• Short Range, High-Speed Datacenter-like Network
  – Link Capacity = 10 Gbps
  – Egress Port Buffer Size = 150 KB
  – Switch Latency = 1 us
  – Link Length = 100 m (.5 us propagation delay)
  – Endpoint response time = 1 us
Workload

- Traffic Type: 100% UDP (or Raw Ethernet) Traffic
- Frame Size Distribution: Fixed length (1500 bytes) frames
- Arrival Distribution: Bernoulli temporal distribution
- Offered Load/Endpoint = 49%
**BCN Parameters**

- **Qeq**
  - 16 (1500-byte frames)
  - 375 * 64 byte pages
- **Frame Sampling**
  - Frames are sampled on average 150 KB received to the egress queue
- **W = 2**
- **Gi = 12.42**
  - Computed as \((\text{Linerate}/10) * [1/(1+2\times W)\times Q_{\text{eq}}]\)  
  - \( Gi = 5.3 \times 10^{-1} \times (1500/64) = 12.42 \)
- **Gd = 6.09 \times 10^{-3}**
  - Computed as \(1/2\times[1/(1+2\times W)\times Q_{\text{eq}}]\)  
  - \( Gd = 2.6 \times 10^{-4} \times (1500/64) = 6.09 \times 10^{-3} \)
- **Ru = 1 Mbps**
BCN(0,0) and BCN(MAX)

BCN(0,0) with Drift (from Cisco)
- Current rate R is set to 0
- Random timer [0, Tmax]: when timer expires, current rate R set to Rmin
- Each time Tmax doubled and Rmin halved (exponential backoff)
- Drift: at fixed time intervals Ti, the current rate is incremented by a unit
- Settings:
  - $Q_{sc} = 112.5 \text{ KB (75\% buffer)}$
  - $T_{max} = 100\text{us}$
  - $R_{min} = 1 \text{ Gbps (10\% max rate)}$
  - Drift = 1 Mbps every 100us

• BCN(MAX):
  - Instead of BCN(0,0) when $Q > Q_{sc}$, send BCN(MAX) to decrease the rate by maximum amount ($Q_{off} = -Q_{eq}$, $Q_{delta} = 2Q_{eq}$)
BCN Detection & Global Pause

- BCN detection is enabled at CS
  - BCN
  - BCN with BCN(0,0)
  - BCN with BCN(MAX)
- Global Pause: send pause msg to each input port based on the output queue
  - CS and ES
    - Xoff thresh = 140 KB
    - Xon thresh = 130 KB
    - Pause detection is enabled
Simulation Statistics

- Fairness Statistics for each BCN scheme
  - Error: % difference from target rate for each flow = |(R_i – T)/T|
    - R_i: rate of individual flows, T = target rate (2.5 Gbps), N = 4 (number of flows)
  - Root Mean Square Fairness:
    \[ \sqrt{\frac{\sum \left( \frac{R_i - T}{T} \right)^2}{N}} \]

- Min, Mean, Max, and Standard Deviation of Fairness Index across different runs
Only BCN: CS Queue

BCN without BCN(0,0)
Only BCN: RLQ Rate

BCN without BCN(0,0)
Pause and BCN: CS Queue

BCN without BCN(0,0)
Pause and BCN: RLQ Rate

BCN without BCN(0,0)
Only BCN with BCN(0,0): CS Queue
Only BCN with BCN(0,0): RLQ Rate
Pause & BCN with BCN(0,0): CS Queue
Pause & BCN with BCN(0,0): RLQ Rate
Observation

• BCN(0,0) has problems in recovery phase:
  – Not efficient, more transient link underutilization
  – Tend to be more unfair

• Next
  – Try BCN(MAX) instead of BCN(0,0)
Only BCN with BCN(MAX): CS Queue
Only BCN with BCN(MAX): RLQ Rate
Pause & BCN with BCN(MAX): CS Queue
Pause & BCN with BCN(MAX): RLQ Rate
## Fairness Result: 20ms – 80ms

<table>
<thead>
<tr>
<th># of Runs</th>
<th>RMS Fairness Index (Pause + BCN) (Min, Mean, Max, Std)</th>
<th>RMS Fairness Index (Pause + BCN(0,0)) (Min, Mean, Max, Std)</th>
<th>RMS Fairness Index (Pause + BCN(MAX)) (Min, Mean, Max, Std)</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>(0.06, 0.21, 0.39, 0.085)</td>
<td>(0.07, 0.27, 0.53, 0.119)</td>
<td>(0.11, 0.35, 0.59, 0.136)</td>
</tr>
<tr>
<td>100</td>
<td>(0.06, 0.20, 0.39, 0.072)</td>
<td>(0.05, 0.24, 0.53, 0.104)</td>
<td>(0.05, 0.34, 0.65, 0.135)</td>
</tr>
<tr>
<td>200</td>
<td>(0.03, 0.20, 0.39, 0.070)</td>
<td>(0.03, 0.24, 0.54, 0.103)</td>
<td>(0.03, 0.33, 0.65, 0.131)</td>
</tr>
<tr>
<td>300</td>
<td>(0.03, 0.20, 0.43, 0.072)</td>
<td>(0.03, 0.24, 0.56, 0.102)</td>
<td>(0.03, 0.33, 0.65, 0.130)</td>
</tr>
</tbody>
</table>
Observation

- Original BCN is best for fairness and efficiency
- BCN(0,0) may cause more unfairness and transient inefficiency.
- BCN(MAX) causes more unfairness. (Some flow may take long time to recover, and the result maybe better with drift.)