Addressing BCN Issues and Questions

Davide Bergamasco (davide@cisco.com)

Frame Sampling

- Packet-arrival based sampling is simple
- However, it introduces some issues
  - Inconsistent message overhead
  - Amount of signaling dependent on packet size
  - Time-warp effect on Qdelta (approx of dq/dt)
- We are exploring a new byte-based alternative
  - P desired sampling probability
  - E[L] is the average frame length
  - Sampling interval is E[L] / P
- E.g., P = 0.01, E[L] = 1 KB
  - Sample a packet every 100 KB received

Queue Depth Units

- **Queue depths and depth variations have been shown in units of packets only for ease of presentation and discussion**
- **All simulations and control theoretical analysis have been performed assuming 64-byte pages as a unit.**

BCN Message Generation

- Detail omitted for presentation sake
- The actual decision process is:

```
frame = sample();
if (Qlen > Qeq)
    send BCN Message;
else
    if ( frame.RLT.CPID == myCPID )
      send BCN Message;
```

Congestion Detection Trigger

- **Using *Qdelta* to detect congestion instead of *Qeq* may actually increase the chances of false positives**
- **Any increase in queue size will trigger a BCN message**
- **If *Qeq* is used to detect congestion, then bursts shorter than *Qeq* will "fly under the radar"**

RL Queue Management

- **When is the RL released?**

```
RL.rate == link_rate && RL.qlen == 0
```

- **How is multicast handled?**
  - **It's not ☺**
- **When is the RL released?**
- **What happens when we run out of RLs?**
  - **Dynamic fall-back:**
    - **SA/DA/PRI ⟶ DA/PRI ⟶ PRI ⟶ Entire link**

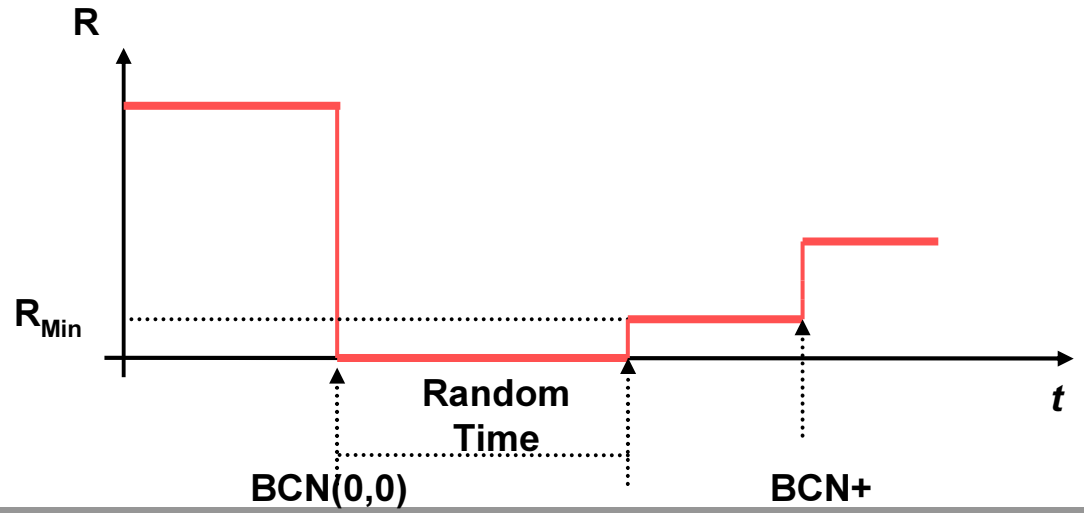
Handling Multiple Congestion Points

- **Again, detail omitted for simplicity**
- **The actual BCN processing is:**

```
bcn = receive_bcn();
Fb = calc_feedback( bcn );
if (Fb < 0)
{
  R = R * ( 1 + Gd * Fb);
  currCPID = bcn.CPID;
}
else
  if ( bcn.CPID == currCPID )
    R = R + Gi * Fb * Ru;
```

Slow Recovery From Congestion

- **BCN(0,0): Special feedback message**
- **Current rate *R* is set to 0**
- **Random timer [0,T_{Max}]: when timer expires Current rate *R* is set to R_{Min}**
- **Each time T_{Max} doubled and R_{Min} halved (mimics exponential back-off)**



Slow Recovery From Congestion

- **A *drift* has been added to the current rate to improve recovery after**
  - **BCN(0,0)**
  - **loss of BCN messages**
- **At fixed time intervals Ti (say 10 – 100 us) the current rate is incremented by a unit**

RL Tag

- **Solicit bit unclear**

  Solicit bit is gone ☺

- **Alignment issue**

  The format of the RL tag and the BCN frame is intended for purely illustrative purposes, it is not a formal proposal

- **Timestamp field**

  It is used for estimating the RTT between reaction and congestion points.

  Once the RTT is known, BCN parameters may be adjusted accordingly

Flow Identification

- **Flows need to be identified in a more explicit way**

  We reached the same conclusion ☺

BCN Message CPID

- **How is the uniqueness of the CPID achieved?**

  No need for configuration or single administrative domain

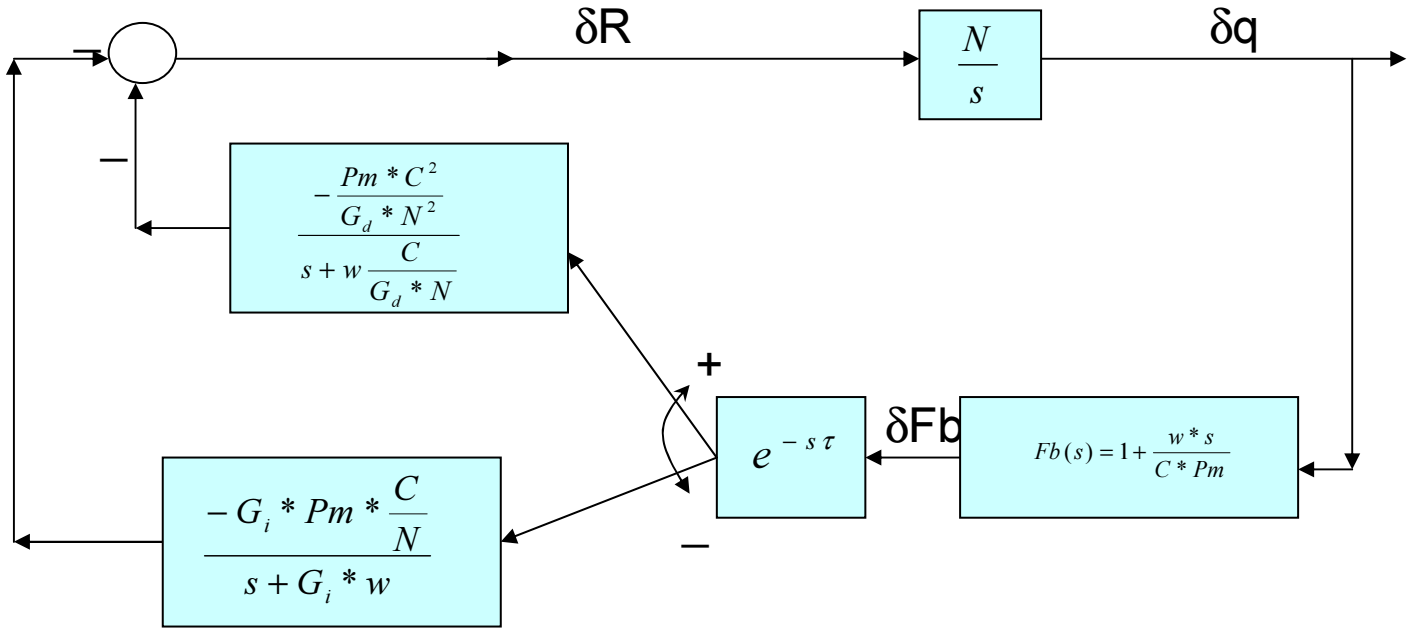
  Today's Bridges have a number of MAC addresses assigned (e.g., management entity)

  CPID = MAC Address + Port Number + Priority

Parameters Selection

- **Are the BCN parameters link speed dependent?**

  Yes, but it's not an obvious relation



Simulation Setup Clarification

- **Value of Q_eq for each simulation run?**

  The equivalent of 16 1500-bytes frames

- **Value of R_u for second part of presentation?**

  1 Mbps

- **Where is the RP located?**

  In Edge Bridges

Simulation Setup Clarification

- **What flavor of TCP is used?**

  Reno

- **What TCP parameters are assumed?**
  - **Window size = 64 KB**
  - **Other parameters are the NS-2 defaults**