Stateless Prioritized Fair Queuing

< draft-venkitaraman-diffserv-spfq-00.txt >

Narayanan Venkitaraman and Jayanth Mysore

{venkitar, jayanth}@rsch.comm.mot.com

48th IETF, Pittsburgh, PA

Dynamic Packet State(DPS)

- A framework for providing end to end services without maintaining per-flow state in the core router
- Edge routers add/initialize state in the packet header
- Core routers utilize/update the state

In tandem the core and edge routers provide per-flow services

SPFQ:*Stateless Prioritized Fair queuing*

- Provides fair allocation to flows sharing the network using a core stateless architecture

  Maintaining per flow state is infeasible in high speed core routers.

- Provide intrinsic network support for intra flow priorities

  Most streams do have intra-flow priorities - e.g. MPEG, RLM. Priority aware dropping may improve perceived QoS.

SPFQ Algorithm Overview

- edge routers
  - mark packets using labels that depend on the rate at which a flow is sending packets
- all routers
  - compute a threshold(T) based on link state(level of congestion)
  - Forward/Drop based on packet label and the threshold

Marking at the edge

Sequential Marking (other variations are possible)

label(pkt)

$served+ = pkt- > size$

$pkt- > label = served/K_p$

served is reset to 0 at the end of every fixed size epoch(say, 1 sec)

Result: For a flow sending at say, 10 pkts per second, this algorithm labels packets sequentially from 1 to 10.

Forwarding Decision

Assume that we have computed the threshold, T.

enque(pkt)

if ($pkt- > label \leq T$)

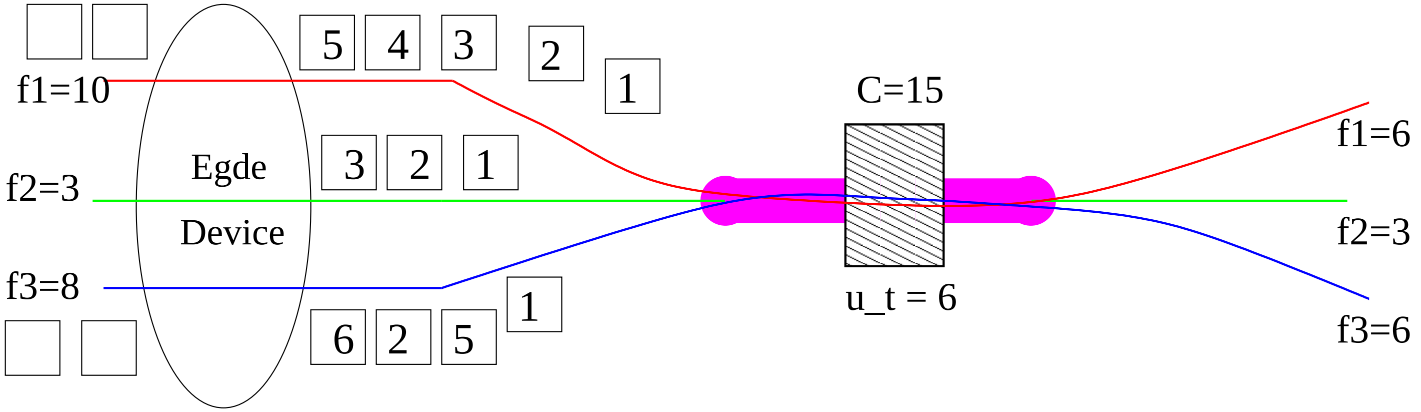
Accept(pkt)

else

Drop(pkt)

Example

Consider a link with capacity 15 units shared by 3 flows.
Assume that the computed threshold, T is 6.



Example Services

- Weighted fair allocation of bandwidth
- Minimum bandwidth assurance
- Minimizing back to back drops
- Support for layered streams