Latency Guarantee with Stateless Fair Queuing

draft-joung-detnet-stateless-fair-queuing-00

Jinoo Joung, Jeong-dong Ryoo, Tae-sik Cheung, Yizhou Li, Peng Liu

IETF 117, July 26

Content

- Overview of the draft
- C-SCORE Framework & Operational procedure
- The E2E latency bound of C-SCORE; a closer look
- Implementation considerations
  - Stateful Entrance node
  - Time difference compensation
- Assessment for Requirements

Overview of the Draft

4. Fair Queuing Schedulers
5. Assumptions
6. Work Conserving Stateless Core Fair Queuing (C-SCORE)
   - 6.1. Framework
   - 6.2. E2E Latency Bound
   - 6.3. Operational Procedure
     - 6.3.1. Metadata
     - 6.3.2. Network Configuration
     - 6.3.3. Operational Procedure in Entrance Node
     - 6.3.4. Operational Procedure in Code Node
     - 6.3.5. Considerations for Entrance Node
     - 6.3.6. Time Difference Between Nodes

Renewed, detailed descriptions

Covered in ADN Framework document

Covered in Data-plane open meeting, April.

New items

Work Conserving Stateless Core Fair queuing (C-SCORE)

- Framework
  - FT, Finish time F(p) = Service order of packet p. Smaller FT gets earlier service.
  - At entrance node 0: $F_0(p) = \max\{F_0(p-1), A_0(p)\} + L(p)/r$;
  - At core node h: $F_h(p) = F_{h-1}(p) + d_{h-1}(p)$.
  - Whenever there are packets in the queue, the link never idles.
  - Packets in the queue are served in the ascending order of FT
- If $d_h(p) = Lmax_h/R_h + L/r$,
- Then the E2E latency of p's flow is bounded [Kaur] by

$$\frac{B-L}{r} + \sum_{h=0}^{H}\left(\frac{Lmax_h}{R_h} + L/r\right)$$

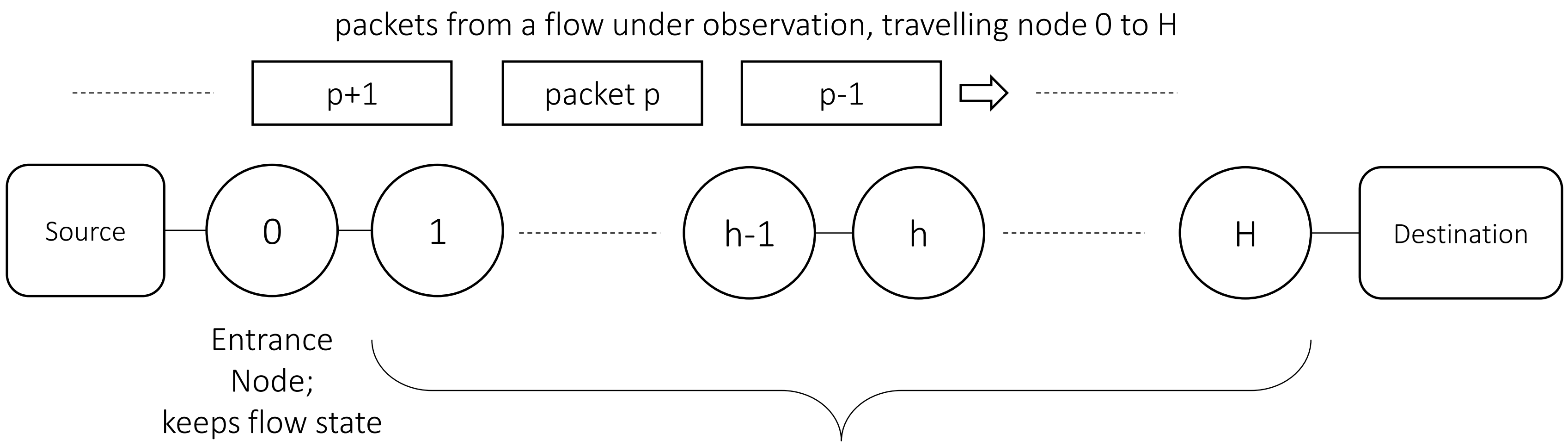
This bound is same with a stateful fair queuing network (PGPS, etc.)

$\frac{Lmax_h}{R_h}$ is the only term external & can be negligible.

- B, L, r are flow specific, which can be controlled according to requirement → Latency bound can be adjusted if necessary

Symbol	Definition
$F_h(p)$	'Finish time' of packet p at node h
$A_0(p)$	Arrival time of p at node 0
L(p)	Length of p
L	Max Packet Length of p's flow
ρ_j	Arrival rate of flow j
Bj	Max burst of flow j
r	Service rate of p's flow
$r_{h,j}$	Service rate of flow j at node h
$Lmax_h$	Max Packet Length at node h
R_h	Link capacity of h
f(h)	Set of flows in node h

C-SCORE Framework Overview



Entrance
Node;
keeps flow state
$F_0(p-1)$, L, r;
calculates $F_0(p)$ & $F_1(p)$;
marks $F_1(p)$, L, r as packet metadata;
serves packets in ascending order of $F_0(p)$.

Core Nodes; stateless.
Node h Keeps $Lmax_h$ & R_h; calculates $F_{h+1}(p)$;
updates metadata $F_h(p)$ to $F_{h+1}(p)$;
serves packets in ascending order of $F_h(p)$.

Symbol	Definition
$F_h(p)$	'Finish time' of packet p at node h
$A_0(p)$	Arrival time of p at node 0
L(p)	Length of p
L	Max Packet Length of p's flow
ρ_j	Arrival rate of flow j
Bj	Max burst of flow j
r	Service rate of p's flow
$r_{h,j}$	Service rate of flow j at node h
$Lmax_h$	Max Packet Length at node h
R_h	Link capacity of h
f(h)	Set of flows in node h

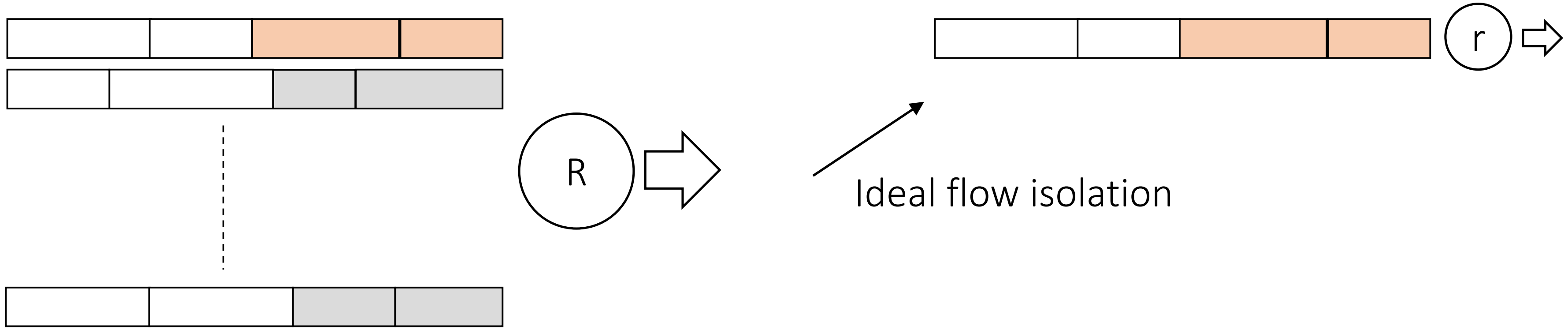
C-SCORE Operational procedures

1. Network configuration stage
   - A source requests latency bound for flow i, with specifying its ρ_i and Bi
   - If the latency bound can be met, admit the flow
   - Network reserves the links in the path such that
     - $\rho_j \le r_{h,j}$ and $\sum_{j \in f(h)} r_{h,j} \le R_h$, for all h
2. The entrance node or the source
   - Maintains the flow state, i.e. F_0(p-1) & r.
   - Maintains a clock, for A_0(p).
   - Maintains the link info $Lmax_0/R_0$.
   - Upon receiving or generating packet p,
     - Obtains $F_0(p) = \max\{F_0(p-1), A_0(p)\} + L(p)/r$. Use it as the FT in 0. Put p in a sorted queue.
     - Obtains $F_1(p) = F_0(p) + Lmax_0/R_0 + L/r$.
     - Records F_1(p) & L/r in the packet as metadata for the use in the next node 1.
   - Update the flow state to F_0(p).
3. A core node h
   - Maintains the link info $Lmax_h/R_h$. (A rather static value)
   - Upon receiving packet p,
     - retrieve meta-data F_h(p) & L/r, use F_h(p) as the FT. Put p in a sorted queue.
     - Obtain $F_{h+1}(p) = F_h(p) + Lmax_h/R_h + L/r$.
     - Update metadata F_h(p) with F_{h+1}(p) before or during p is in the queue.

Symbol	Definition
F_h(p)	‘Finish time’ of packet p at node h
A_0(p)	Arrival time of p at node 0
L(p)	Length of p
L	Max Packet Length of p’s flow
ρ_j	Arrival rate of flow j
Bj	Max burst of flow j
r	Service rate of p’s flow
$r_{h,j}$	Service rate of flow j at node h
$Lmax_h$	Max Packet Length at node h
R_h	Link capacity of h
f(h)	Set of flows in node h

Closer look at E2E latency bound of C-SCORE

- An ideal flow isolation is achieved by a scheduler, which serves the flow as if there is no other flow in an imaginary link whose capacity is equal to the allocated service rate, r, to the flow.



- In this case the latency upper bound D is a function of the flow's parameters only.
- D <= (B-L)/r + L/r.   If B=L then D <= L/r.

This is the transmission delay.

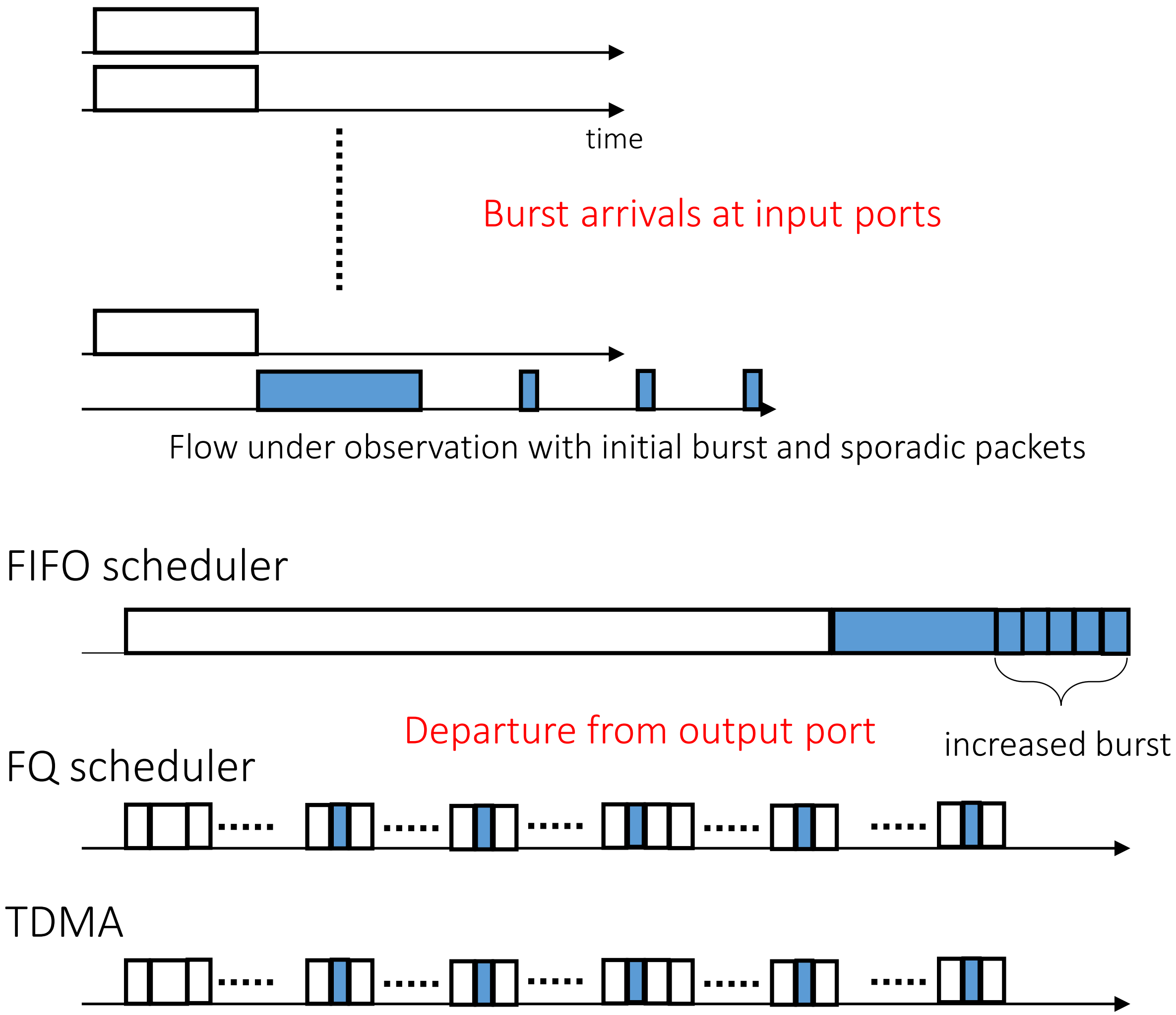
- For a network with the ideal flow isolation schedulers: D <= (B-L)/r + H*L/r, where H is the # of hops.

"Pay burst only once"

Closer look at E2E latency bound of C-SCORE

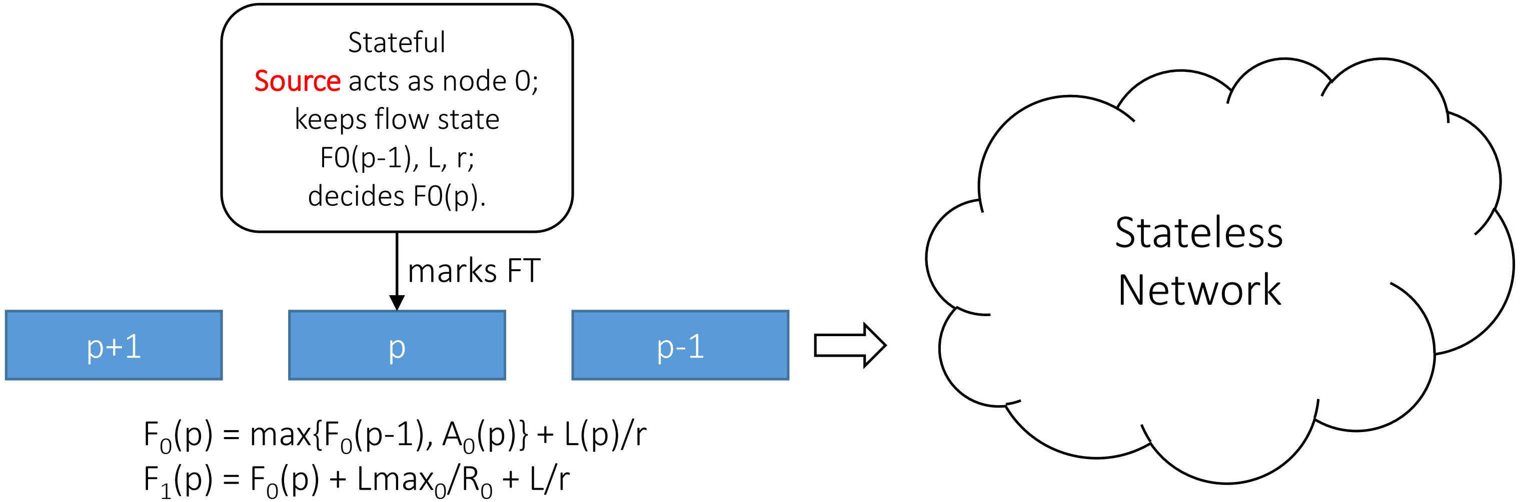
- FIFO accumulates bursts, and flows are not isolated at all.
- TDMA isolates flows perfectly, but loses efficiency and robustness.
- Fair Queuing isolates flows almost perfectly, with efficiency, robustness & statistical multiplexing gain.
- $D_{ideal} \le (B-L)/r + H*L/r$
- $D_{FQ} = D_{C\text{-}SCORE} \le (B-L)/r + H*(L/r + Lmax/R)$

This term is due to the non-preemptive nature of the FQ scheduler.



Considerations of Stateful entrance node

- Flow states still have to be maintained in entrance nodes.
- The notion of an entrance node, however, can be mitigated into various edge devices, including a source itself.

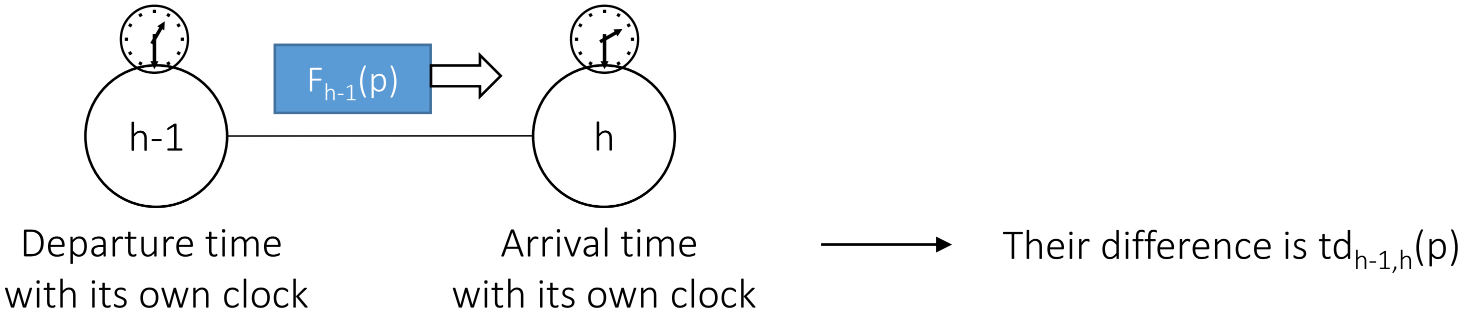


$F_0(p) = \max\{F_0(p\text{-}1), A_0(p)\} + L(p)/r$
$F_1(p) = F_0(p) + Lmax_0/R_0 + L/r$

- FT of a packet is decided based on the maximum of $F_0(p\text{-}1)$ and $A_0(p)$; and $L(p)/r$. These parameters are flow specific.
  - There is no need to know any other external parameters.
  - The arrival time of p to the network, $A_0(p)$, can be approximated by the generation time of p at the source.
- Then $F_0(p)$ is determined at the packet generation time and can be recorded in the packet.
- Therefore, we can simplify the proposed solution to a great degree, and can apply to any network with robustness and scalability.

Considerations of Time difference between nodes

- In reality, there are time differences between nodes, including the differences due to the propagation delays.



- Note that FT does not need to be precise. It is used just to indicate the packet service order. Therefore, we can assume that the propagation delay is constant and the clocks do not drift.
- $td_{h-1,h}(p)$ can be simplified to a constant value, $td_{h-1,h}$.
- In this case the delay factor should be modified to be

$$d_h(p) = \frac{Lmax_h}{R_h} + L/r + td_{h,h+1}.$$

- The E2E latency bound increases as much as the sum of propagation delays from node 0 to h.
- C-SCORE does not need global time synchronization.

Requirements check: $D_{FQ} <= (B-L)/r + H*(L/r + Lmax/R)$

			Remark
3.1	Tolerate Time Asynchrony	Yes	Synch is not necessary.
3.2	Support Large Single-hop Propagation Latency	Yes	Independent of propagation delay
3.3	Accommodate the Higher Link Speed	Partial	Priority queue can be supported up to 600Gbps Ethernet with 2.5GHz clock ASIC (See Note next page). The throughput is independent of the queue length.
3.4	Be Scalable to The Large Number of Flows and Tolerate High Utilization	Yes	Independent of # of flows or Utilization level
3.5	Tolerate Failures of Links or Nodes and Topology Changes	Yes	Requires re-admission control & resource reservation (like all the other candidates)
3.6	Prevent Flow Fluctuation - Tolerate Dynamic Flows Join/Leave - Burst accumulation	Yes	- Requires admission control & resource reservation (like all the other candidates) - Prevents burst accumulation
3.7	Be Scalable to a Large Number of Hops with Complex Topology	Yes	Independent of topology, but the E2E latency bound is linear function of hop counts
3.8	Support Multi-Mechanisms in Single Domain and Multi-Domains	Not applicable	It copes well with other asynchronous solutions, such as TSN ATS, deadline-based forwarding, etc.
4.1	Support Aggregated Flow Identification	Not applicable	Flow aggregation is not necessary.
4.2	Support Information used by Functions Ensuring Deterministic Latency		Metadata support is necessary.

Note

- [Bhagwan00] showed that, with a pipelined heap, a priority queue is supported up to 15Gbps, 2^{32} priority levels, for 53 Byte ATM cells, with 0.35 micro technology, ~ 100MHz clock.
- This is equivalent to one {enqueue & dequeue} operation per TWO clocks.
  - For 250MHz clock, (2 clk / 250M clk per sec) = 8ns is required to enqueue & dequeue
  - For 2.5GHz clock (ASIC), it is 0.8ns.
- For a 64byte (minimum sized) Ethernet packet,
  - 600Gbps line speed means (512bit / 600Gbps) = 0.85ns budget to process a packet
  - 60Gbps → 8.5ns
- We can support up to 60Gbps link speed with 250MHz clock
- and 600Gbps link speed with 2.5GHz ASIC

Thank you

- Please take a look at

  https://datatracker.ietf.org/doc/draft-joung-detnet-stateless-fair-queuing/

- Comments and Questions are welcome!

- [Bhagwan00] Ranjita Bhagwan and Bill Lin, "Fast and Scalable Priority Queue Architecture for High-Speed Network Switches", IEEE Infocom 2000 Conference, 26-30 March 2000
- [Sivaraman16] Anirudh Sivaraman, et. al. "Programmable Packet Scheduling at Line Rate", ACM SIGCOMM '16, August 22 - 26, 2016
- [Kaur] Jasleen Kaur, and Harrick M. Vin. "Core-stateless guaranteed rate scheduling algorithms." In Proceedings IEEE INFOCOM 2001. Conference on Computer Communications. Twentieth Annual Joint Conference of the IEEE Computer and Communications Society (Cat. No. 01CH37213), vol. 3, pp. 1484-1492. IEEE, 2001.
- [ADN] Jinoo Joung, Juhyeok Kwon, Jeong-Dong Ryoo, and Taesik Cheung. "Asynchronous Deterministic Network Based on the DiffServ Architecture." IEEE Access 10 (2022).
- [Zhang] Lixia Zhang. "Virtual clock: A new traffic control algorithm for packet switching networks." In *Proceedings of the ACM symposium on Communications architectures & protocols*, pp. 19-29. 1990.
- [Stoica] Ion Stoica and Hui Zhang. "Providing guaranteed services without per flow management." *ACM SIGCOMM Computer Communication Review* 29, no. 4 (1999): 81-94.
- [Stiliadis] Dimitrios Stiliadis and Varma Anujan. "Rate-proportional servers: A design methodology for fair queueing algorithms." IEEE/ACM Transactions on networking 6, no. 2 (1998): 164-174.