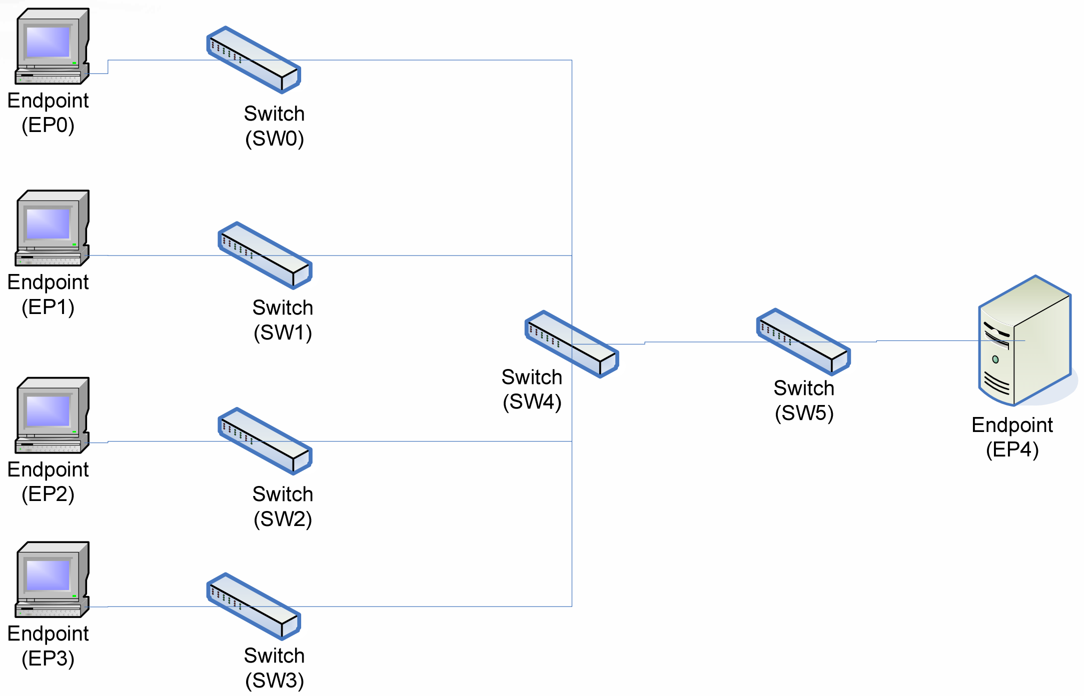
# BCN Calibration Simulation with Global Pause & Drift

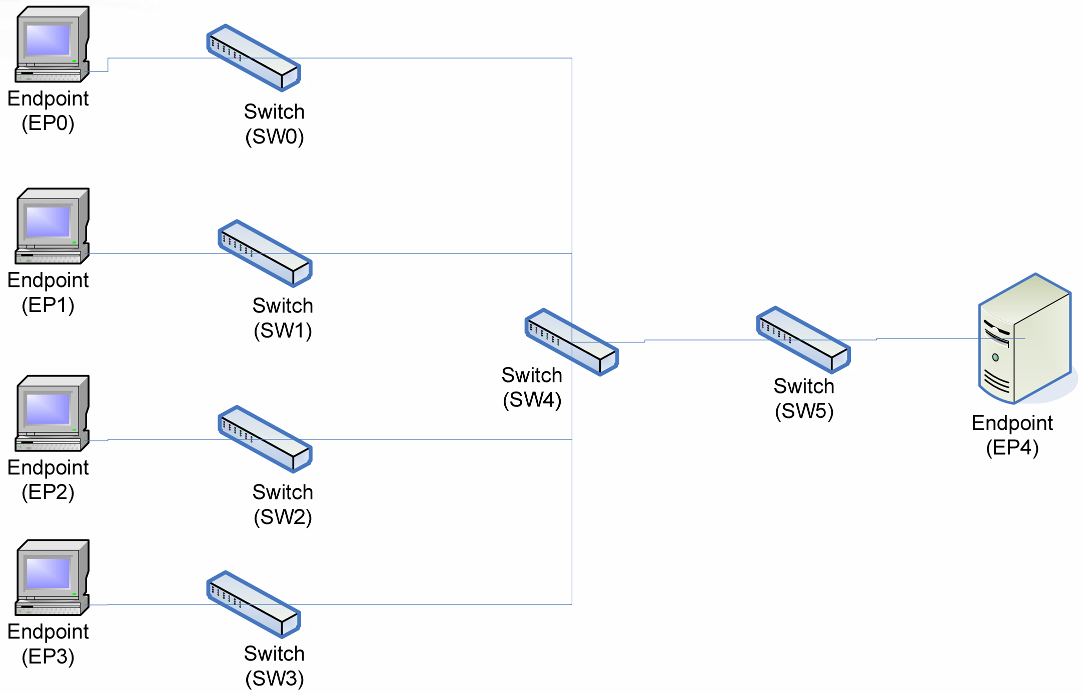
October 23, 2006

# Topology



- 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
- Destination Distribution: EP0-EP3 send to EP4
- 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  (Linerate/10) * [1/((1+2*W)*Q_eq)]
  - Gi = 5.3 x $10^{-1}$ * (1500/64) = 12.42
- Gd = 6.09 x $10^{-3}$
  - Computed as 1/2*[1/((1+2*W)*Q_eq)]
  - Gd = 2.6 x $10^{-4}$ * (1500/64) = 6.09 x $10^{-3}$
- Ru = 1 Mbps

# BCN(0,0), BCN(MAX), Drift

BCN(0,0) (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)
- Settings:
  - Qsc = 112.5 KB (75% buffer)
  - Tmax = 100us
  - Rmin = 1 Gbps (10% max rate)

- BCN(MAX):
  - Instead of BCN(0,0) when Q>Qsc, send BCN(MAX) to decrease the rate by maximum amount (Qoff = -Qeq, Qdelta = 2Qeq)
- Drift:
  - At fixed time intervals Ti, the current rate is incremented by a unit
  - Never stop drift except timeout in BCN(0,0)
  - Drift = 1 Mbps every 100us

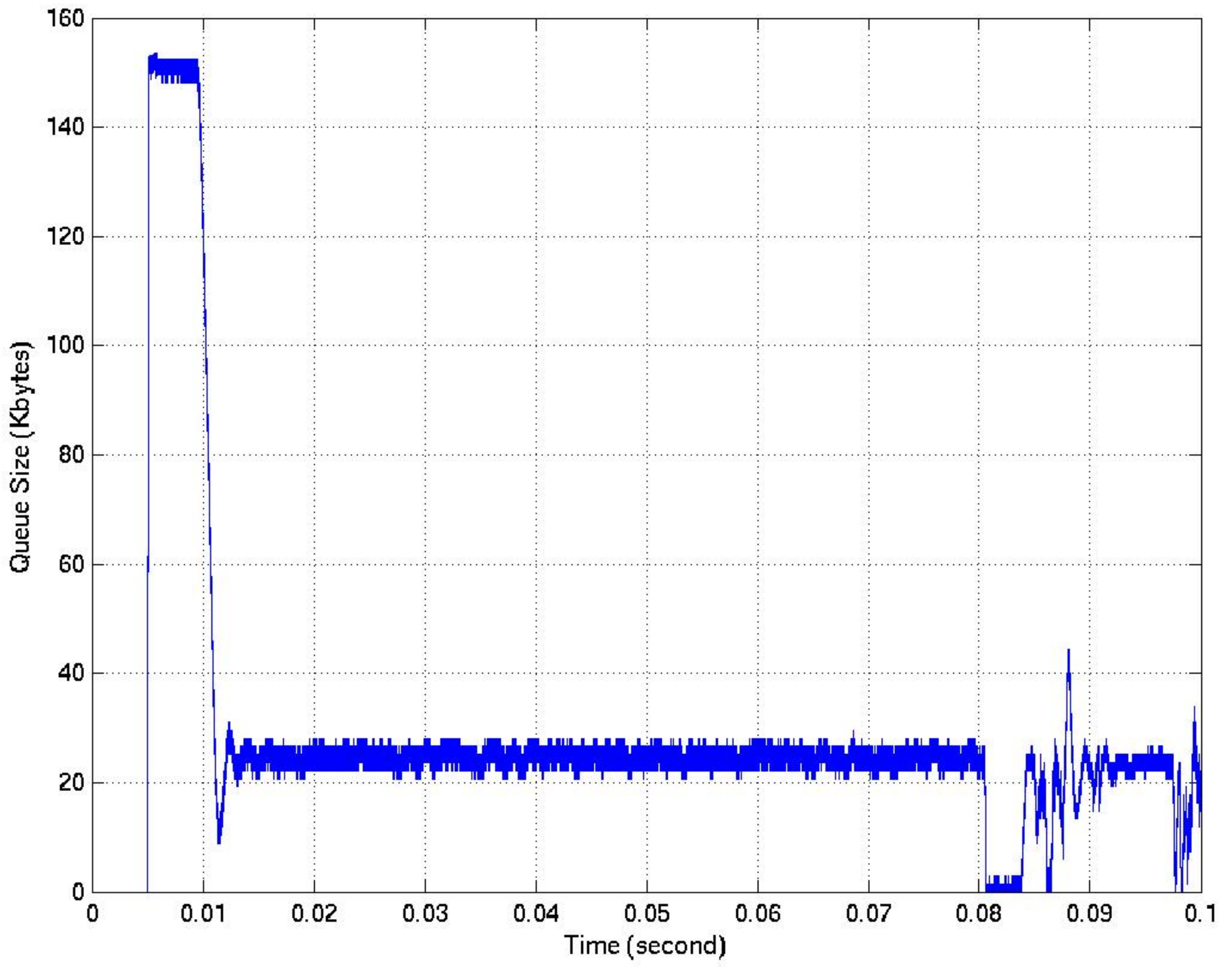
# BCN Detection & Global Pause

- 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

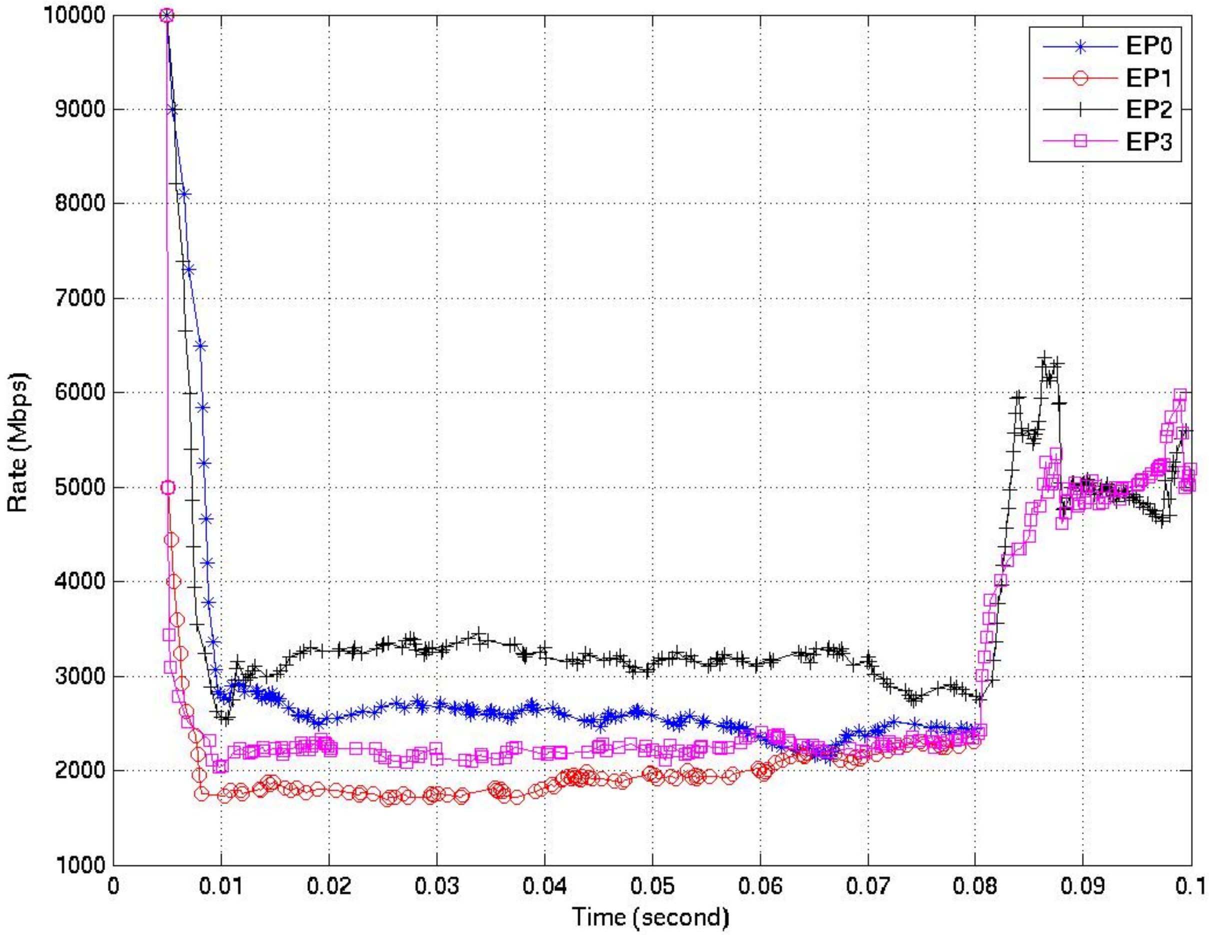
- Fairness Statistics for each BCN scheme
  - Error: % difference from target rate for each flow = |(Ri – T)/T|
    - Ri: rate of individual flows, T = target rate (2.5 Gbps), N = 4 (number of flows)
  - Root Mean Square Fairness: $\sqrt{\dfrac{\sum(\dfrac{R_i - T}{T})^2}{N}}$
- Min, Mean, Max, and Standard Deviation of Fairness Index across different runs

# Only BCN: CS Queue



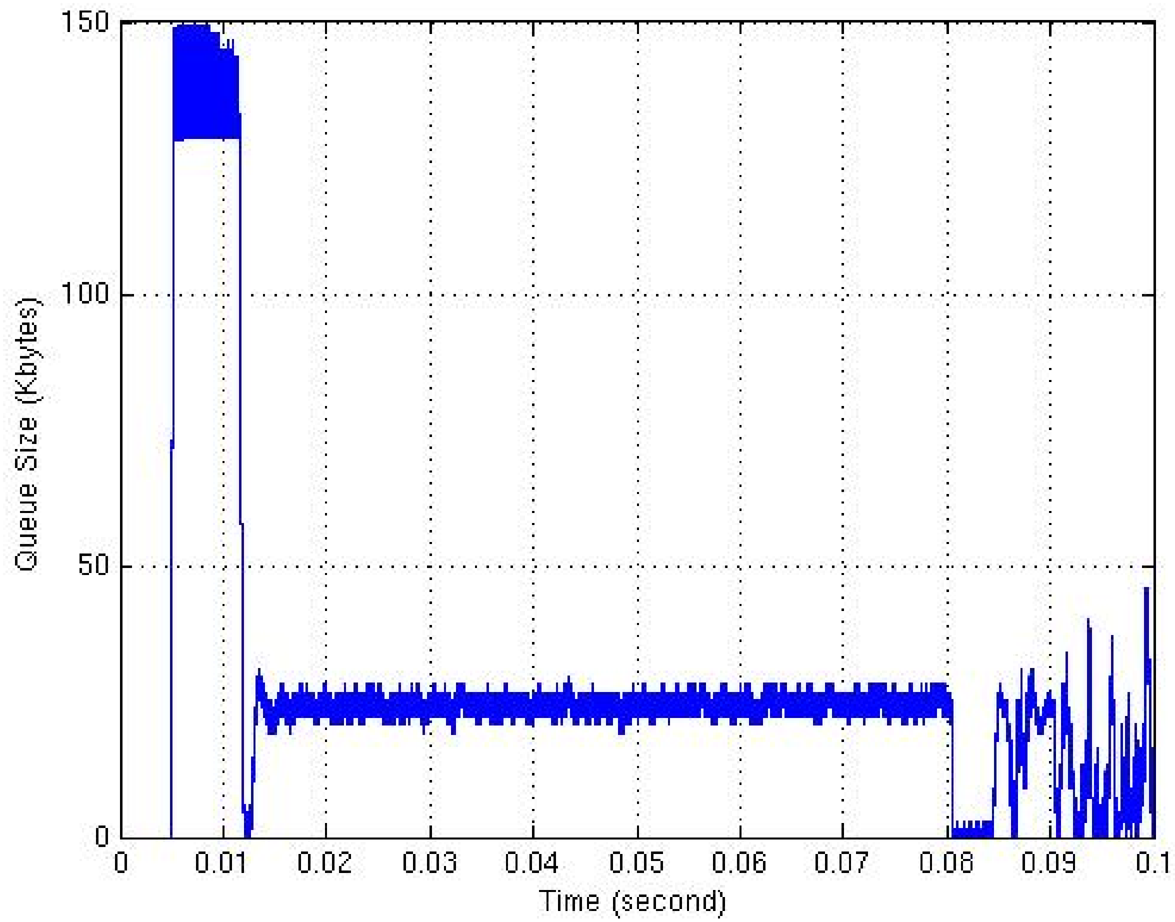
BCN without BCN(0,0)

# Only BCN: RLQ Rate



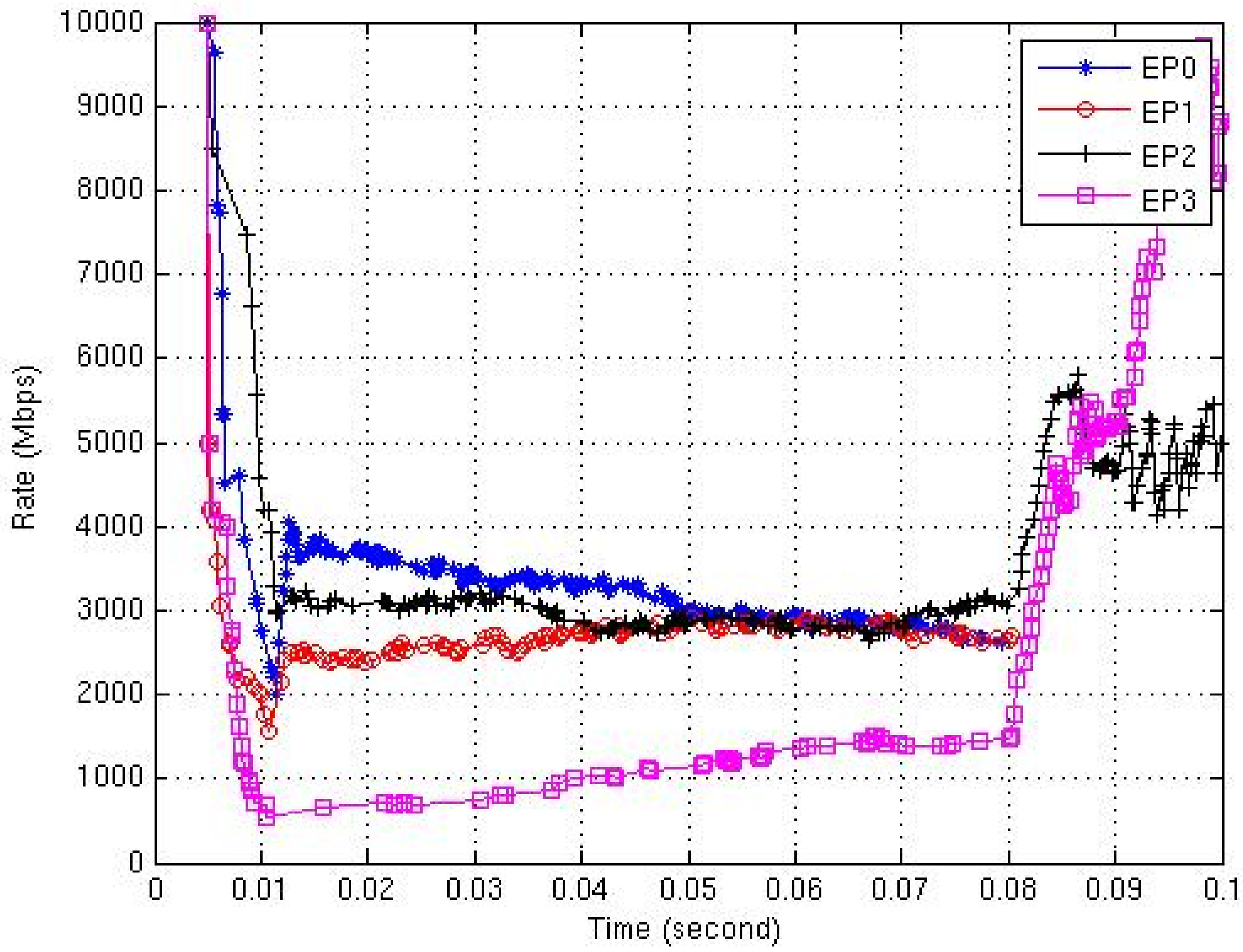
BCN without BCN(0,0)

# Pause and BCN: CS Queue



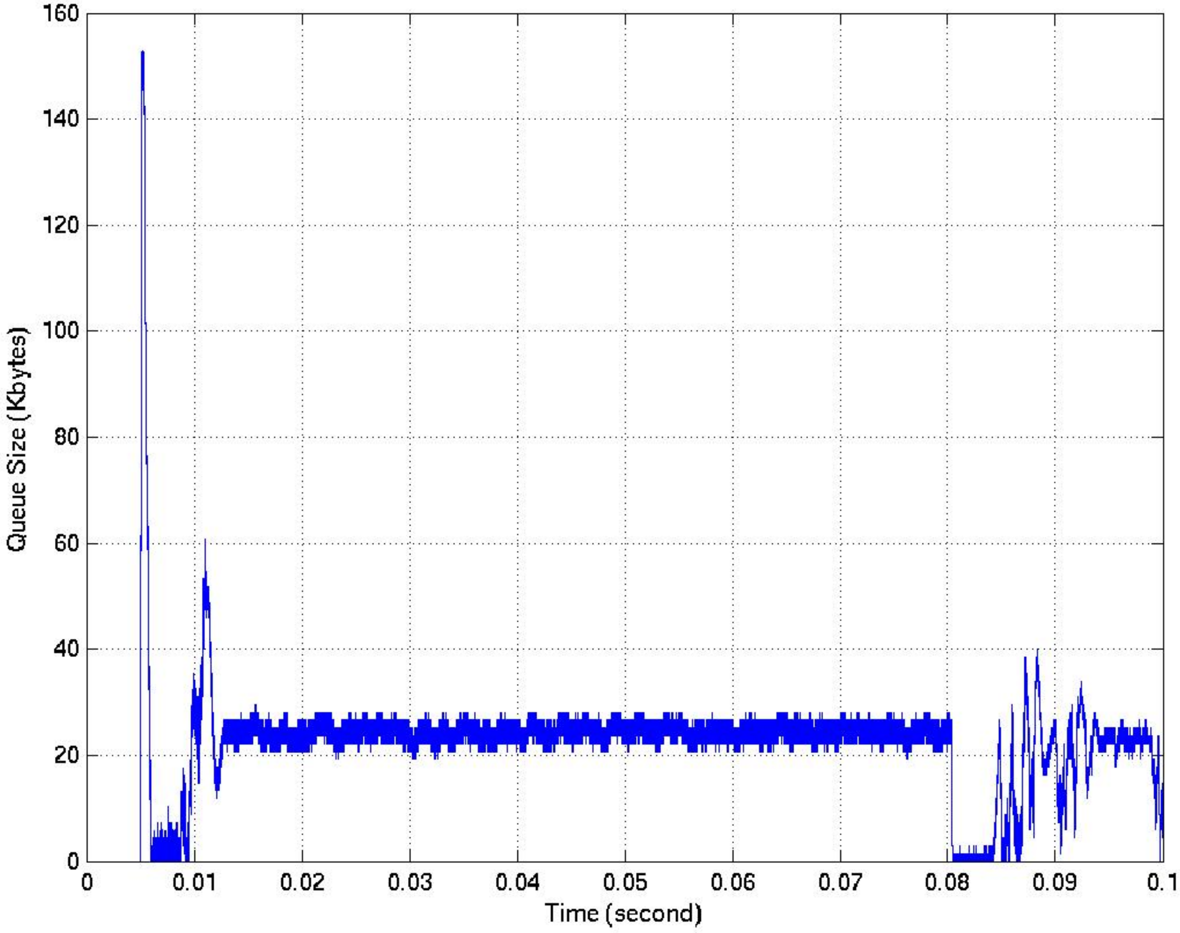
BCN without BCN(0,0)

# Pause and BCN: RLQ Rate

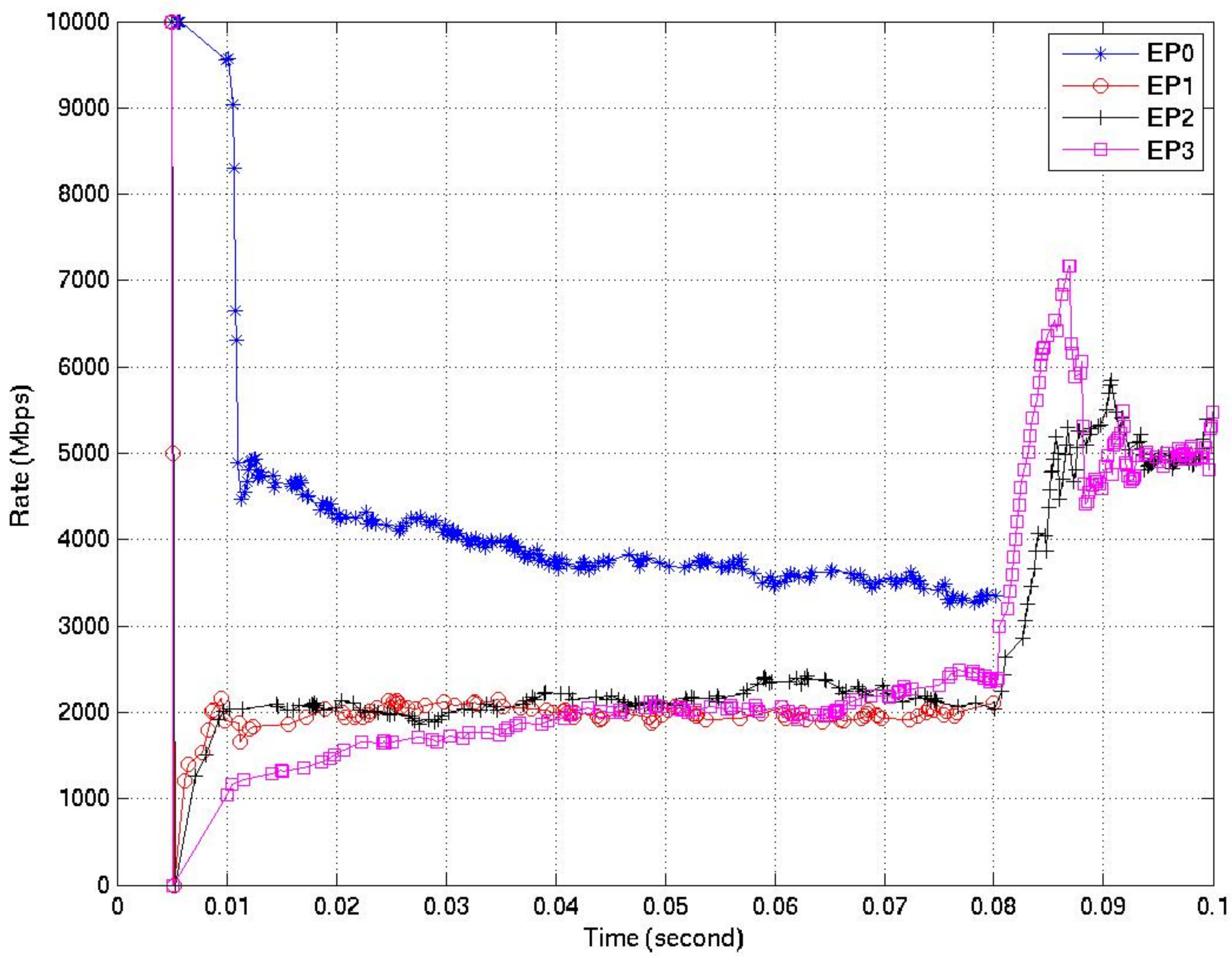


BCN without BCN(0,0)

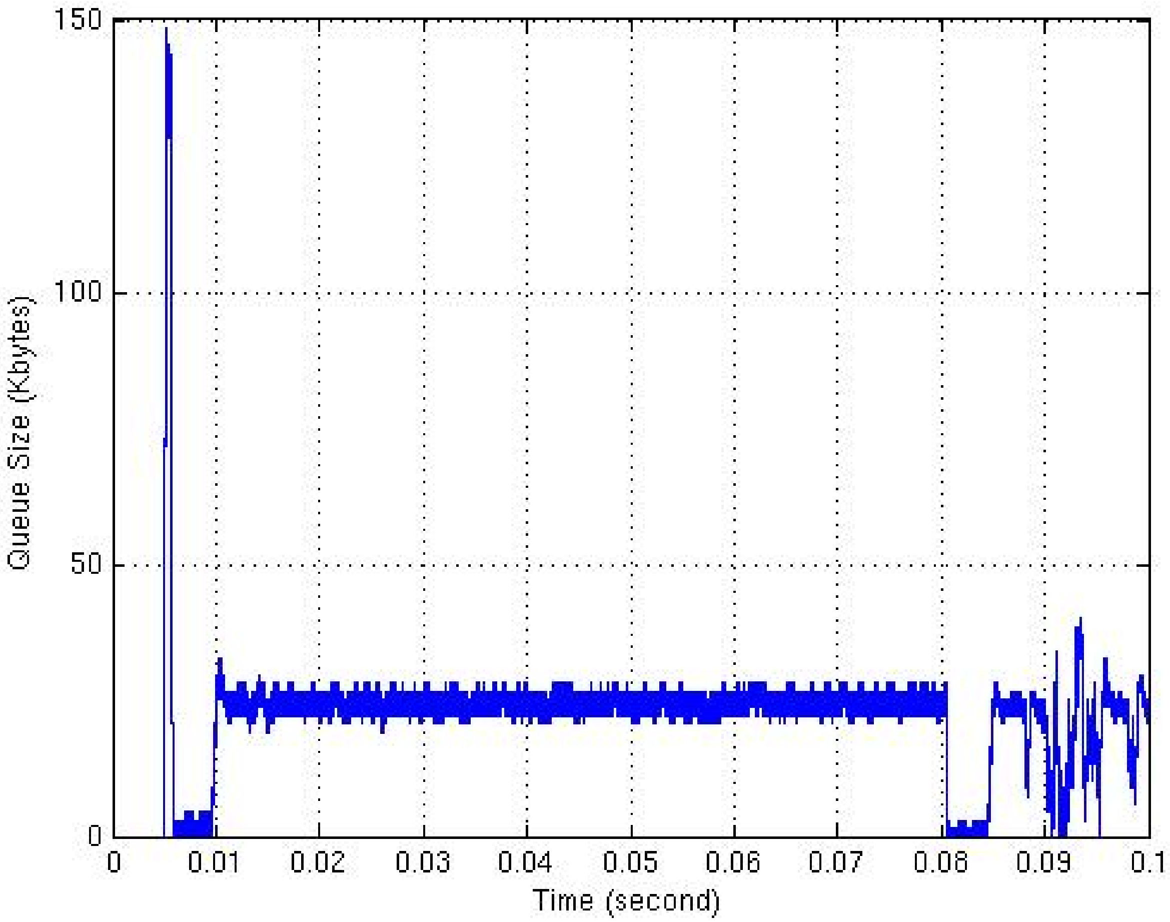
# Only BCN with BCN(0,0): CS Queue



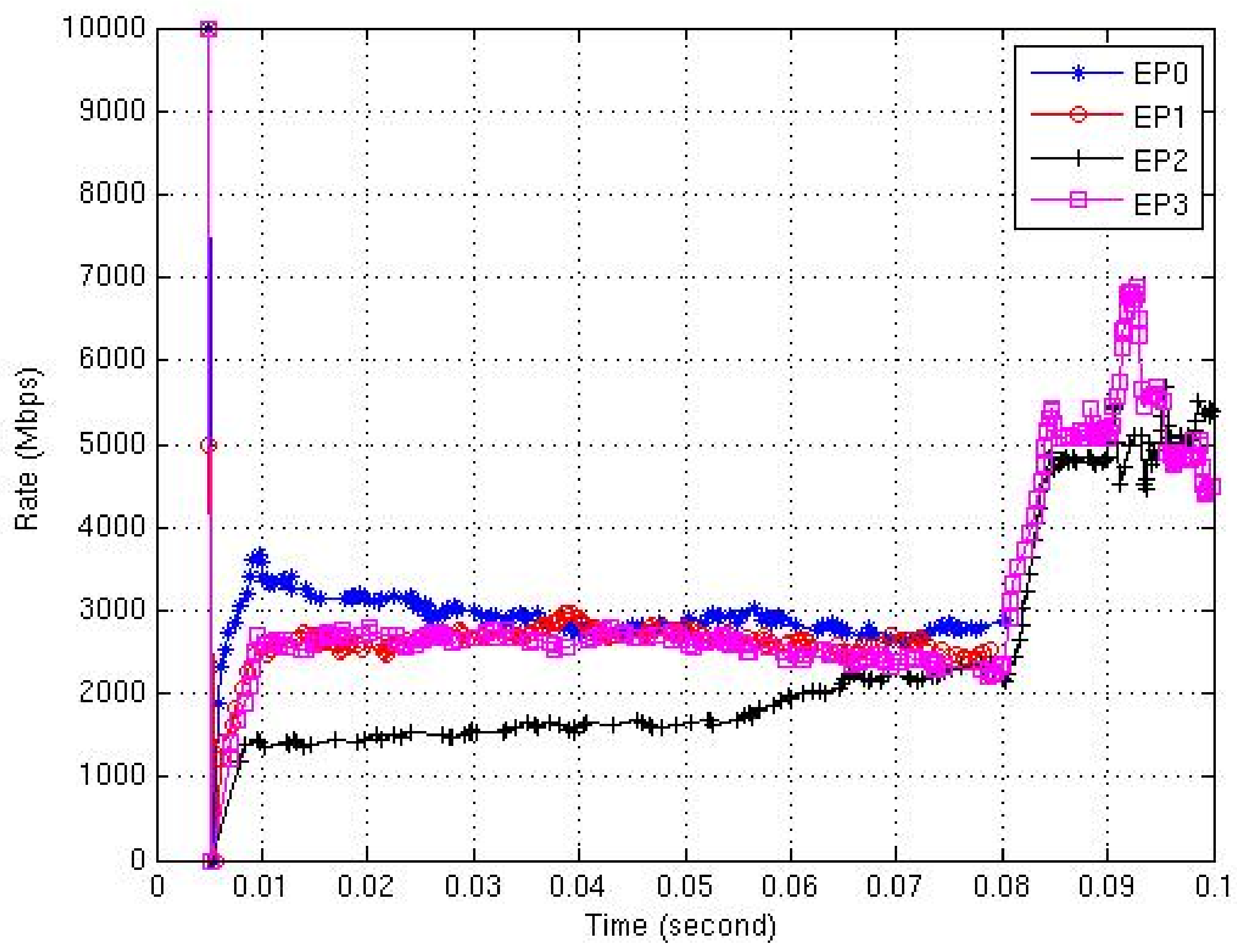
# Only BCN with BCN(0,0): RLQ Rate



# Pause & BCN with BCN(0,0): CS Queue



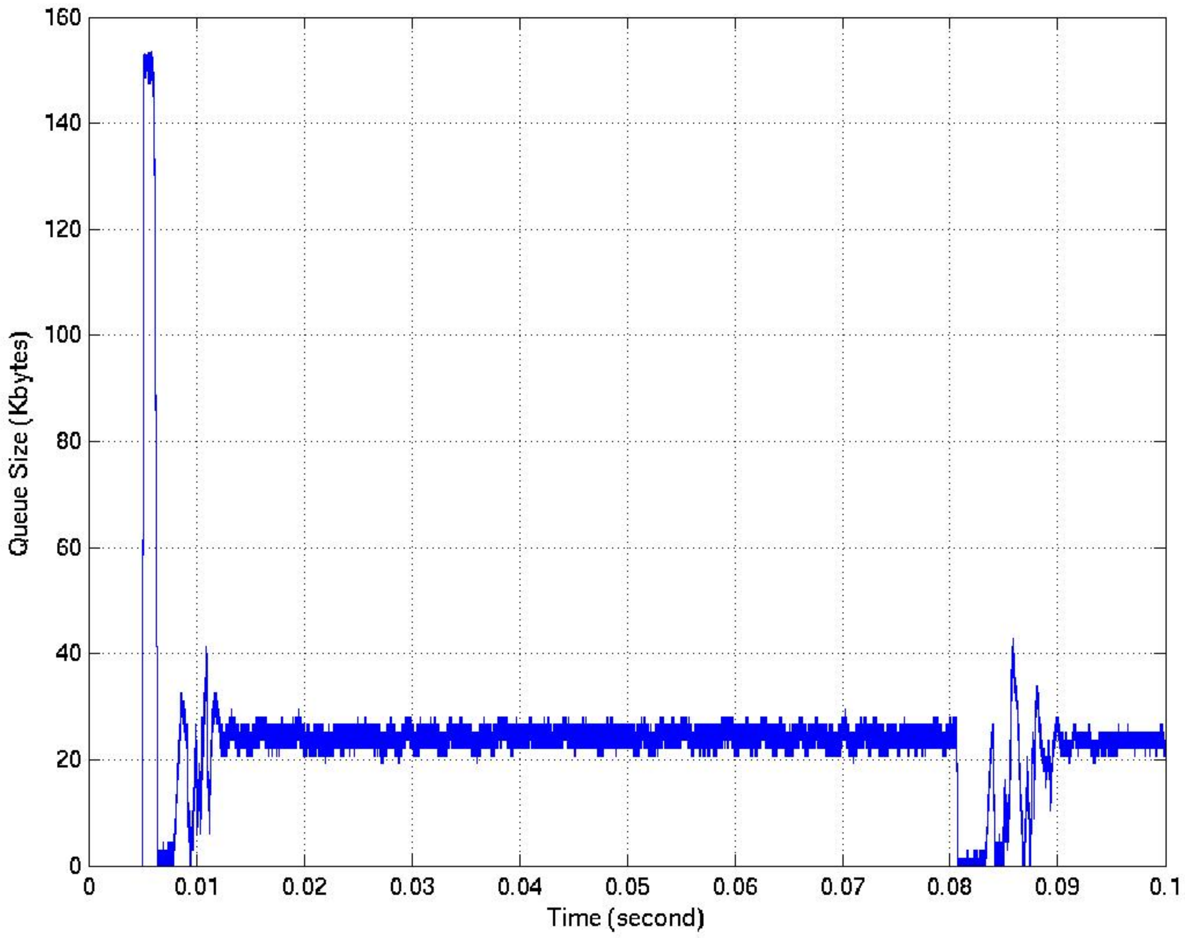
# Pause & BCN with BCN(0,0): RLQ Rate



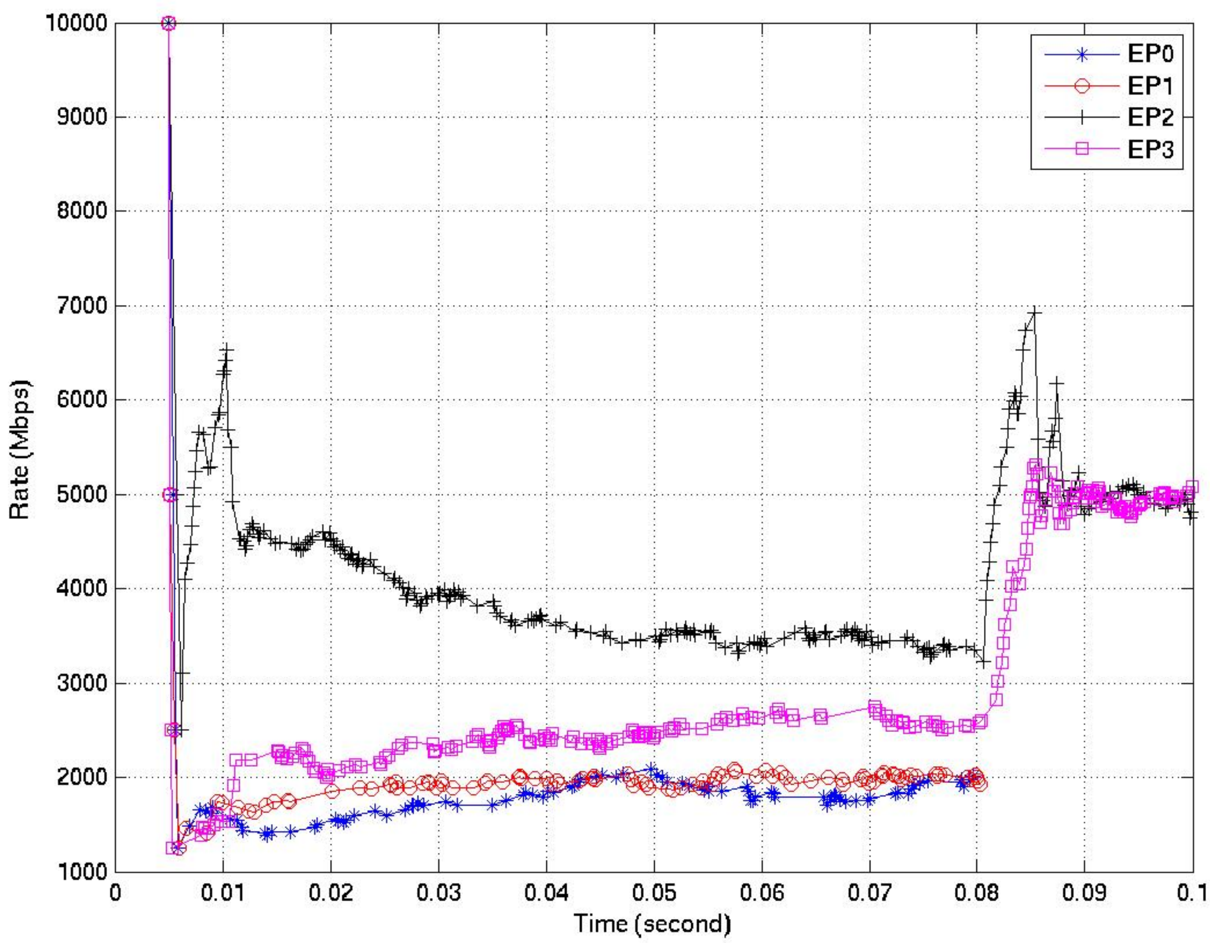
# Observation

- BCN(0,0) in recovery phase:
  - More transient link underutilization on congested links
  - Tends to be more unfair but drift helps
  - Much shorter period of drop (no Pause) or shorter Pause duration
- Next
  - Try BCN(MAX) instead of BCN(0,0)

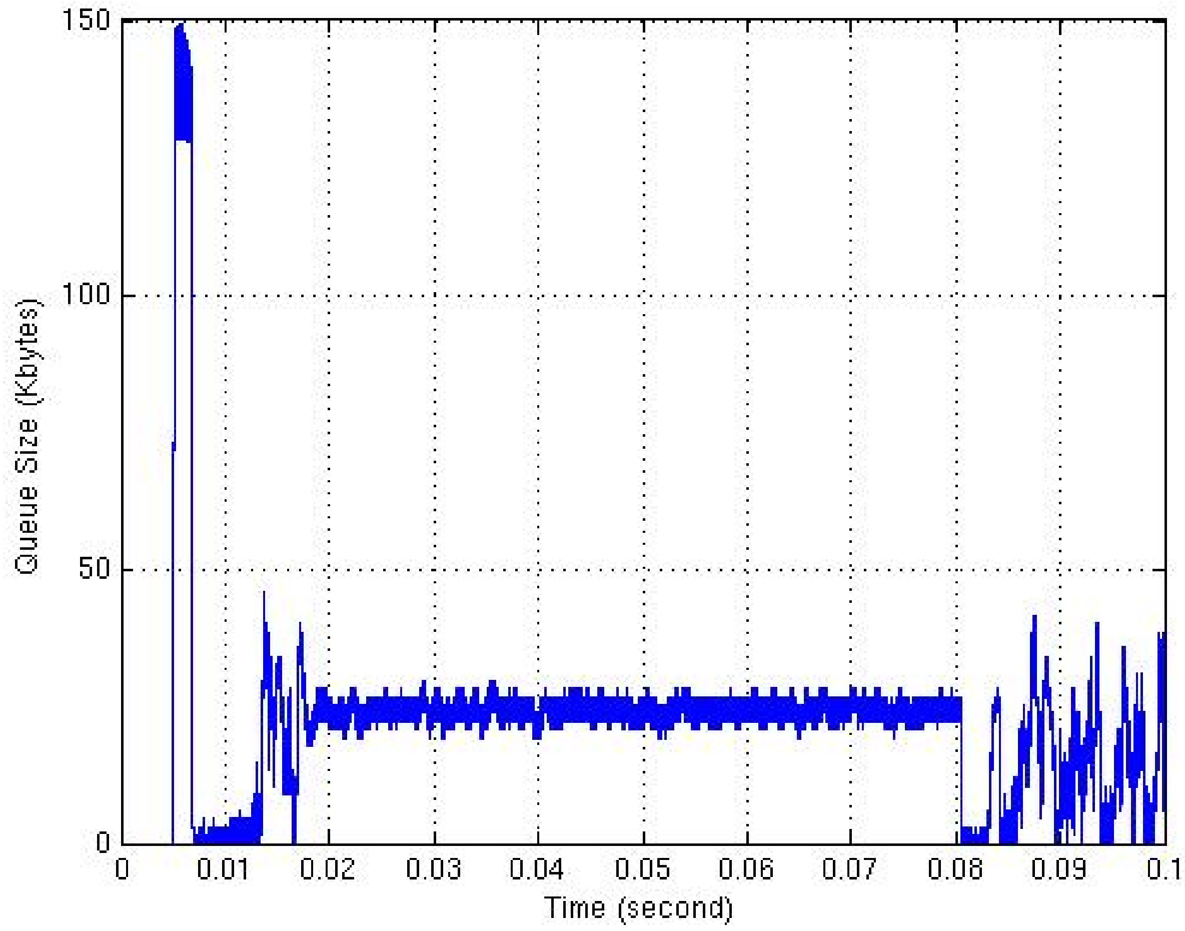
# Only BCN with BCN(MAX): CS Queue



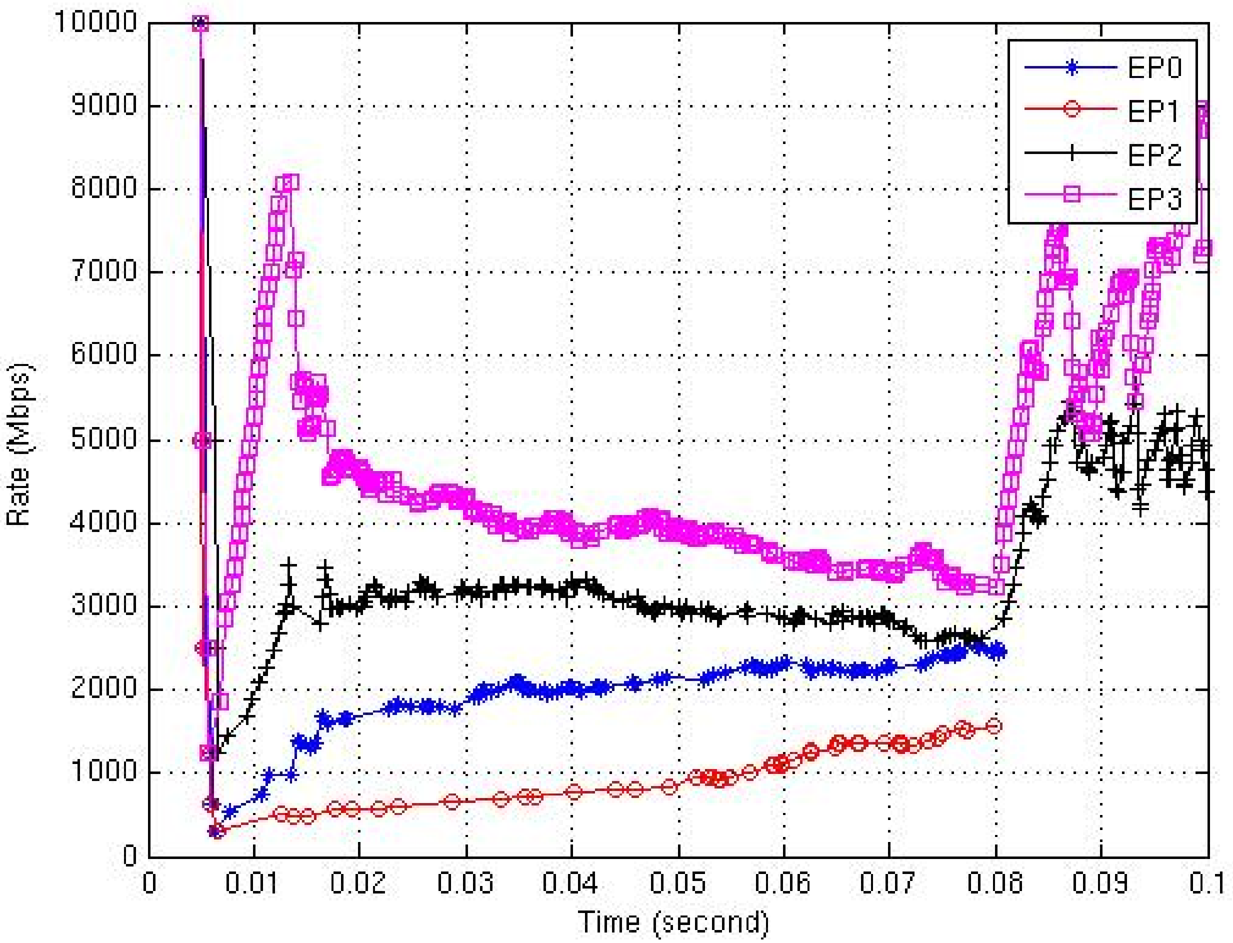
# Only BCN with BCN(MAX): RLQ Rate



# Pause & BCN with BCN(MAX): CS Queue



# Pause & BCN with BCN(MAX): RLQ Rate



# Fairness Result: 20ms – 80ms

| With Drift | | | |
|---|---|---|---|
| # of Runs | RMS Fairness Index (Min, Mean, Max, Std) (BCN) | RMS Fairness Index (Min, Mean, Max, Std) (BCN(0,0)) | RMS Fairness Index (Min, Mean, Max, Std) (BCN(MAX)) |
| 300 | (0.02, 0.11, 0.31, 0.048) | (0.03, 0.16, 0.32, 0.055) | (0.01, 0.15, 0.34, 0.066) |
| | (Pause + BCN) | (Pause + BCN(0,0)) | (Pause + BCN(MAX)) |
| 300 | (0.03, 0.14, 0.30, 0.056) | (0.02, 0.13, 0.30, 0.057) | (0.04, 0.22, 0.54, 0.078) |

| Without Drift | | | |
|---|---|---|---|
| # of Runs | RMS Fairness Index (Min, Mean, Max, Std) (BCN) | RMS Fairness Index (Min, Mean, Max, Std) (BCN(0,0)) | RMS Fairness Index (Min, Mean, Max, Std) (BCN(MAX)) |
| 300 | (0.01, 0.15, 0.33, 0.063) | (0.06, 0.25, 0.46, 0.086) | (0.05, 0.22, 0.48, 0.087) |
| | (Pause + BCN) | (Pause + BCN(0,0)) | (Pause + BCN(MAX)) |
| 300 | (0.03, 0.20, 0.43, 0.072) | (0.00, 0.18, 0.40, 0.074) | (0.03, 0.33, 0.65, 0.130) |

# CS Utilization: 5ms – 20ms

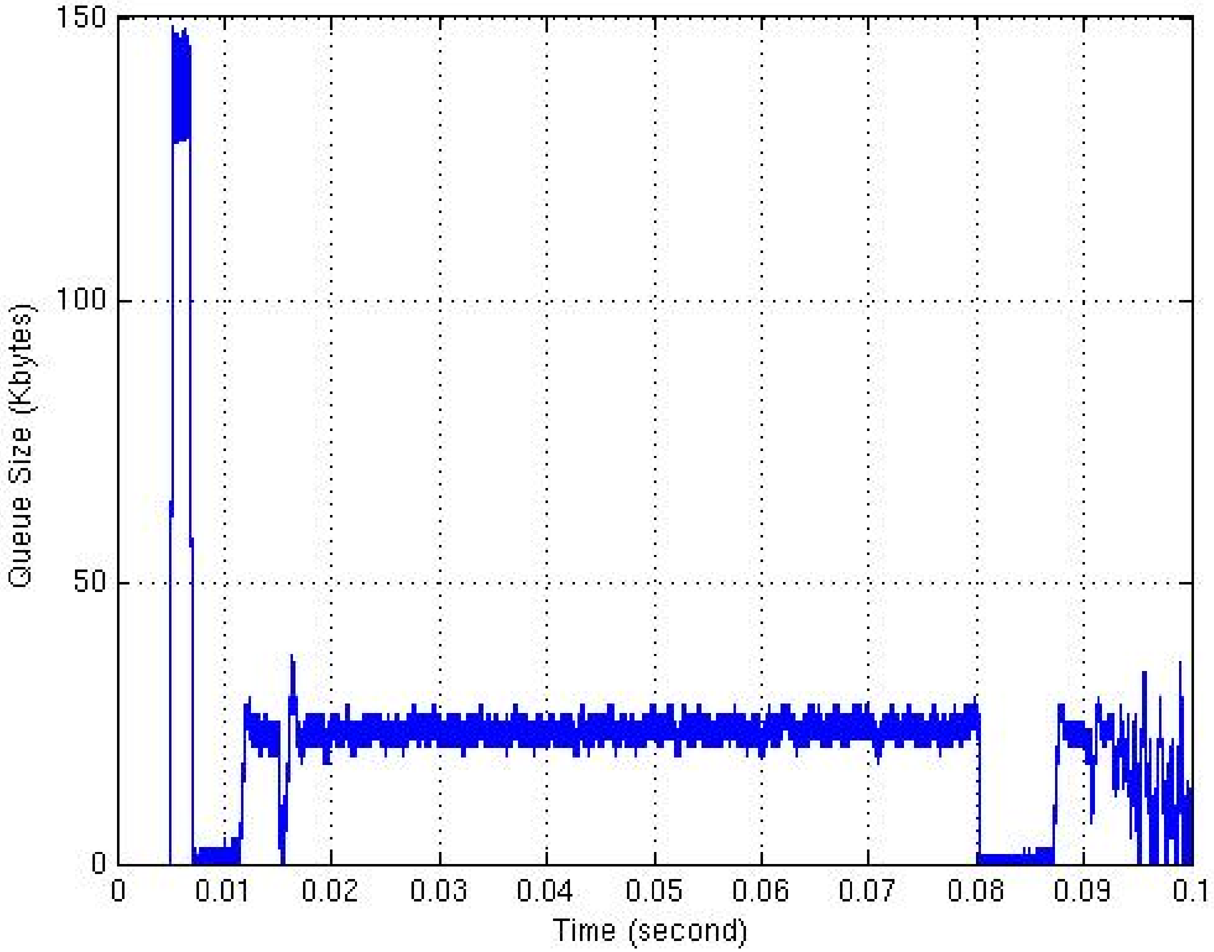
| With Drift | | | |
|---|---|---|---|
| # of Runs | (Min, Mean, Max, Std)<br>(BCN) | (Min, Mean, Max, Std)<br>(BCN(0,0)) | (Min, Mean, Max, Std)<br>(BCN(MAX)) |
| 300 | (1.00, 1.00, 1.00, 0.000) | (0.92, 0.98, 1.00, 0.027) | (0.94, 1.00, 1.00, 0.006) |
| | (Pause + BCN) | (Pause + BCN(0,0)) | (Pause + BCN(MAX)) |
| 300 | (0.99, 1.00, 1.00, 0.003) | (0.91, 0.94, 1.00, 0.016) | (0.78, 0.88, 0.95, 0.031) |

| Without Drift | | | |
|---|---|---|---|
| # of Runs | (Min, Mean, Max, Std)<br>(BCN) | (Min, Mean, Max, Std)<br>(BCN(0,0)) | (Min, Mean, Max, Std)<br>(BCN(MAX)) |
| 300 | (1.00, 1.00, 1.00, 0.000) | (0.92, 0.97, 1.00, 0.028) | (0.95, 1.00, 1.00, 0.007) |
| | (Pause + BCN) | (Pause + BCN(0,0)) | (Pause + BCN(MAX)) |
| 300 | (0.99, 1.00, 1.00, 0.003) | (0.90, 0.93, 1.00, 0.015) | (0.74, 0.87, 0.96, 0.034) |

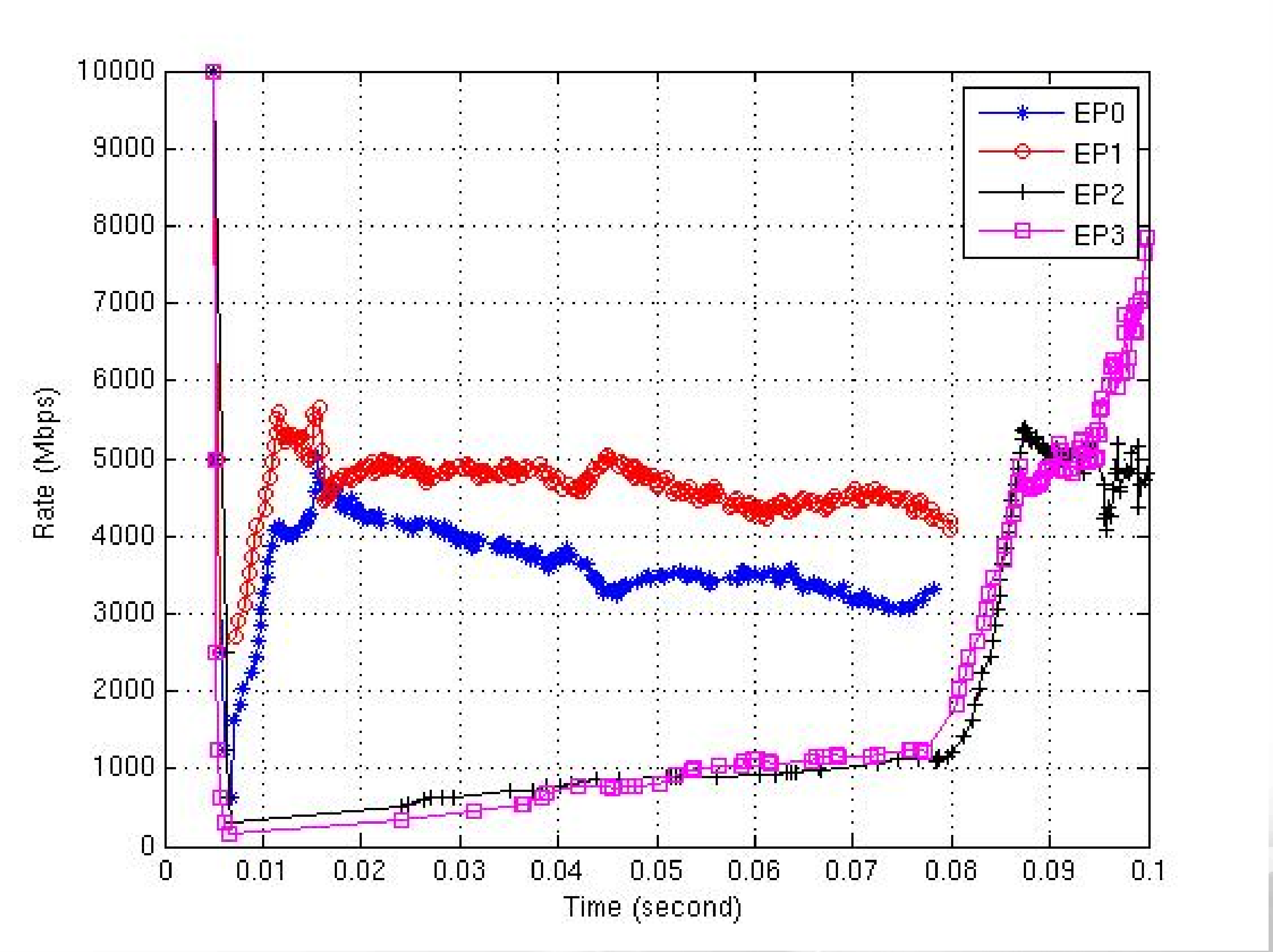
# Observation

- Basic BCN is best for transient inefficiency on congested flows
  - But without Pause drops cause inefficiency at higher layers
  - With Pause long pause assertion causes inefficiency in victim flows
- Basic BCN long duration of drops or pause assertion
- BCN (0,0) and BCN(MAX) best for reducing drops or pause assertion
- BCN(0,0) with no pause and no drift causes more unfairness than basic BCN
- BCN (0,0) causes more transient inefficiency than basic BCN
- BCN(MAX) somewhat more unfairness than BCN(0,0) but better less transient inefficiency.
- Drift improves BCN fairness

# Pause and BCN(MAX) (Wosrt Case): CS Queue



# Pause and BCN(MAX) (Wosrt Case): RLQ Rate


Rate (Mbps)
Time (second)
EP0
EP1
EP2
EP3