - for the flows without reservation
 - for the out-of-profile fraction of the flows with reservation

Fair – BAC: Key insights

- congestion not defined by buffer occupancy (w/w thresholds)
- non-congestion associated with buffer being emptied

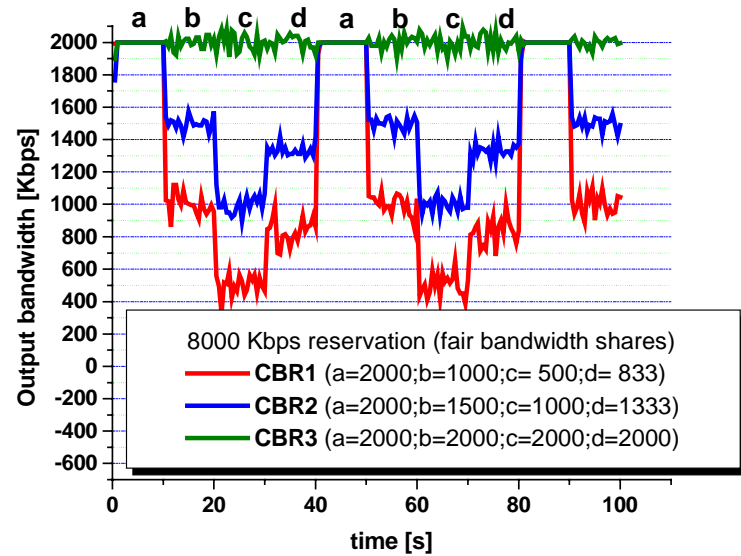
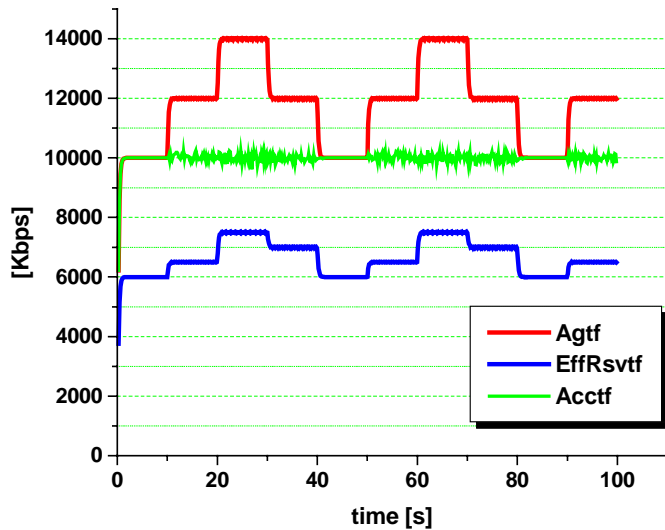
$$Agtf < C$$

=> all packets are forwarded

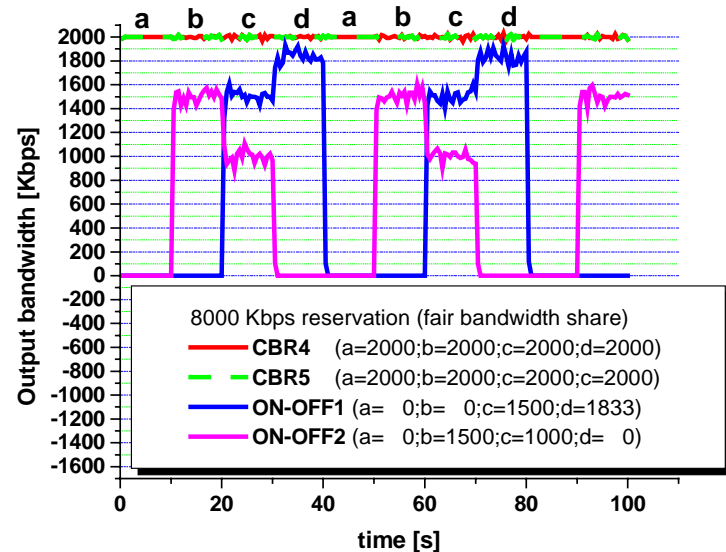
- congestion associated with the buffer filling up

$$fp = \left\{ \begin{array}{l} \min \left\{ 1, \frac{fr}{abr} \cdot \frac{Buffer_availability}{Buffer_occupancy} \cdot coef \right\} \quad \text{for } rsv = 0 \\ 1 \quad \text{for } rsv > 0 \ \& \ abr \leq rsv \\ \min \left\{ 1, \frac{rsv + fr \cdot \frac{Buffer_availability}{Buffer_occupancy} \cdot coef}{abr} \right\} \quad \text{for } rsv > 0 \ \& \ abr > rsv \end{array} \right\}$$

FAIR-BAC Simulation Example



- the bandwidth reservations are granted
- in-profile traffic is isolated from congestion effects
- the bandwidth capability left after providing for all in-profile traffic with reservation is fairly distributed
- different reservation and traffic contexts induce a differentiation in bandwidth allocation while assuring the bandwidth guarantees and the fair allocation of the excess bandwidth



Delay management in CS networks

- end-to-end delay performance for delay-constrained applications is strongly influenced by the choice of the packet scheduling algorithm employed in the nodes along the path.
- to offer strict delay bounds → per-flow information and/or tight requirements on resource allocation and admission control

Core-Stateless Earliest Deadline First (CS-EDF)

- delay management information carried by the packet:
 1. maximum remaining delay (initialized by the requesting application with the maximum acceptable delay), *maxrd*;
 2. time-stamp (initialized at the moment when the packet was emitted), *tstmp*;
 3. maximum remaining number of hops to destination, *maxrnh+1*.

CS-EDF – cont.

- delay management information carried by the packet is updated at packet arrival to a node according to:

$$\mathit{maxrd} = \mathit{maxrd} - (\mathit{current_time} - \mathit{tstamp})$$

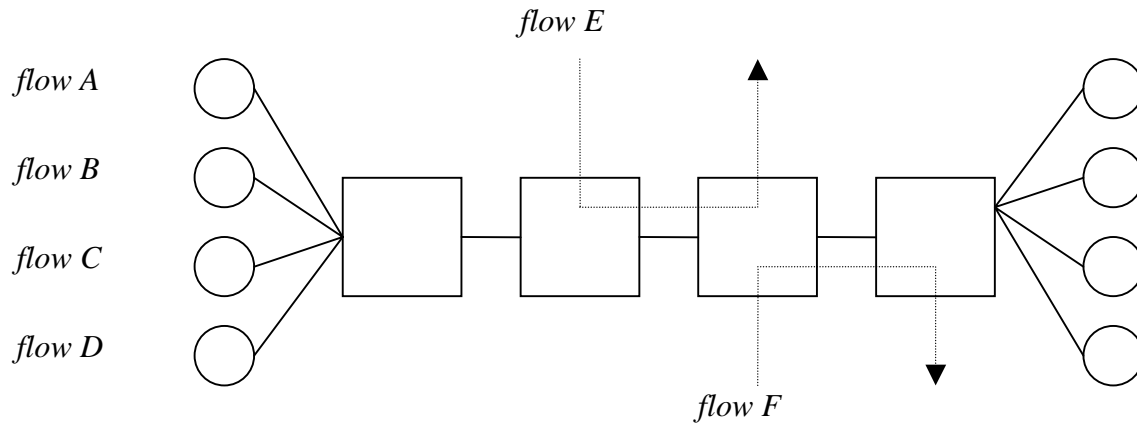
$$\mathit{tstamp} = \mathit{current_time}$$

$$\mathit{maxrnh} = \mathit{maxrnh} - 1$$

- packet queue in each node is organized as a linked list in increasing order of the variable:

$$\mathit{max_leave}_Q = \frac{\mathit{max\ rd}}{\mathit{maxrnh}} + \mathit{tstamp}$$

Simulation results



- four flows: A, B, C and D
- bitrates (Mbps): 0.3, 0.2, 0.2 and 0.1
- delay requirement (ms): 40, 60, 100 and none respectively
- cross-traffic flows E,F : ON-OFF type with 0.6 Mbps in ON
- all the links have 1 Mb/s transmission capacity and 2 ms propagation delay

Comparison between EDF and CS-EDF

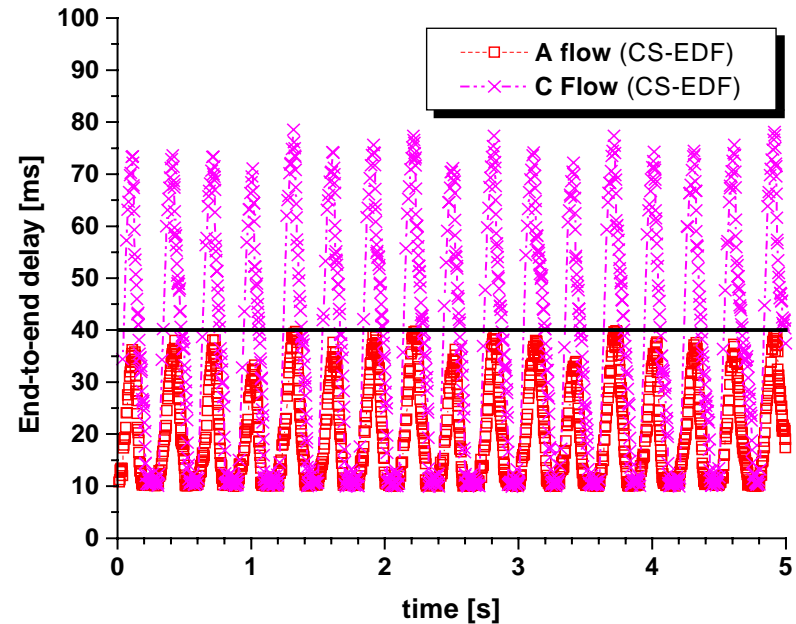
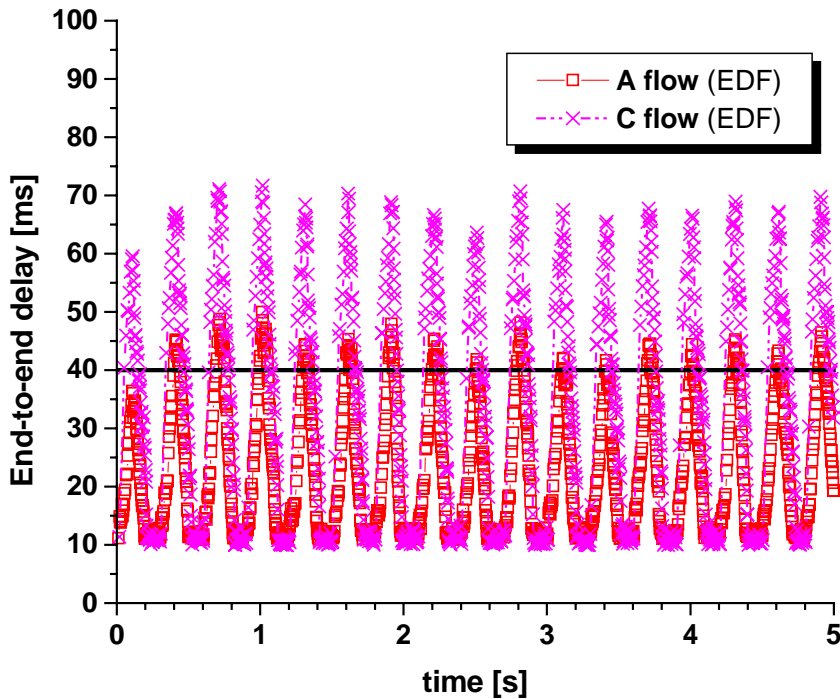


Fig.1 EDF end-to-end delay for A and C flows **Fig.2.** CS-EDF end-to-end delay for A and C flows

- once the local delay bounds are violated, there is no mechanism in the EDF approach to pursue purposely the recovery of the excess delay
- in CS-EDF, once a local delay violation occurs, there is a systematic attempt to compensate it along the remaining path

Comparison between EDF and CS-EDF

(2)

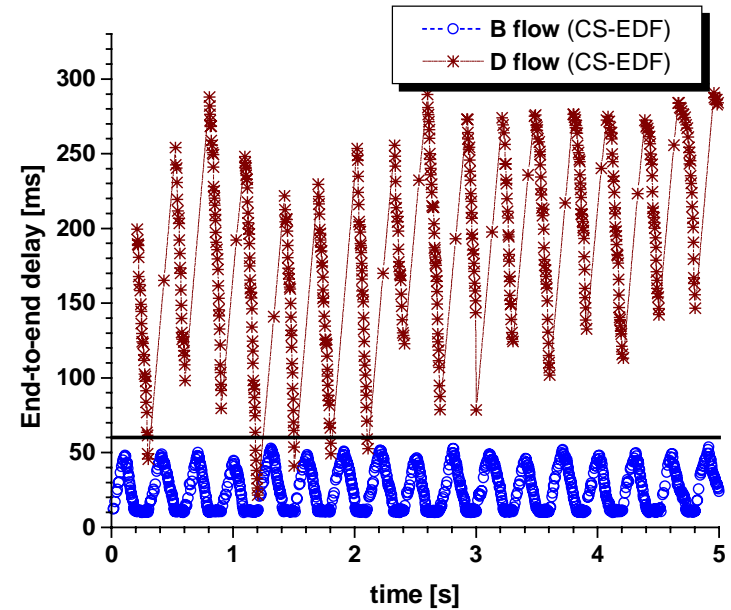
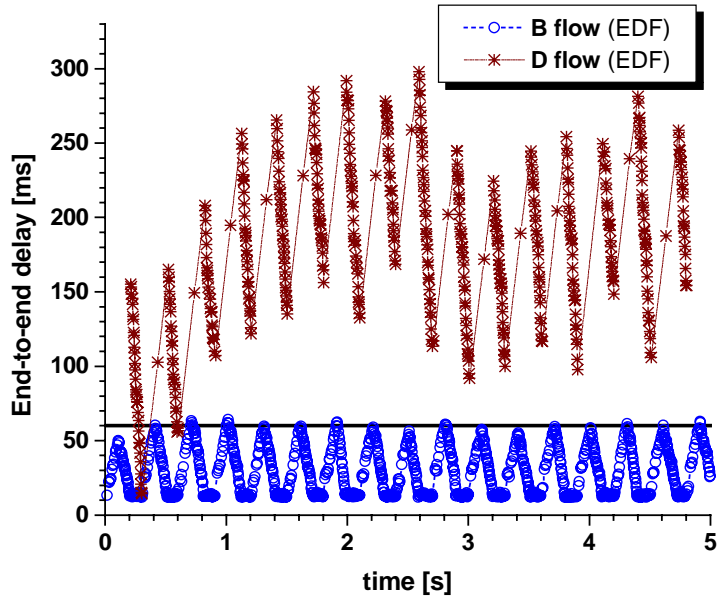


Fig.3 EDF end-to-end delay for B and D flows **Fig.2.** CS-EDF end-to-end delay for B and D flows

Conclusions & Further Work

- **FAIR-BAC**, a core-stateless architecture aiming both at delivering guaranteed services and at the fair allocation of the excess bandwidth was presented
- **CS-EDF** packet scheduling mechanism attempts to ensure the end-to-end delay requirement
- in **CS-EDF** once a local delay violation occurs, there is a systematic attempt to compensate it along the remaining path. Similarly, if the packet is ahead with respect to its local delay bound, it receives a larger delay requirement on the downstream path, allowing packets with more stringent delay requirements to be forwarded in front of it.
- simulations using more complex/realistic traffic characteristics and more complex congestion patterns are required for a better analysis
- mechanism to ensure that the resources for the traffic with reservations are not over-committed