Sensitivity Analysis of BCN with ZRL Congestion Benchmark

Part 1

Mitch Gusat and Cyriel Minkenberg

IEEE 802

Dallas Nov. 2006

Outline

- Next phase: BCN validation
  - larger datacenter networks
  - demanding traffic patterns
- ZRL congestion benchmarking
  - congestion taxonomy and a practical toolbox
- Analytical dual ranking: The APS method
  - BCN's algorithmical sensitivity to parameters
  - Parameters' sensitivity to benchmarking traffic
- Simulation results
  - validation of analytical selection
  - parameters' sweep: stability plane
- Conclusion

Next phase of BCN validation

- Baseline BCN: validated by multiple parties
  - joint effort of the .1au adhoc simulation teams
- Basic scheme is functional
  - for detail conclusions see .1au repository

- Next: BCN w/ *larger networks under stress traffic*

- How to proceed?
  - Empirical approach: Brute force simulations (see next foil)
  - More rigorous approach: ZRL congestion benchmarking
    - Iterate between analytical and simulation models to systematically parse the combinatorial tree and reduce the dimension of the parameter space

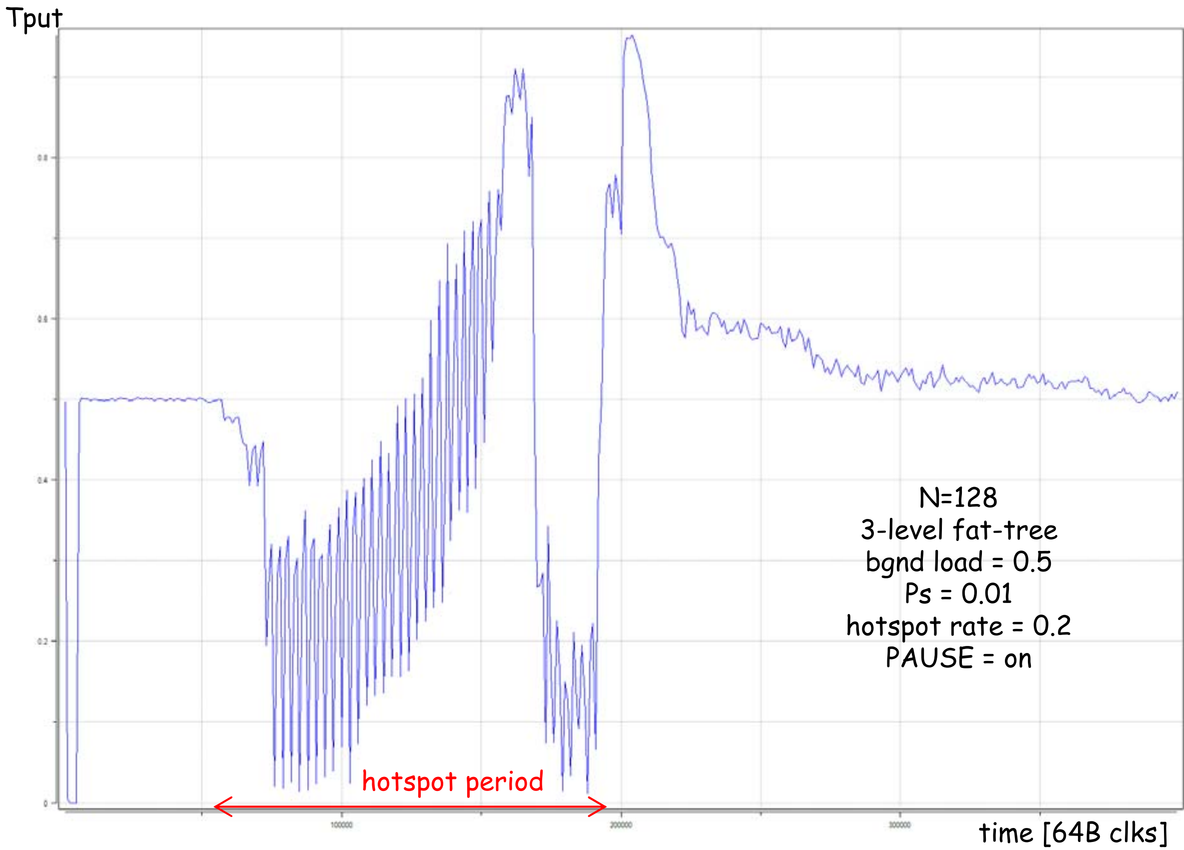
Empirical approach: Brute force simulations

Multi-dimensional problem

1. no. nodes
2. switch / adapter arch.
3. topology
4. LL-FC settings
5. BCN params
6. traffic scenario
7. metrics of interest
8. no. of simulation points

⇒ Combinatorial explosion of an 8D (actually 20+ dim's) search space.

⇒ Not practical for standard work



BCN with baseline settings: unstable.

Which dimension to explore 1st?

A More Rigorous Alternative

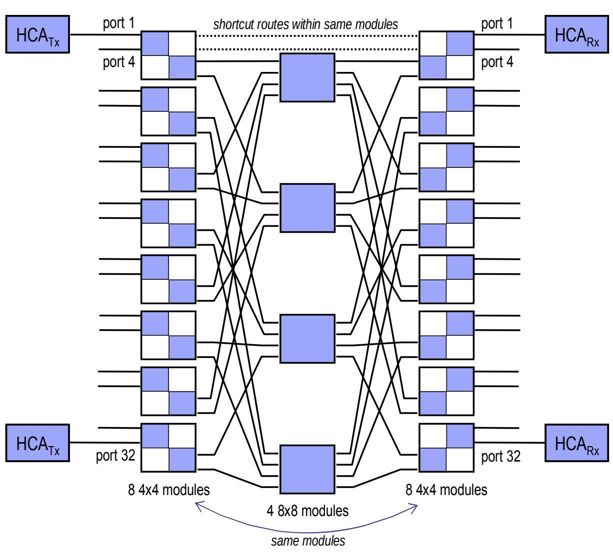
- Dimensions 1-3 (architectural) are determined by
  - market of datacenter and HPC
  - 802 architectural definitions (e.g., ideal OQ)

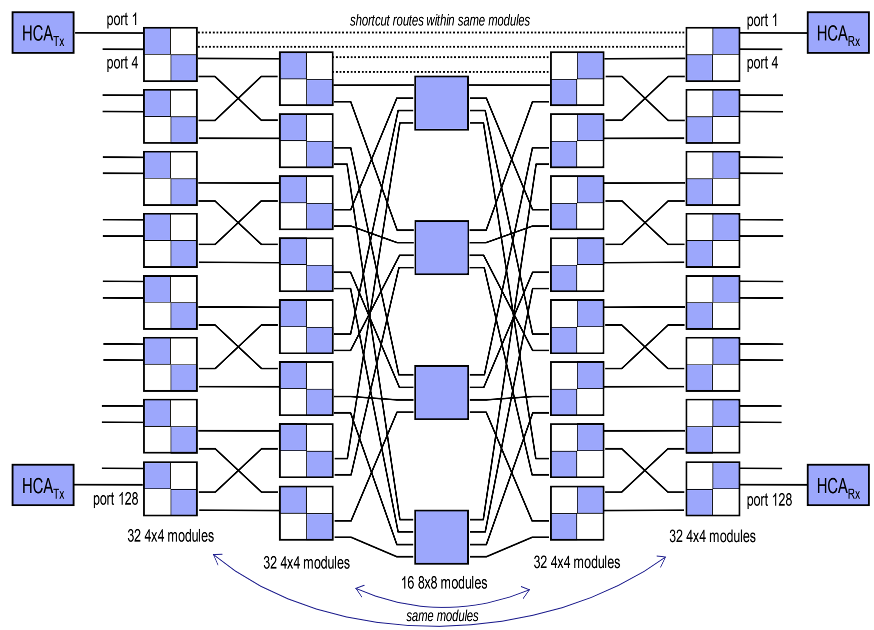
- Dim's 4,5 (scheme settings) => Our main target.

- Dim's 6-8 (methodology) => Toolbox

- Toolbox proposal: "ZRL Congestion Benchmarking"
  1. Benchmarks designed for datacenter environments
  2. Combines analysis w/ simulation in a systematical method
  3. Tried and improved thru work in related standards.

Baseline Topology Proposal: Bidir Fat Trees (FT)





- 2-level / 3-stage bidir MIN
- Simulate: 8 – 32 nodes
- Time per run: < 1hr

- 3-level / 5-stage bidir MIN
- Simulate: 128 – 2K nodes
- Time per run: TBD

Fat-trees: Scalable, w/ excellent routing and performance properties. Optimum performance/cost with current trends in technology. Can emulate <u>any</u> k-ary n-fly *and* n-cube topology. Large body of knowledge.

Toolbox-1. Traffic: ZRL Congestion Benchmark

- Source nodes generate* one or more hotspots according to matrix $[\boldsymbol{\lambda}_{ij\_hot}]$: $t_{p->q} = \alpha_{k\_hot}\,[\boldsymbol{\lambda}_{ij\_hot}]{:}t_{p->q}$, $[\boldsymbol{\lambda}_{ij\_hot}]$ is specified** per case as below

1. Congestion **type**: IN- or OUT- put generated

2. Hotspot **severity**: HSV = $\boldsymbol{\lambda}_{aggr}$ / $\boldsymbol{\mu}_{HS}$, $\lambda_{aggr} = \sum \lambda_i$ at hotspotted output, μ_{HS} = service rate of the HS
   - *Mild* 1 < HSV <= 2
   - *Moderate* 2 < HSV <= 10
   - *Severe* HSV > 10.

3. Hotspot **degree**: HSD is the fan-in of congestive tree at the measured hotspot
   - *Small* HSD < 10% (of all sources inject hot traffic)
   - *Medium* HSD ~ 20..60%
   - *Large* HSD > 90%.

* Traffic generation is a Markov-modulated process of burstiness B (indep. dimension)

**Metrics and measurement methodology are subject of another deck

Toolbox-2: BCN Parameters. How to proceed?

BCN entails 6 params

1. Equilibrium threshold Q_{eq}
2. Rate unit R_u
3. Sampling rate P_s
4. Feedback weight W
5. Increase (additive) gain G_i
6. Decrease (multiplicative) gain G_d

Next step?

a) The empirical approach is unsustainable because it generates too many singular points, as seen on foil #4

b) A purely analytical approach is difficult owing to non-linearity of model. Would also require validation by simulation.

c) However, a combined analytical and simulation method is feasible!

Reduction of Simulation Space: Dual Ranking

- Using ZRL Benchmarking, the smallest simulation space is given by the tuple product
  - SimRuns = {topology, HS type, HS severity, HS degree, burstiness} x {BCN param} = 2*2*3*3*4 x {BCN param} = 144 x {6D}

- SimRuns = 144 x $\{Q_{eq,} R_{u,} P_{s,} W, G_{i,} G_d\}$ ... still a VERY large space!

- Further reduction by (simplified) dual ranking analysis
  1. algorithmical sensitivity to BCN params: which param matter most?
  2. parametrical sensitivity to traffic: which benchmarks are critical?

Next: Algorithmic and parametrical (AP) sensitivity of BCN

Sensitivity is often a more accurate metric of stability margin than either gain or phase margin! However, here we didn't use canonical sensitivity.

Ranking by AP Sensitivity - 1

From BCN stability model

1. Conservation: $dq/dt = HSD*\lambda(t) - \mu_{HS}$ =>
2. $q(s) = HSD* \lambda(s) / s$
3. Feedback: $Fb(t) = -(q(t) - Q_{eq}) + w*(dq/dt) / (\mu_{HS} * p_s)$ =>
4. $Fb(s) \approx G * [1 + w*s / (\mu_{HS} * p_s)]$

5. AI: $d\lambda(t)/dt = G_i*\lambda(t)* p_s*Fb(t-\tau)$
6. $\delta AI(t)/\delta Fb(t-\tau) = G_i*p_s*\mu_{HS}/HSD$ =>
7. AP sensitivity of $G_i = \delta AI(t)/\delta Fb(t-\tau) * HSD/(p_s*\mu_{HS})$

8. MD: $d\lambda(t)/dt = G_d*\lambda(t)*\lambda(t-\tau)* p_s*Fb(t-\tau)$
9. $\delta MD(t)/\delta Fb(t-\tau) \approx G_d*p_s*(\mu_{HS}/HSD)^2$ =>
10. AP sensitivity of $G_d = \delta MD(t)/\delta Fb(t-\tau) * (\mu_{HS}/HSD)^{-2} / p_s$.

$q(t)$ =queue occupancy; HSD=no. of hot flows, each with rate $\lambda(t)$, at hotspot served w/ rate μ_{HS}

Ranking by AP Sensitivity - 2

(7,10) =>

a) p_s *directly* impacts G_i and G_d

- 1st order sensitivity on p_s

b) G_i and G_d depend on the HSD/μ_{HS} ratio

- congestion w/ high HSD and low μ_{HS} stresses stability

(10) =>

c) G_d is *more* sensitive than G_i to the HSD/μ_{HS} ratio (squared)

(4,7,10) =>

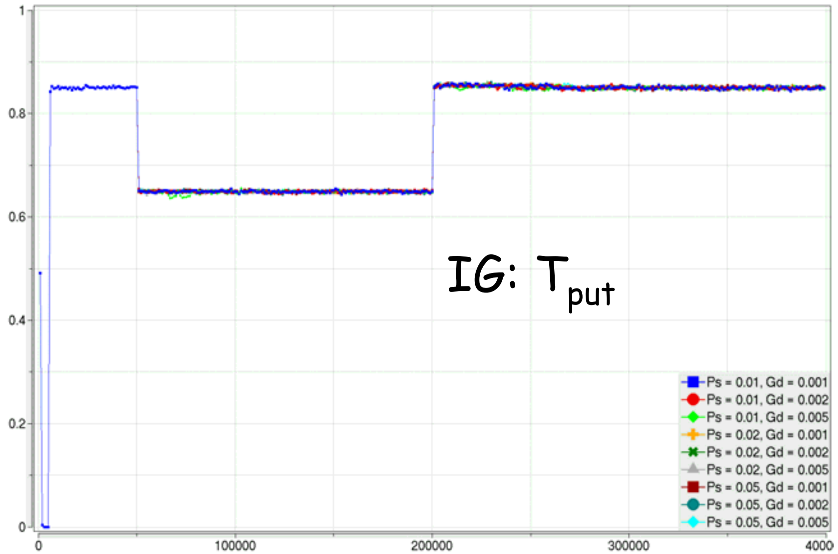
if denominator ~ f ($p_s * \mu_{HS}$), where $p_s \ll 1$ and $\mu_{HS} \leq 1$, -> the hotspot drain rate *further* increases the sensitivity to p_s

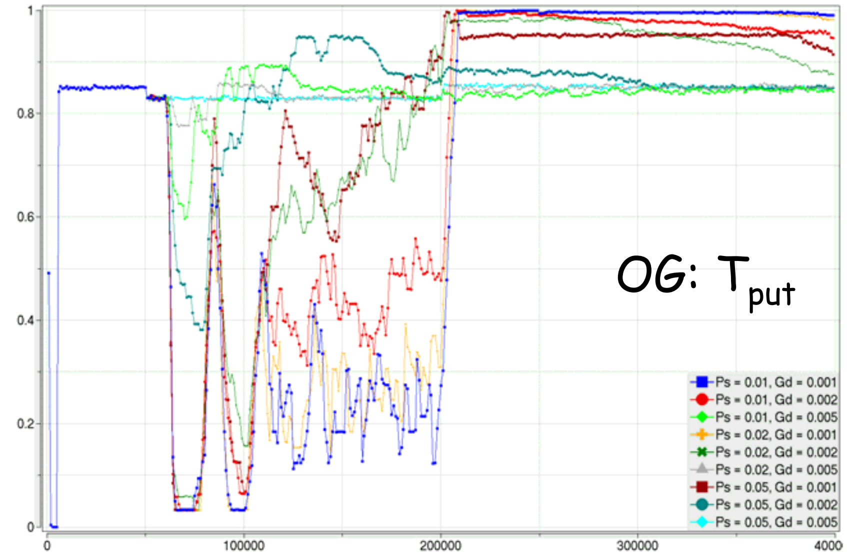
d) everyting else being equal, *output-generated* (OG) congestion is more stressful for BCN's stability than IG

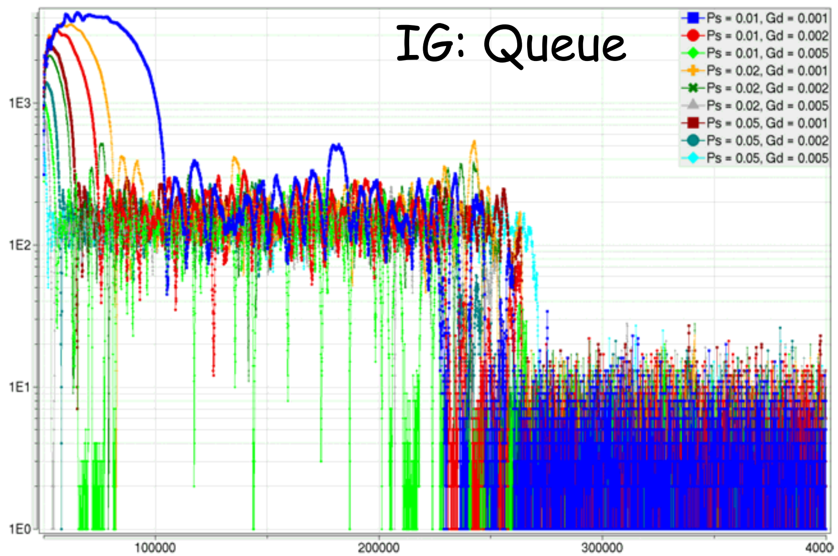
What to begin with?

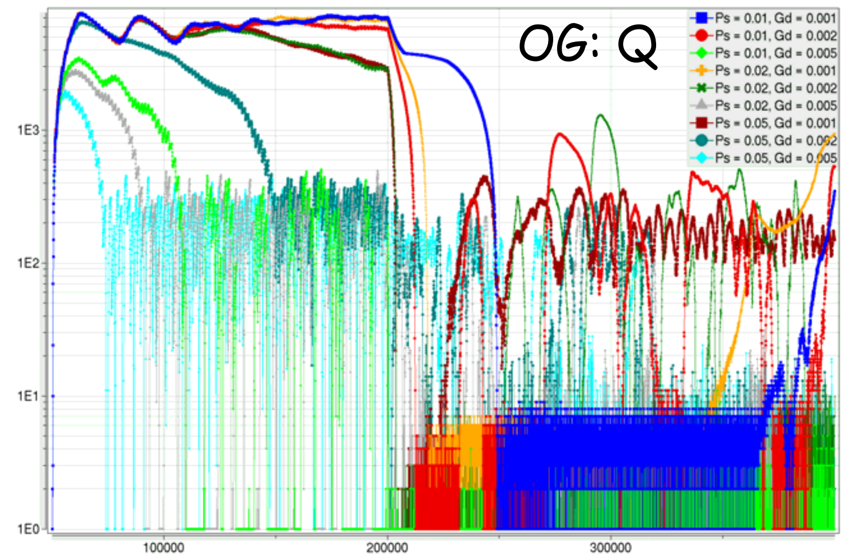
- BCN params: p_s and G_d
- Traffic: Output-generated congestion w/ high HSD and low μ_{HS} .

Qualitative Validation: Input- vs. Output-Generated HS









Simulations Confirm Our Sensitivity Ranking

- OG requires higher control effort than IG
  - Slower throughput recovery; overshoot
  - Higher queue size fluctuations
  - Less stability margin: more sensitive to parameter settings
- BCN's impulse response improves as P_s and G_d increase (within bounds!)
  - Applies to both scenarios => as P_s and G_d increase, so does the system's distance between pole(s) and origin... up to a point
- Next: Simulation-based sensitivity analysis of P_s and G_d

Simulation Overview

- Single-stage network, 32 nodes
- Shared-memory switch
- Background traffic is uniformly distributed
- All frames minimum size (64 B, time slot = 51.2 ns)
- No TCP/IP, raw Ethernet!

- Parameters
  - Mean load λ
  - Mean burst size B
  - Shared-memory size M
  - Round-trip time RTT (in slots)
  - BCN parameters (P_s, G_d, G_i, Q_{eq}, W, R_u)

- Metrics
  - Throughput (aggregate and per port/flow)
  - Latency (measured per burst)
  - Queue length (congested queue)
  - Fairness (RJFI, ALFI)
  - Number of PAUSE and BCN frames sent

Switch and Adapter Model

- Shared-memory output-queued switch
- PAUSE enabled
  - Global high- and low-watermark memory threshold trigger pause and unpause
  - High watermark $T_h = M - N*(RTT*B + L_{max})$
  - Low watermark $T_l = T_h / 2$
  - PAUSE renewed before expiry (take into account RTT)

- VOQ-ed per end node
- Round-robin service discipline
- Number of rate limiters unlimited
- Egress buffer flow-controlled using PAUSE (high/low watermarks)

Lossless operation:
No frame drops due to buffer overflows!

Traffic Scenarios

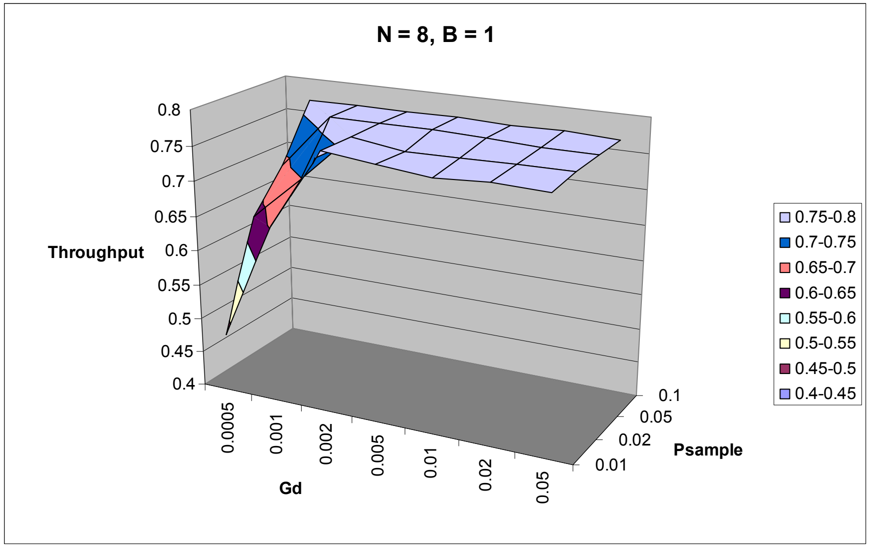
- Output-generated hotspot
  - Service rate of output 0 is reduced to 20% of full line rate
  - Results in an *N*-degree hotspot
  - Without CM, *aggregate* throughput is limited to 20% due to hogging

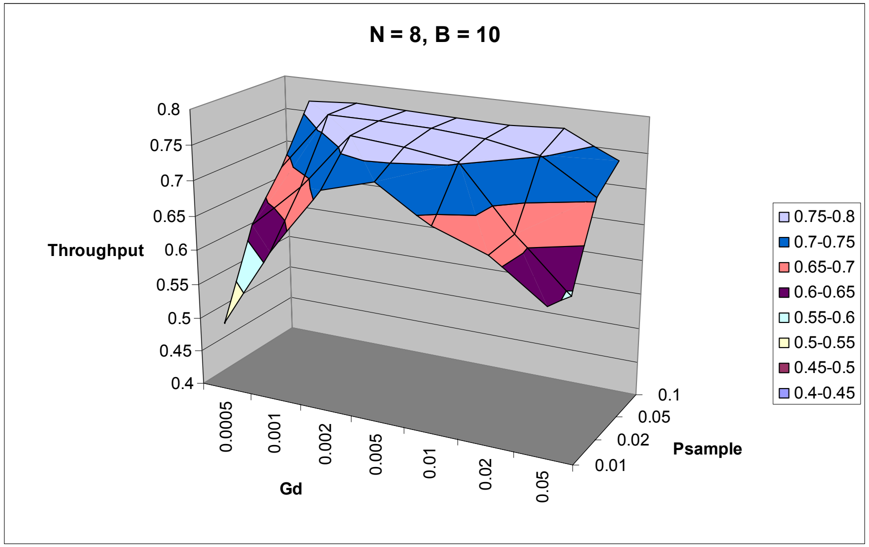
Initial Param Settings

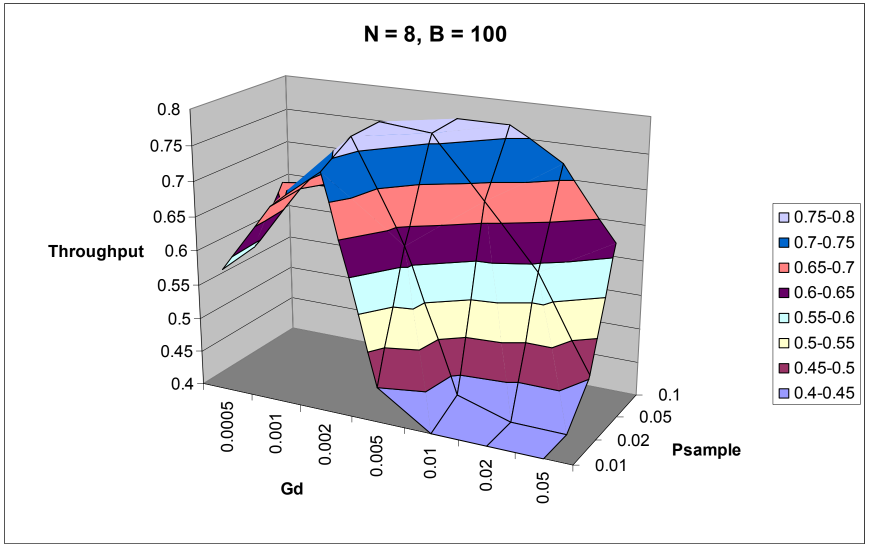
1. Q_{eq} <= M / N (memory is partitioned to reduce hogging)
2. $R_u = R_{max} / 1000$
3. $P_s = [0.01, 0.1]$
4. W = 1
5. $G_i = 1$
6. $G_d = [0.0005, 0.05]$

Note: Above settings may be neither optimal nor a baseline match.

Results: OG hotspot (N=8)

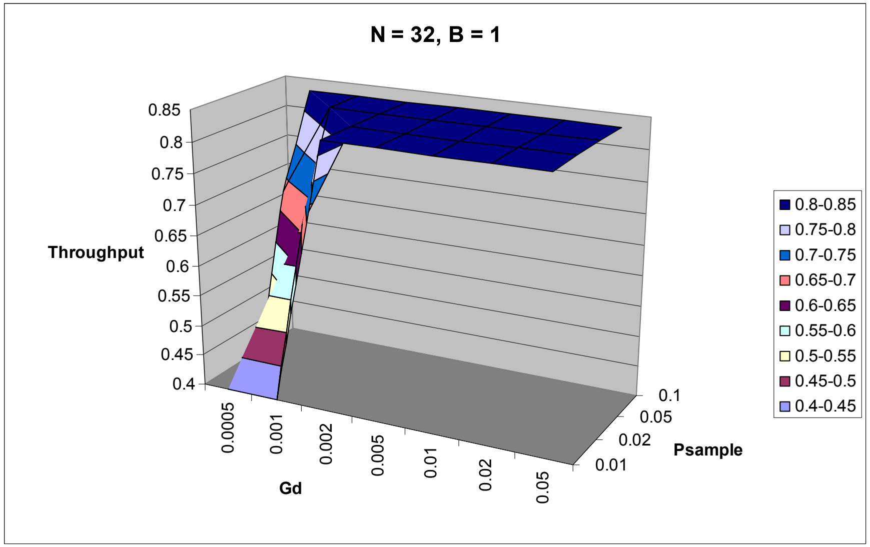


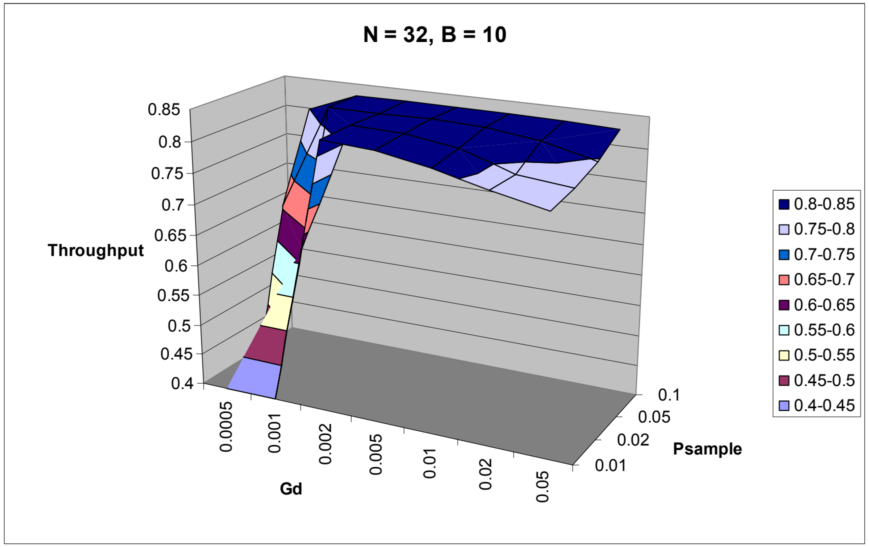


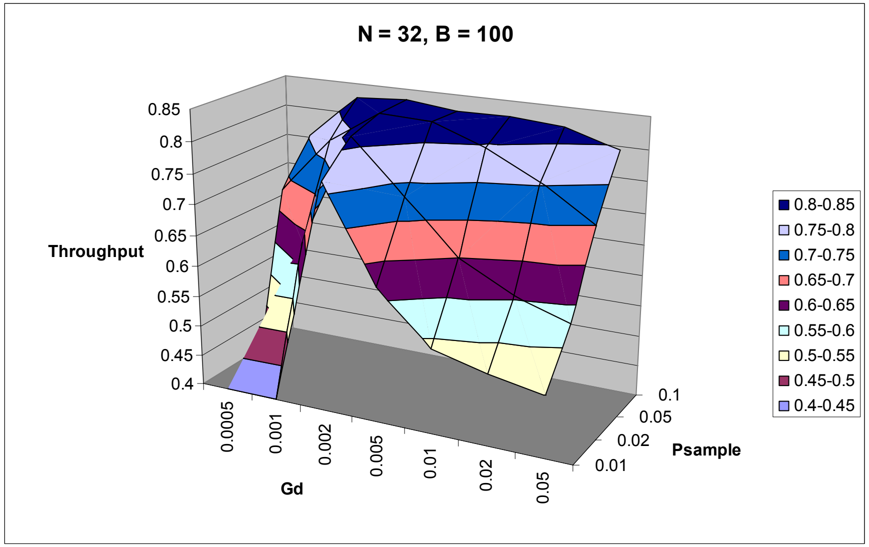


- RTT=0, M=256*N, Q_{eq}=M/N
- Throughput measured during hotspot
- Hotspot rate = 20%
- $Tp_{max} = \lambda * (N-1)/N + 0.2/N$
- λ=85%, N=8 => Tp_{max} = 0.77
- Varying G_d and P_s

Results: OG hotspot (N=32)

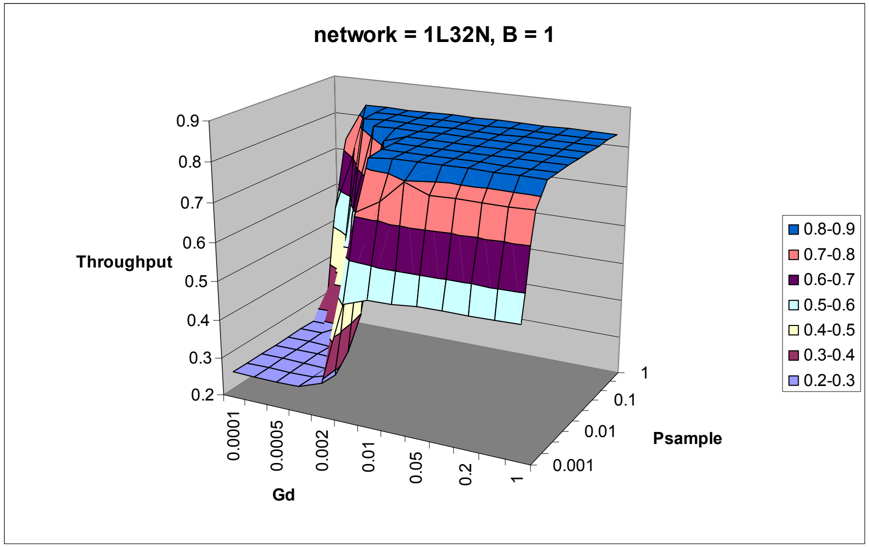


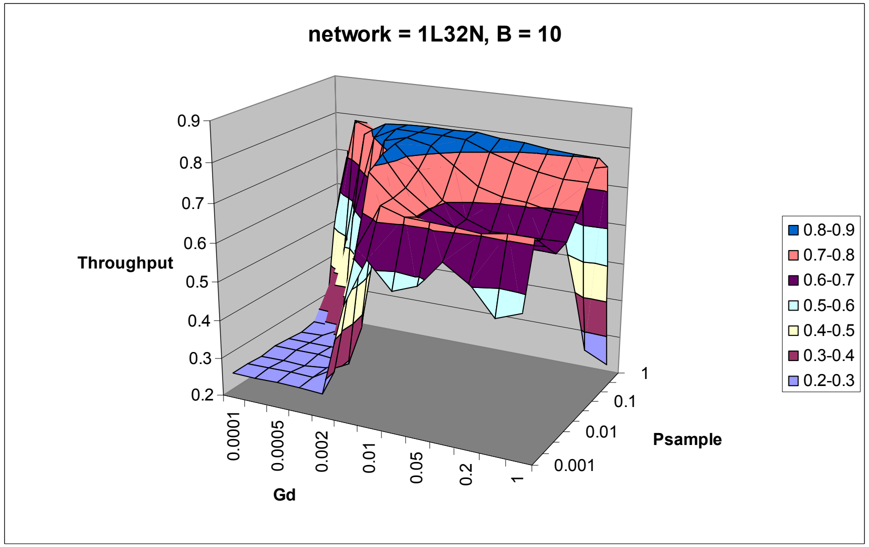


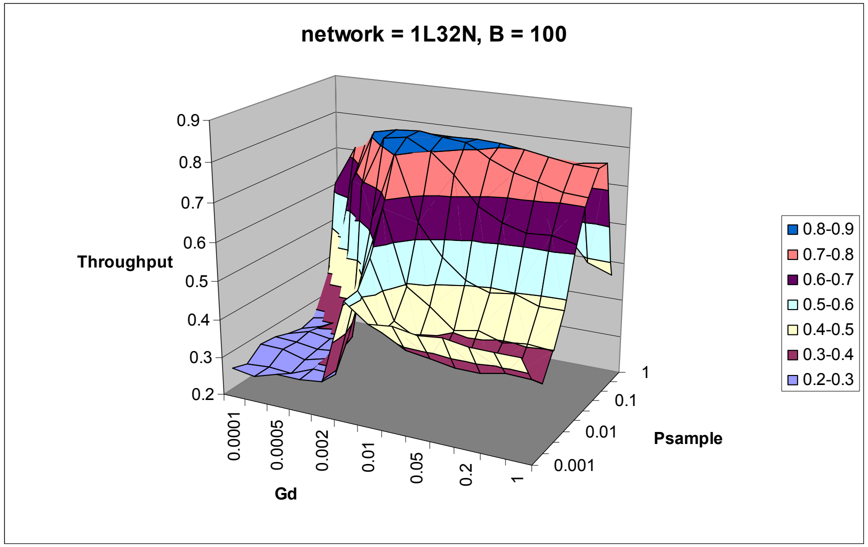


- RTT=0, M=256*N, Q_{eq}=M/N
- Throughput measured during hotspot
- Hotspot rate = 20%
- $Tp_{max} = \lambda * (N-1)/N + 0.2/N$
- λ=85%, N=32 => Tp_{max} = 0.83
- Varying G_d and P_s

Results with M/(2N) Memory Partitioning: OG hotspot 1L32N





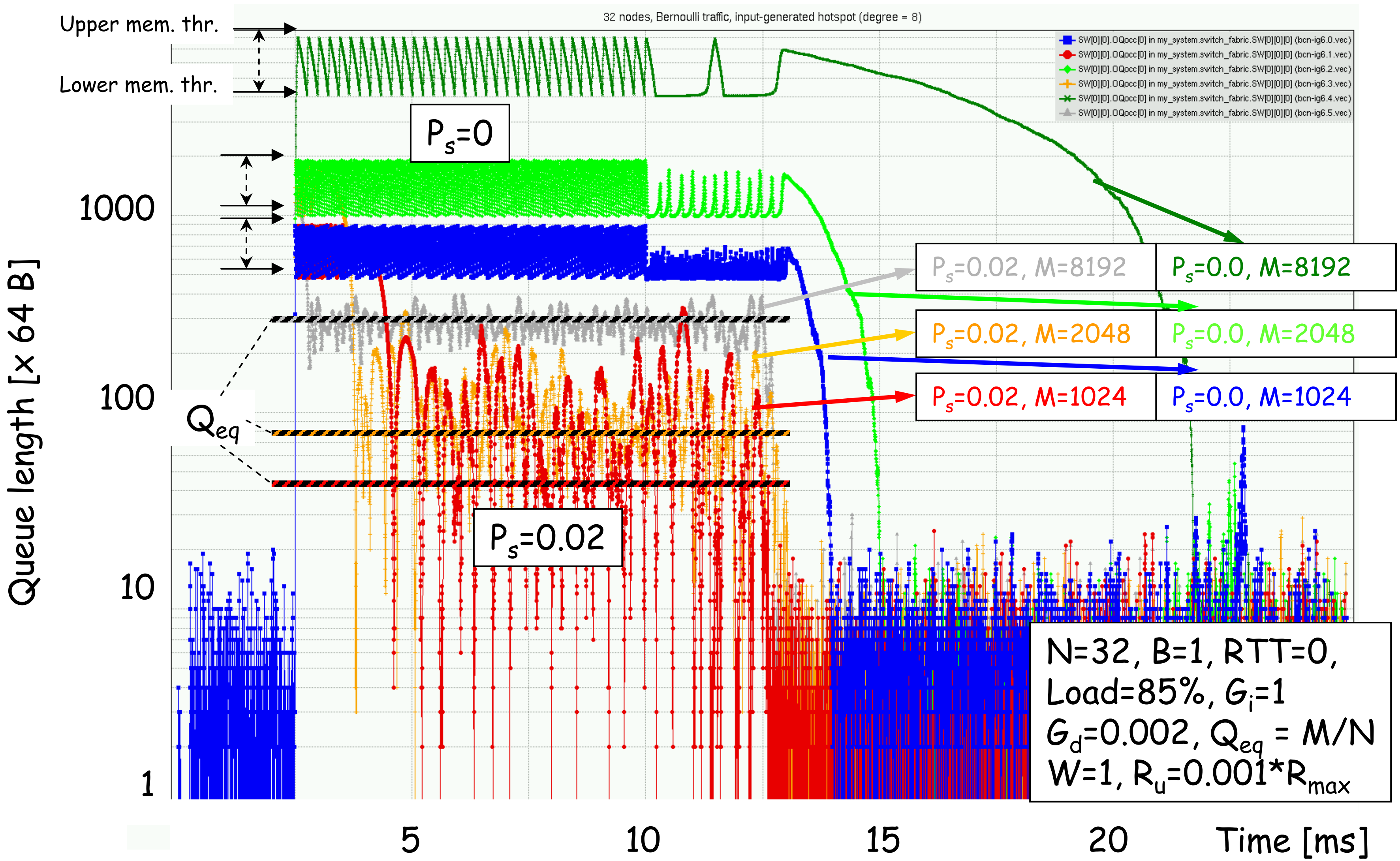


- RTT=0, M=256*N, Q_{eq}=M/(2N) !
- Throughput measured during hotspot
- Hotspot rate = 20% => severity = 85%/20% = 425%
- $Tp_{max} = \lambda * (N-1)/N + 0.2/N$
- λ=85%, N=32 => Tp_{max} = 0.83
- Varying G_d and P_s

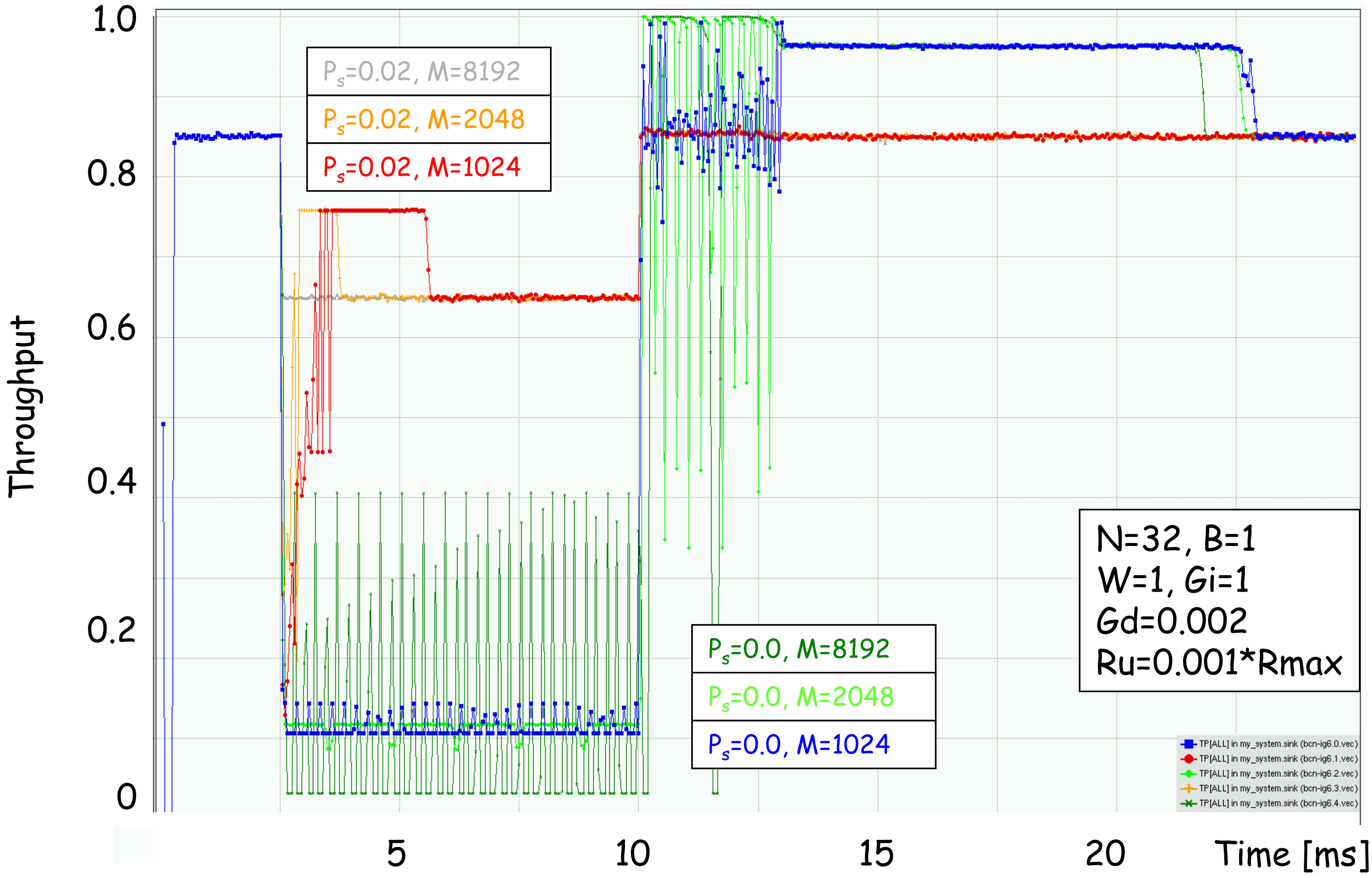
IG results

- Input-generated severe hotspot
  - Uniform background traffic load = 85%
  - Multiple (HSD) inputs send 100% of their traffic to output 0
    - Primary HSD = 8 (all the other also send a smaller quota)
    - Hotspot is targeted by 8 hot flows and 24 background flows
    - Aggregate severity = (8*100% + 24*85%/32) = 863%
  - Without BCN, aggregate throughput is limited to about 100% / (HSD((N-1)/N)+1)

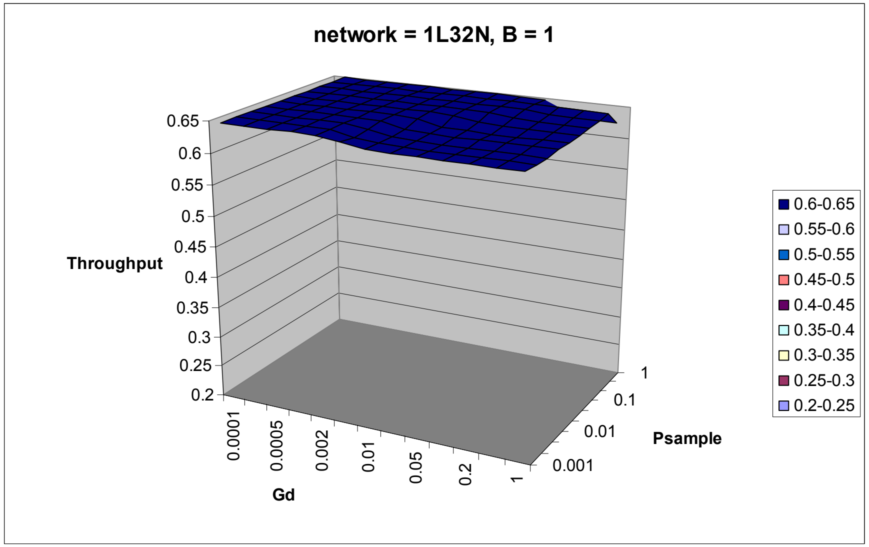
Results: Input-gen'd hotspot (1)

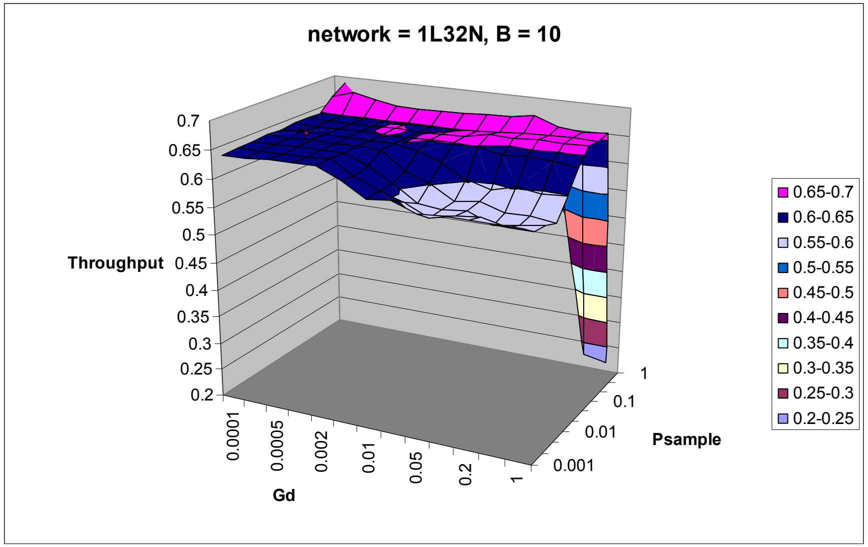


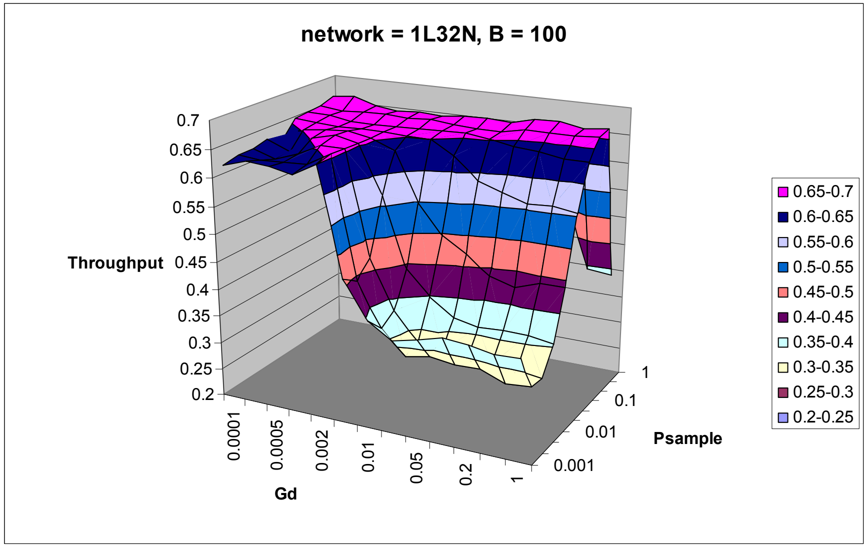
Results: Input-gen'd hotspot (2)



Results with M/(2N) Memory Partitioning: IG hotspot 1L32N







- RTT=0, M=256*N, Q_{eq}=M/(2N)
- Throughput measured during hotspot
- Hotspot severity = 863%
- Tp_{max} = 0.65
- Varying G_d and P_s

Conclusions

- Analytical and simulation modeling show that BCN's stability and performance depend on
  - Two 1st order params: p_s and G_d
  - Type of traffic: Output-generated congestion is a stress test
- Optimal ranges for OG (assuming fixed W*, G_i, R_u, Q_{eq})
  - P_s = [0.02, 0.05]
  - G_d = [0.002, 0.005]
- Burstiness also determines sensitivity
  - Large bursts (MTU-Jumbo) increase the sensitivity
- Upcoming
  - Increase network size to 128, with 2 and 3 levels.

* In simulations W proved less sensitive than we've analytically expected