The Role of Mathematical Modeling in the Design of Protocols for High-Speed Networks

R. Srikant

ECE & CSL

University of Illinois at Urbana-Champaign

Outline

- End-User Protocol:
  - *TCP-Illinois: A Loss and Delay-based congestion algorithm*
  - What does math modeling tell us about the algorithm?
  - Joint work with Shao Liu and Tamer Başar
- Router Buffer Sizing:
  - *The impact of core-router buffer sizing on the performance of TCP and other protocols*
  - Mathematical models of file arrivals/departures versus models of fixed number of users: what insight does each model provide?
  - Joint work with Ashvin Lakshmikantha and Carolyn Beck

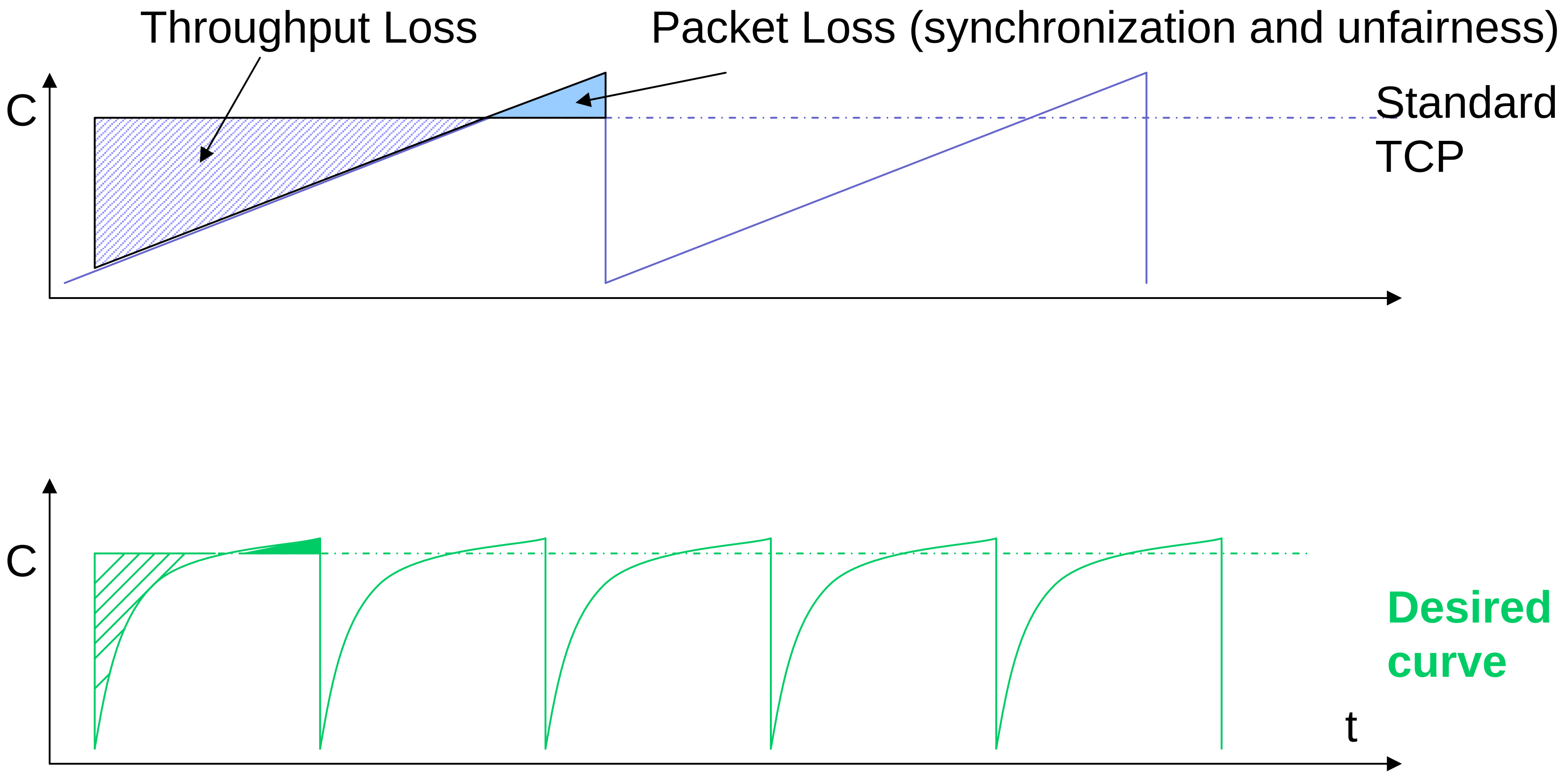
Prior work

- High-speed protocols: HS-TCP (Floyd), FAST (Low et al), Scalable TCP (Vinnicombe, T. Kelly), H-TCP (Shorten, Leith), Compound TCP (Tan et al), BIC (Rhee et al), LTCP (Reddy et al)….
- Models of Protocol Dynamics: Chiu-Jain, Kelly et al, Low-Paganini-Doyle et al, Kunniyur-S., Misra-Hollot-Towsley et al, Baccelli-Hong, Shorten-Leith-Wirth, Altman, Avrachenko et al,…
- Core router buffer sizing and TCP: Appenzellar et al, Enachescu et al…
- Fixed-user models: Raina-Wischik, Deb-S.,…
- File arrivals and departures: Das-S., Roberts et al, Dhamdhere-Dovrolis,…

New TCP for High Speed Networks (TCPv2)

- Requirements for TCPv2:
  - Efficiency: larger throughput than TCP in high-speed networks
  - Fairness: allocation among competing users should be fair
    - What is fair?
  - Compatibility with TCP: TCP/TCPv2 should not be too small
  - Incentive for TCP users to switch to new protocol: TCPv2 > TCP
- Inheritance from TCP:
  - Increases W if no congestion, decreases W if congestion
- How we can modify TCP? → Two directions:
- I: How to detect congestion?
  - Both packet loss and queueing delay are congestion signals.
  - Standard TCP uses loss only. We can use delay, or both.
- II: How to increase/decrease W after detection is made?
  - TCP uses AIMD. Can choose other options.

Ideal window curve for loss-based algorithms: Motivation for TCP-Illinois

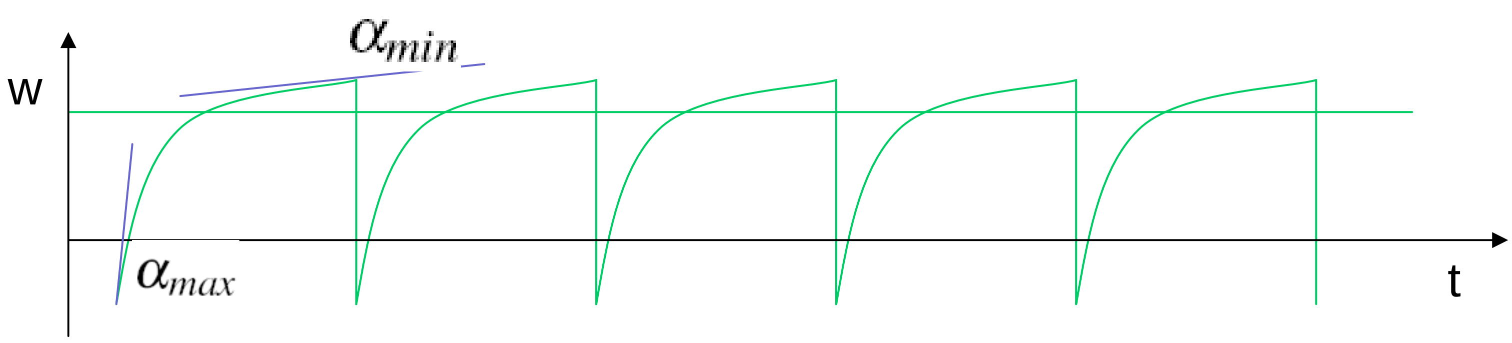


Key ideas: TCP-Illinois

- Key ideas:
  - Loss determines **whether** W **increases or decreases**
  - Delay determines **amount by which** W **increases or decreases**
- Algorithm
  - Queueing delay $d=RTT-RTT_{min}$
  - Estimate maximum queuing delay d_m
  - d far from d_m:
    - congestion is not imminent or not severe,
    - increase W rapidly,
  - d close to d_m:
    - congestion is imminent or severe
    - increase W slowly

TCP-Illinois

- Congestion avoidance phase: Concave-AIMD
  - W←W+(α(d)/ W) for each ACK
    - α ⇩ as delay ⇧
  - W←W-β W for each loss
  - Result: W is a concave curve
    - W ⇧ → delay ⇧ → α ⇩ → Δ W ⇩



Stochastic Model: Congestion Event

- Window evolution:
  - Baccelli & Hong; Shorten, Leith & Wirth
  - Congestion event: link drops packets
  - Multiple users, W is column vector of all W_i for all i
  - Consider the window size before/after each congestion event: index k, before W[k], after $W[k^+]$
  - W[k] is then a discrete-time random process

$$W_i[k^+] = W_i[k]\theta_i[k]$$

  - $\theta_i[k]$: window backoff factor, a random variable

Window Dynamics

Between congestion events:

$$W_i[k+1] = W_i[k^+] + \int_{t_k}^{t_{k+1}} \frac{\alpha_i(t)}{T_i(t)} dt$$

Congestion event: $\sum_{i=1}^{N} \frac{W_i[k]}{\bar{T}_i} = C', \ \forall k$

At congestion event:

$$W_i[k^+] = W_i[k]\theta_i[k]$$

$$\theta_i[k] := 1 - E_i[k]\beta_i[k]$$

$$E_i[k] := \begin{cases} 1 & \text{if i backs off} \\ 0 & \text{else} \end{cases}$$

$$q_i[k] := Prob(E_i[k] = 1)$$

All users may or may not back off during a congestion event: at least one user does

Unsynchronized Backoff Model

- At least one flow experiences a loss during a congestion event, but not all flows may experience congestion
- The probability of loss for user i, $q_i[k]$ is an increasing continuous function of $x_i[k]$:
  - Models the fact that a user with a larger rate is more likely to experience loss
  - E[X[k+1]|X[k]]=A(X[k]) X[k], where A(X[k]) =E[A[k]]
  - The system is nonlinear and quite complicated
- Result 1: Stochastic stability still holds (Shorten et al):
  - unique invariant distribution and ergodicity
- Result 2: Fairness (only at congestion events):
  - $E[W_i[k]T_i q_i(\mathbf{x}[k])]$ is approximately the same for all i
  - Recall x=W/T
  - If q is proportional to x, $E(W^2)$ is independent of T

Unsynchronized Backoff: A Realistic Model

- A special case:
  - At each congestion event k, the total # of packets dropped is a random variable independent of k.
  - Prob(a dropped packet belongs to i)=$x_i[k]/C'$
- q is approximately proportional to x
  - if M (# of packets dropped) << N (# of flows)
  - Light congestion and E[W*W] the same for all users
  - If Var(W)<<E[W]*E[W], E(W) approximately the same for all users
- In contrast if M is large compared to N
  - Leads to small W for large RTT flows

Behavior of TCP-Illinois

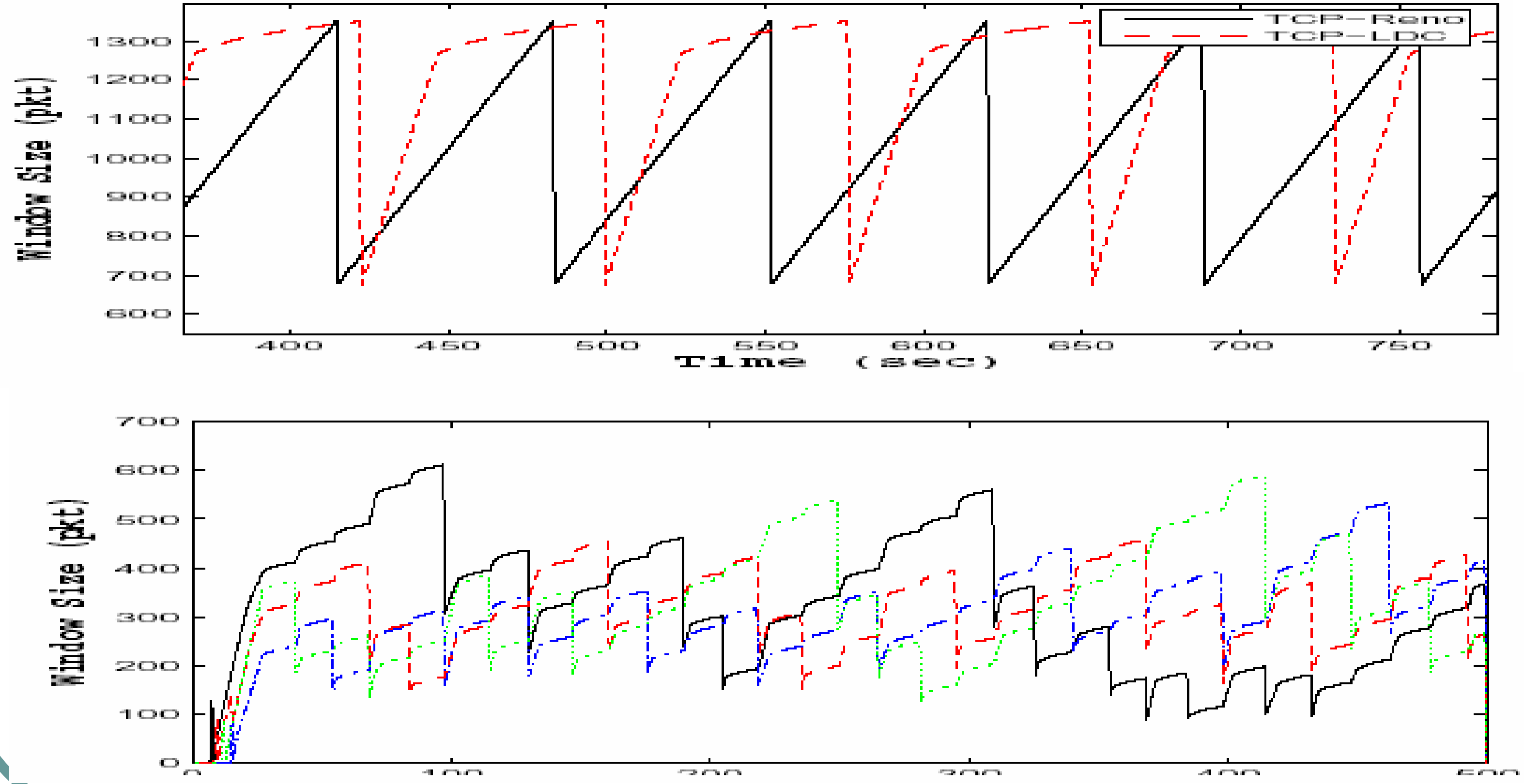
- Increment rate α just before congestion determines the number of packets dropped, M:
  - E[M]=(sum of all α) / 2
- M determines the window backoff behavior, which determines fairness:
  - M<<N: completely unsynchronized: $W \propto T^0$
  - M>>N: completely synchronized: $W \propto 1/T$
  - In the middle: partially synchronized, $W \propto 1/T^a$, $0<a<1$
- TCP-Illinois:
  - small α before congestion
  - small M and hence, unsynchronized backoff
  - W independent of T
  - fairness similar to Reno

Comparison with diff eq models

- The resolution of the diff eq model is not sufficient to capture the behavior of the protocol at congestion events
- Diff eq models cannot capture synchronization behavior at loss events
- There are well-known examples in which the diff eq model is stable, but simulations show wild oscillations
- Lessons to be learnt?
  - Diff eq models are appropriate when used with AQM schemes with marking
  - Good for design under the assumption that congestion feedback is ideally spread out among the flows
  - Fine-grained matrix models seem to capture behavior at congestion events more accurately

Simulation: concave curve

TCP-Illinois parameter settings for 10 Mbps:
α_{min}=0.1, α_{max}=10,β_{min}=1/8, β_{max}=1/2, W_{thresh}= 15



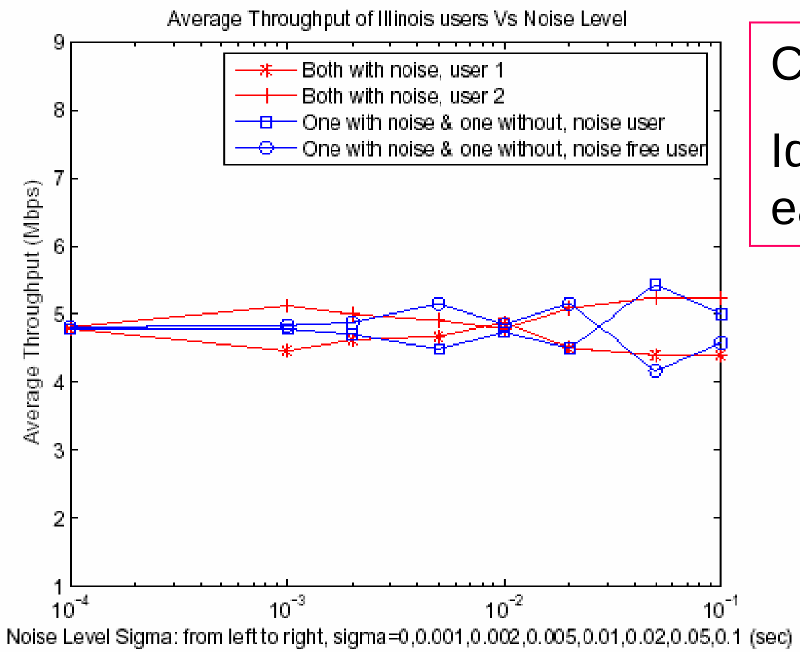
Efficiency & Compatibility

C=100 Mbps: Three experiments

	User 1	User 2	User 3	User 3
TCP-Illinois: (RTT=100 ms)	Illinois: 22 Mbps	Illinois: 22 Mbps	Illinois: 22 Mbps	Illinois: 23 Mbps
Illinois vs. Reno (RTT=100 ms)	Reno: 16 Mbps	Reno: 16 Mbps	Illinois: 24 Mbps	Illinois: 25 Mbps
Window sizes for TCP-Illinois	270 packets (RTT: 60 ms)	273 packets (RTT: 80 ms)	269 packets (RTT: 100 ms)	275 packets (RTT: 120 ms)

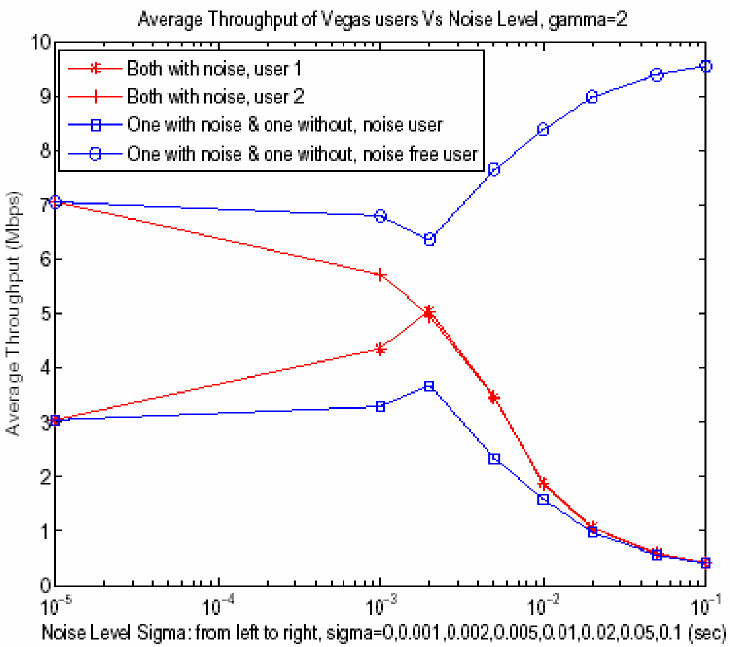
Inaccurate delay measurement

- RTT=dp+dq+n → measured dq=dq+n
- dq* is the dq if Vegas works effectively
- Vegas fails to work if E[n]>dq* → Vegas is not robust to n
- Illinois is very robust to n



C=10

Ideal: each 5

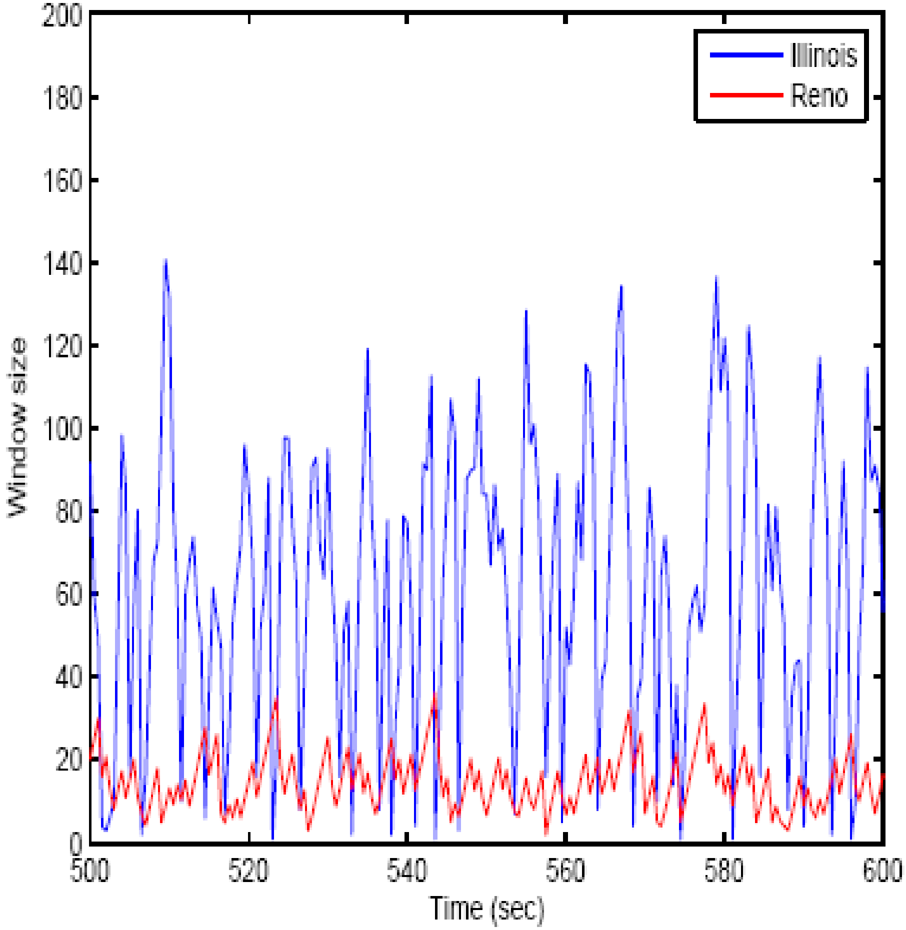


Dealing with random losses

- Make the decrease factor also a function of delay
- Congestion avoidance phase: Concave-AIMD
  - W←W+(α(d)/ W) for each ACK
    - α ⇩ as delay ⇧
  - W←W-β(d) W for each loss
    - β ⇧ as delay ⇧
- Suppose you have a wireless link on the path and packets are dropped due to non-congested related reasons, then β would be small and thus, would not decrease the window dramatically

TCP-Illinois over a wireless link

Illinois   Reno



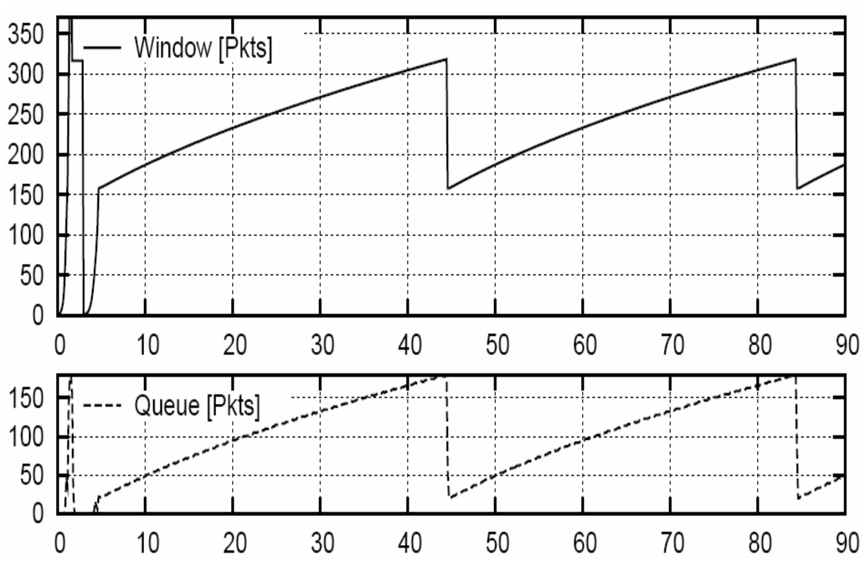
C=40 Mbps
RTT=100 ms
B=200 pkts
p=0.005

Summary: Congestion Control

- TCP-Illinois
  - Combines loss and delay
  - Loss determines direction and delay adjusts rate of window change
  - Achieves better throughput than TCP
  - Allocates network resources fairly
  - Compatible with Reno and provides incentive to switch
- Stochastic Matrix Model:
  - Rate of increase just before congestion event determines number of packets dropped, which determines the amount of synchronization in the backoff behavior.
    - Backoff behavior determines fairness
  - TCP-Illinois has fairness properties similar to those of TCP-Reno

Buffer sizing in core routers

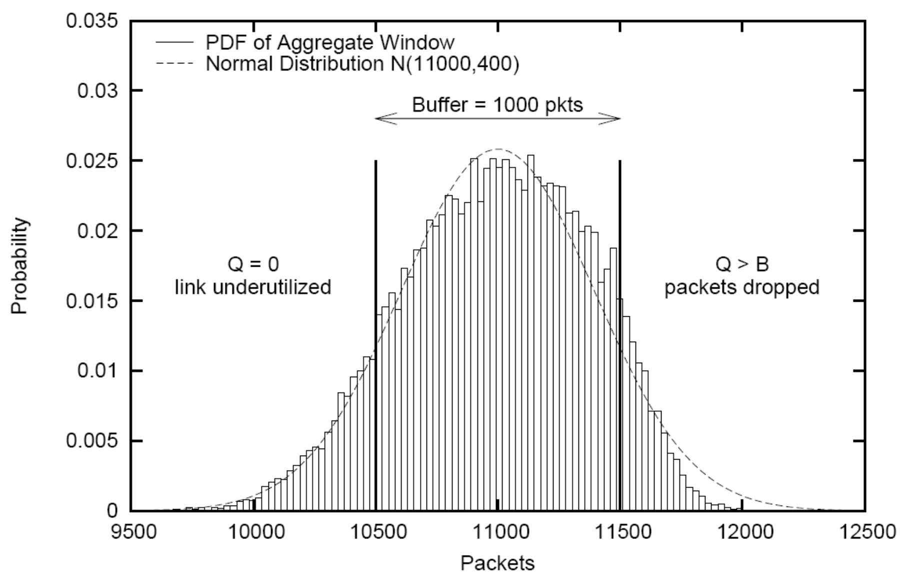
- Assuming TCP-Reno is the protocol for data transfer, how much buffering is needed in the core routers ?
- Model valid for other protocols as well
- Traditional Design goal: 100% link utilization by a single user must be able to achieve 100% throughput.
- Design Rule 1: Buffer Size large enough to feed the queue during timeouts.



$$B = 2 \cdot C \cdot RTT$$

Design rules for buffer sizing: Rule 2 [Appenzeller et al 2004]

- Design goal: Near 100% link utilization.
- Assumption: A large number of flows pass through the router.
- Assumption: Flows are nearly independent of each other.
- Arrival to the core router is nearly Gaussian with variance O(√N).
- Buffers are required to absorb bursts of O(√N).



$$B = \frac{2 \cdot C \cdot RTT}{\sqrt{N}}$$

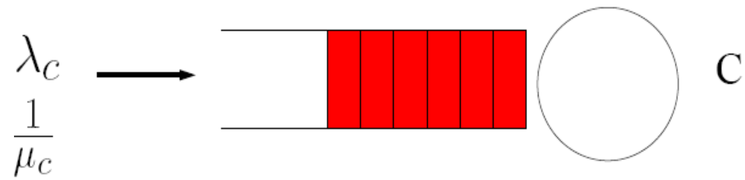
Design rules for buffer sizing: Rule 3 [Enachescu et al, 2006]

- Design goal: High (not 100%) link utilization
- Assumption: A large number of flows (N) access the core router.
- Assumption: Core router is not congested:

$$\frac{NW_{Max}}{RTT} < C$$

- Arrival process to the core router can be approximated by a Poisson process.
- Buffers can be chosen based on an approximate analysis based on an M/M/1/B queuing model.
- Buffer requirement independent of core router capacity.

$$P_{loss} = \frac{\rho_c^B(1-\rho_c)}{(1-\rho_c^{B+1})}$$



$$\rho_c = \frac{\lambda_c}{\mu_c}$$

$$B = O\left(\log(W_{Max})\right)$$

Static versus Dynamic Networks

- Consider router with C=10Gbps, RTT = 250ms
- Number of flows: 10,000
- Results based on a static network with fixed number of users

Utilization	Buffer Required		
	2CRTT	O(√N)	log(Wmax)
95.00%	5Gb	25Mb	1Mb
99.00%	5Gb	25Mb	4Mb
99.90%	5Gb	50Mb	21.6Mb

How should buffers be sized under flow arrivals and departures ?

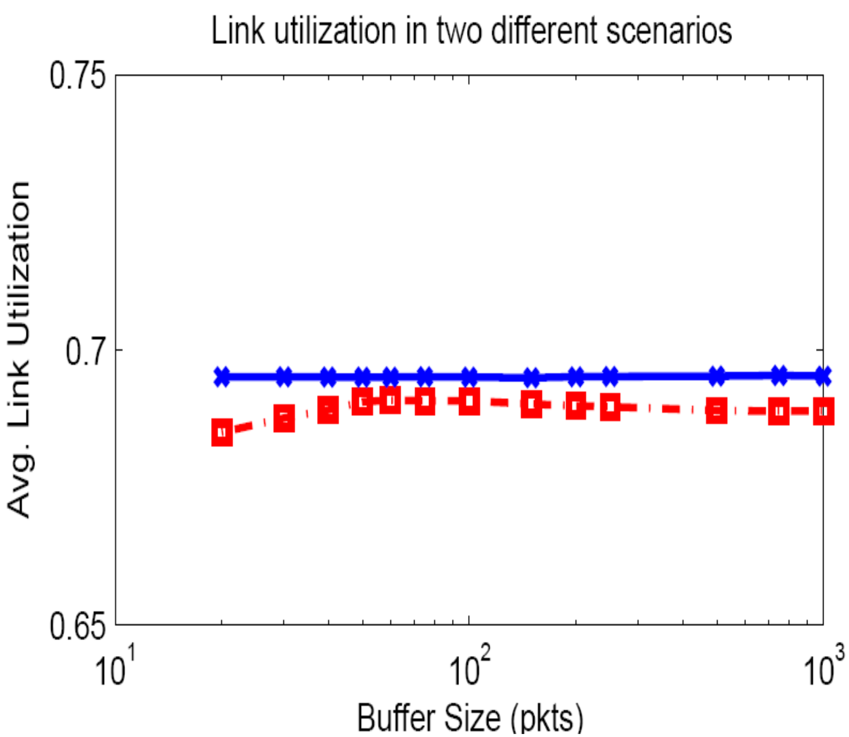
Objectives

- Model arrivals and departures of flows explicitly.
- Develop an unified model that can be applicable to a variety of network conditions.
- Provide design guidelines for buffer sizing to maintain high-end user QoS under different network scenarios.

What is the appropriate metric of end-user QoS ?

End user QoS

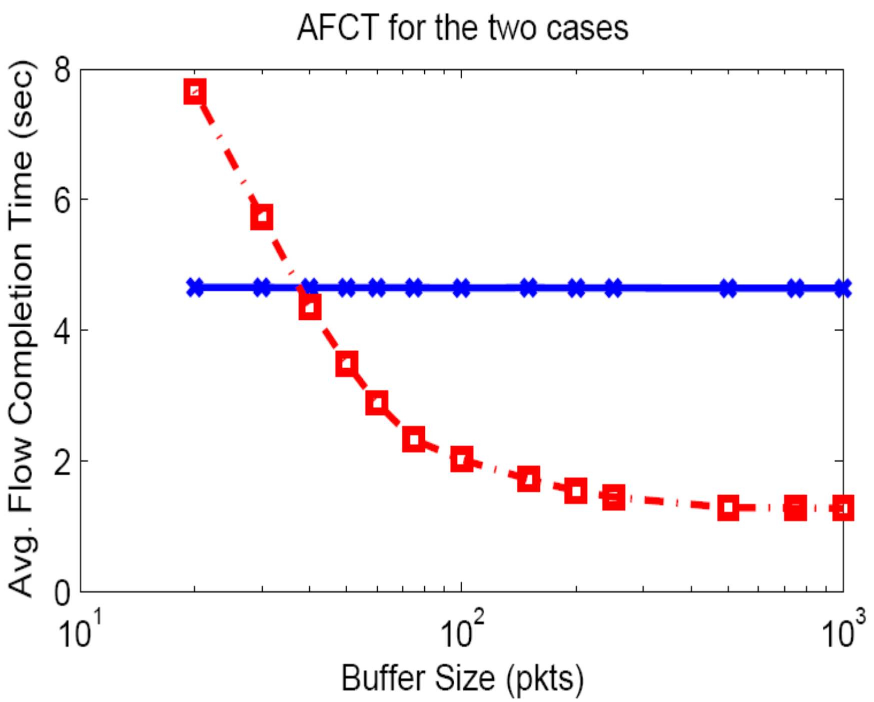
- Assuming stability, with flow arrivals and departures, average link utilization is *always* equal to the offered load, *independent* of the buffer size!
- End users are interested in download times.
- Use average flow completion time (AFCT) as a performance metric



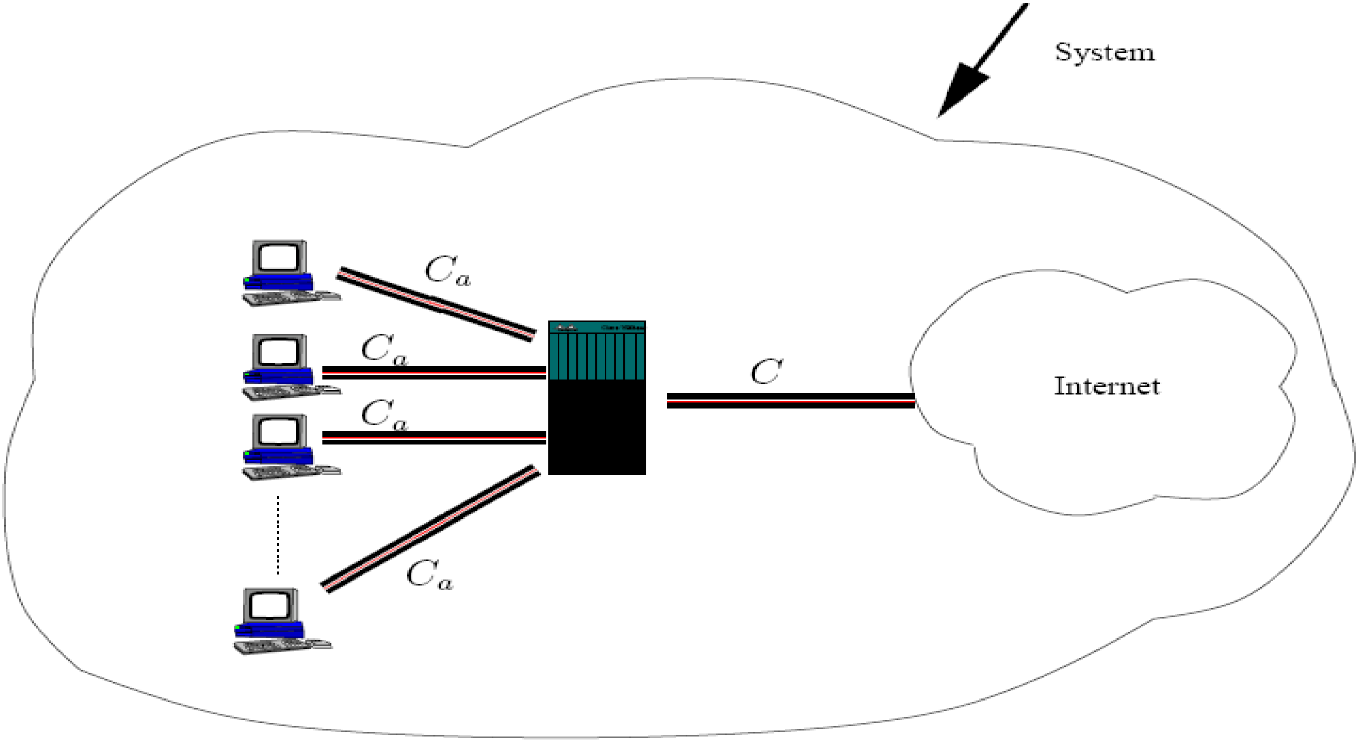
Core to access speed is 3
Core to access speed is 50

End user QoS

- Under flow arrivals and departures, average link utilization is *always* equal to the offered load, *independent* of the buffer size!
- End users are interested in download times.
- Use average flow completion time (AFCT) as a performance metric

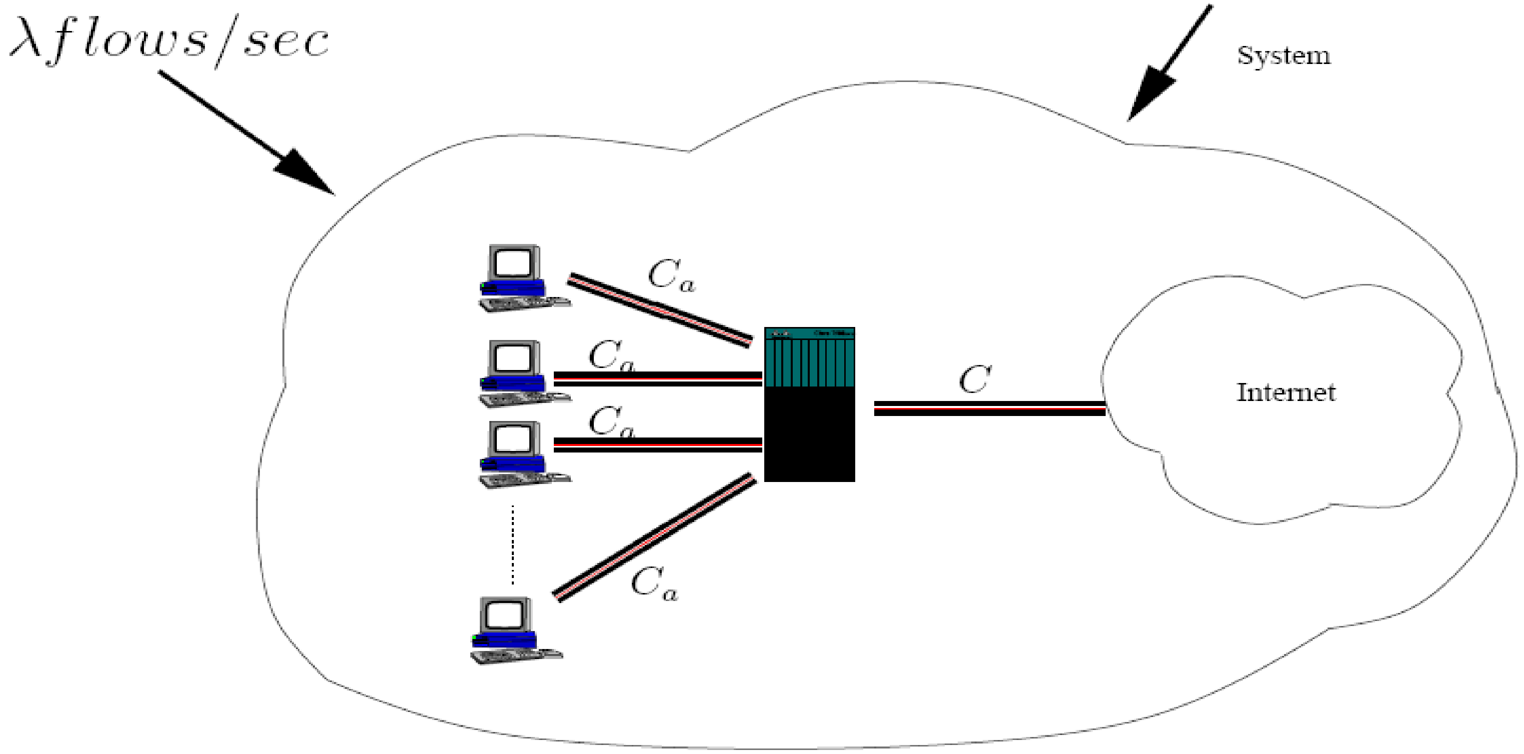


System model



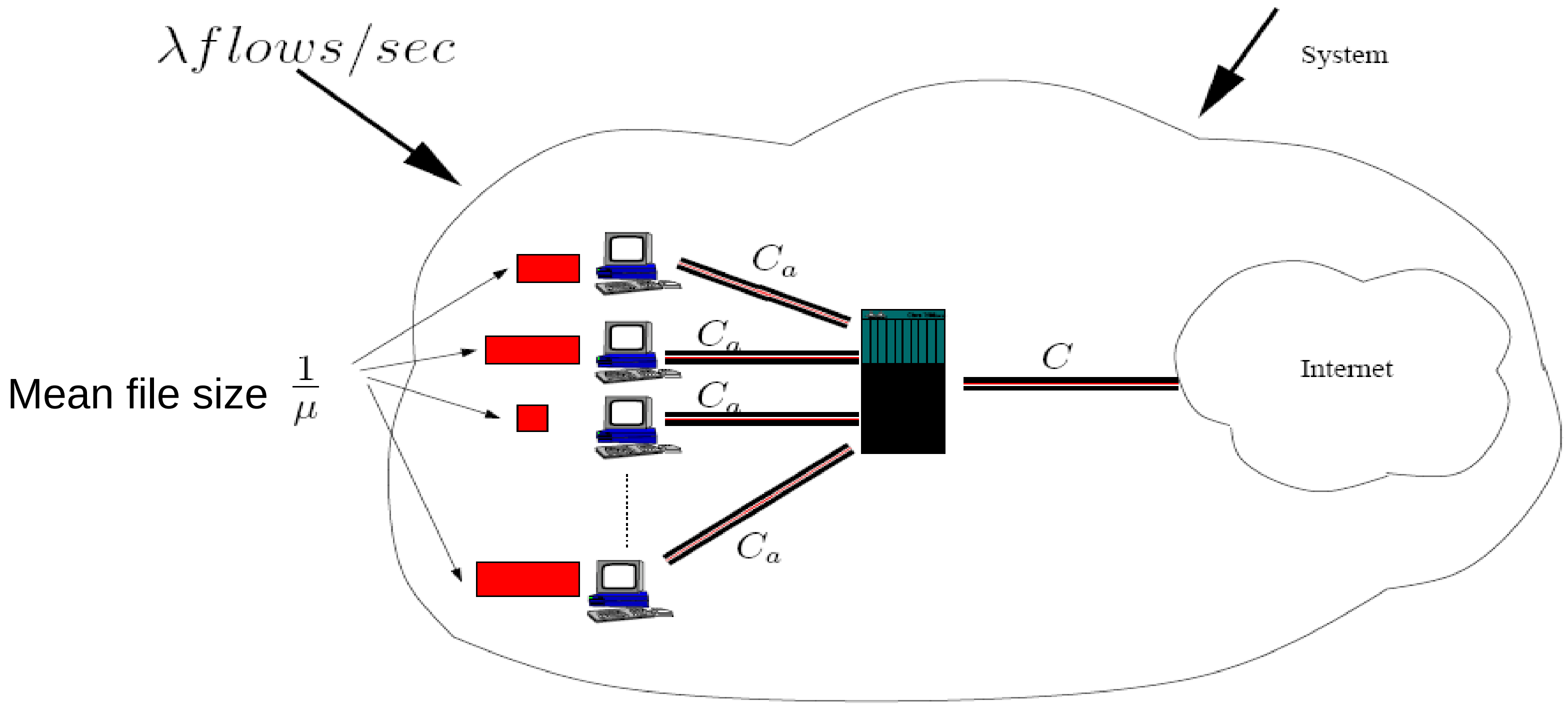
$$K = \frac{C}{C_a}$$

System model



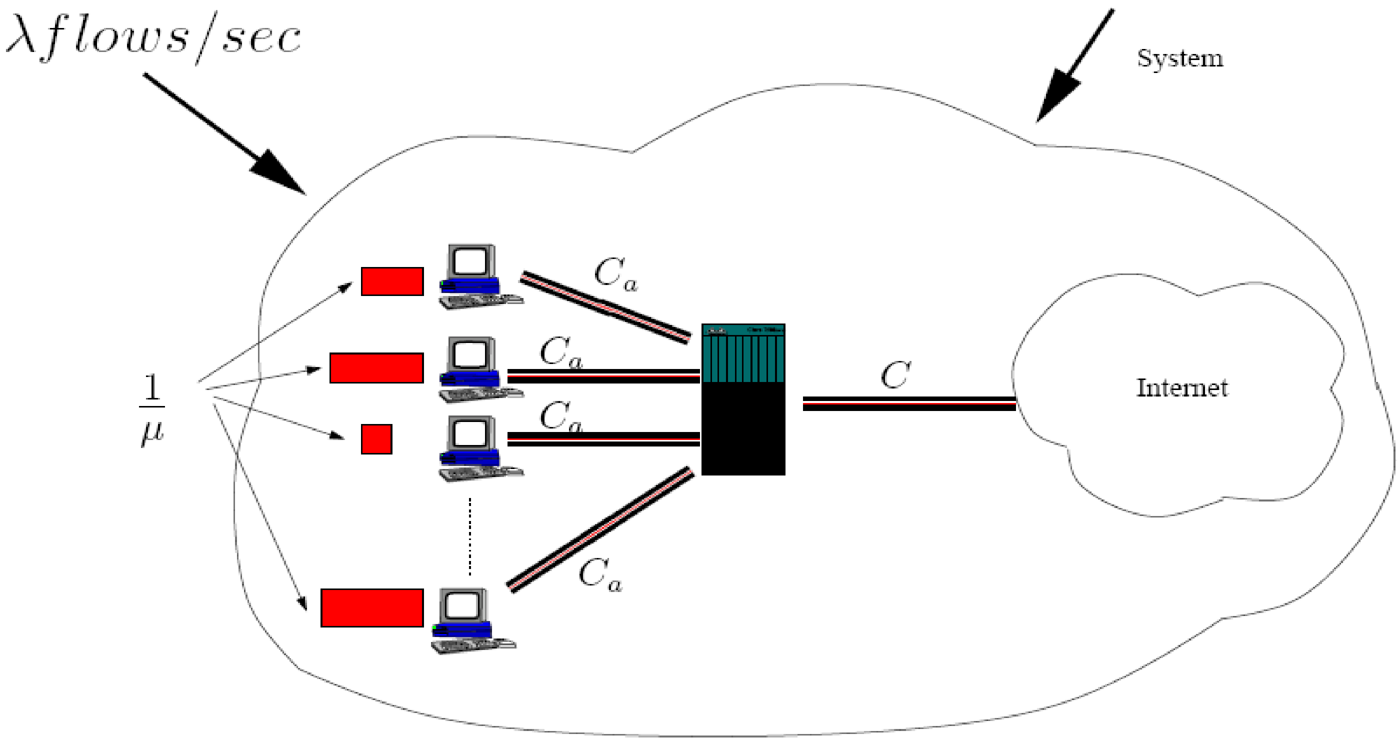
$$K = \frac{C}{C_a}$$

System model



$$K = \frac{C}{C_a}$$

System model



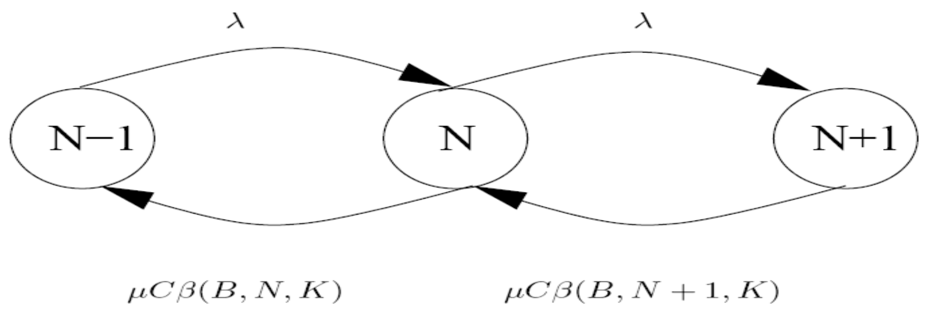
$$K = \frac{C}{C_a}$$

Analysis

- To compute AFCT, we need to know how fast the packets are being drained from the system, i.e., the link utilization (a.k.a. efficiency) of the core router.
- Link utilization (β) depends on
  - Number of users in the system (N)
  - Core router buffer size (B)
  - Core to access speed ratio (K)
- Given β(N,B,K), we can calculate AFCT using a Markov chain analysis.
  - Arrivals are Poisson
  - Service times are exponentially distributed. This assumption is not necessary. The results have an insensitivity property to service-time distributions.

Markov chain model

- The flow level queueing process is a standard birth-death model.
- Steady state distribution can be easily characterized. Obtain E[N] using the steady state distribution.
- Little's law: AFCT = E[N]/λ



- Internet-type networks
  - Large core to access bandwidth ratio. Requires extremely large number of flows to congest the core router (typically thousands).
- Data-center networks
  - All routers have very similar capacities. Single flow can cause congestion on the core router.

Internet-Type Networks

- Obtain β(N,B,K) using an analysis based on N long-lived flows.
- Steady state distribution:

$$Prob\{N = i\} = \pi_i = \frac{\rho^i \prod_{j=0}^{i} \frac{1}{\beta_j}}{\sum_{k=0}^{\infty} \rho^k \prod_{j=0}^{k} \frac{1}{\beta_j}}$$

$$AFCT = \frac{\sum_{i=1}^{\infty} i\pi_i}{\lambda}$$

- Expression for AFCT is difficult to analyse directly.

Internet type networks

- Suppose the access routers are mostly the bottleneck:

$$\sum_{i=0}^{\beta K} \pi_i \approx 1$$

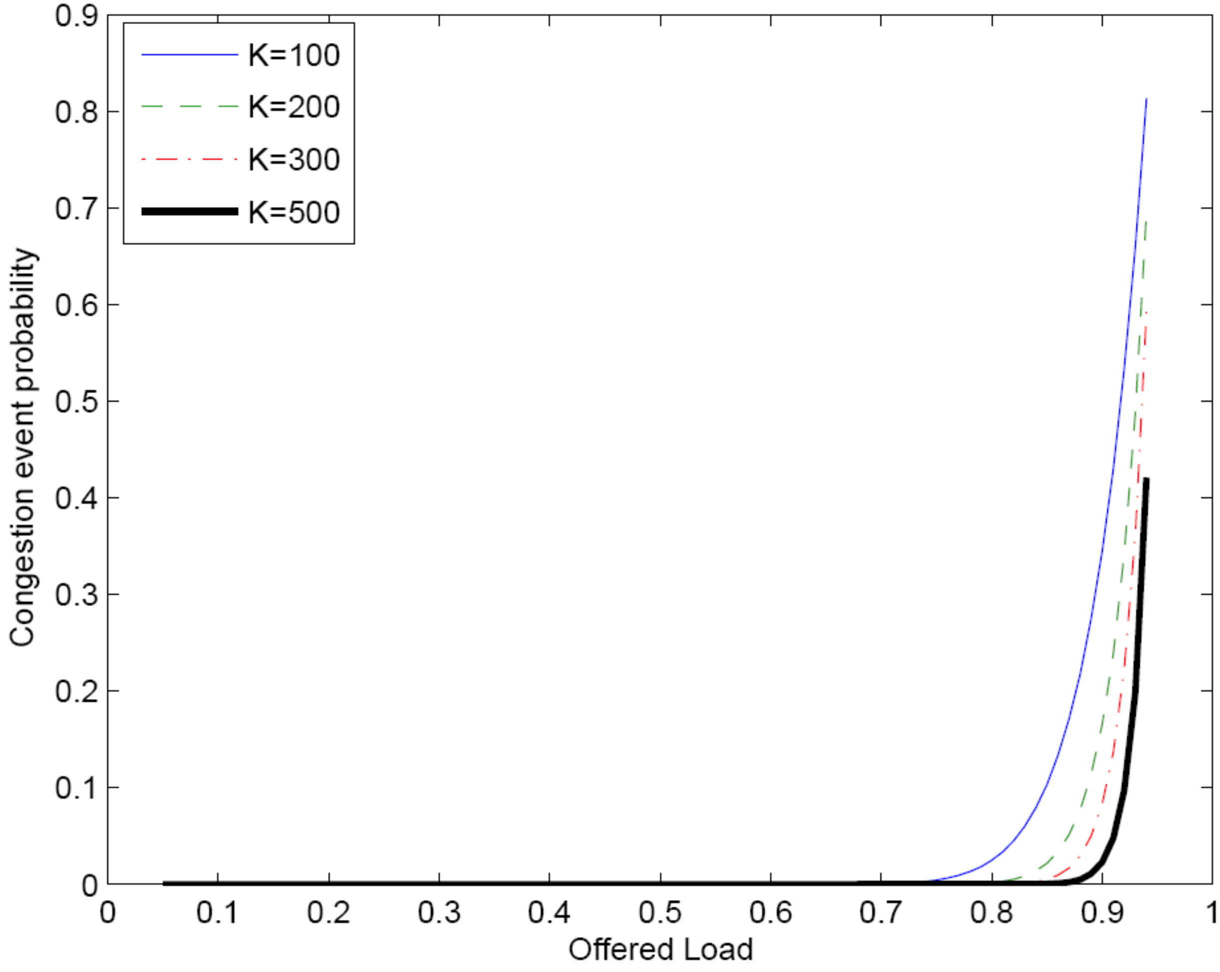
$$AFCT = \frac{1}{\mu C_a}$$

- On systems that are not congested, AFCT is independent of the core router buffer size

  →Very small buffers can be used at the core router!

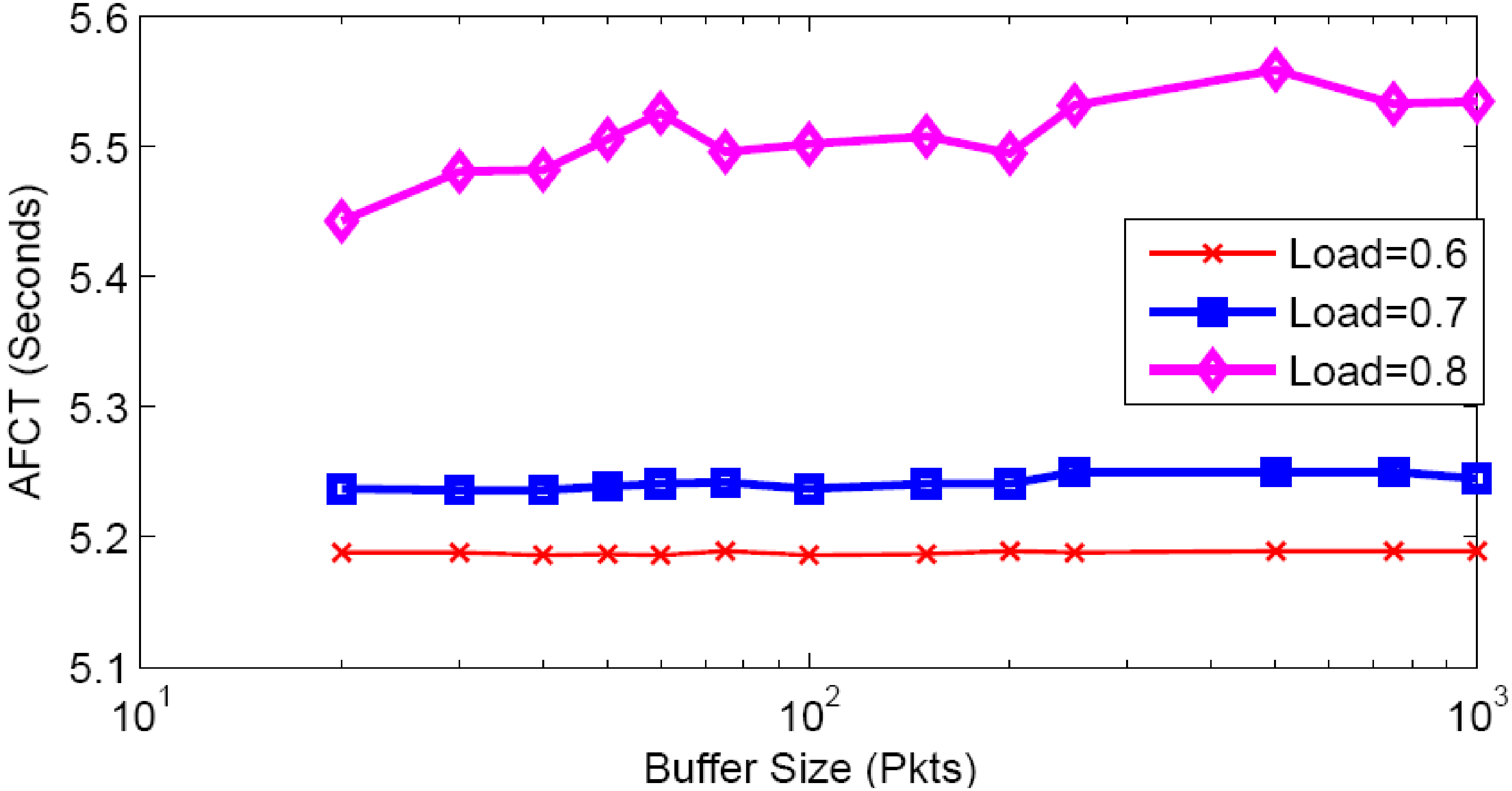
- Congestion on the core router depends on the offered load
  - How small should the offered load be for the core router to remain largely uncongested?

Internet-type networks

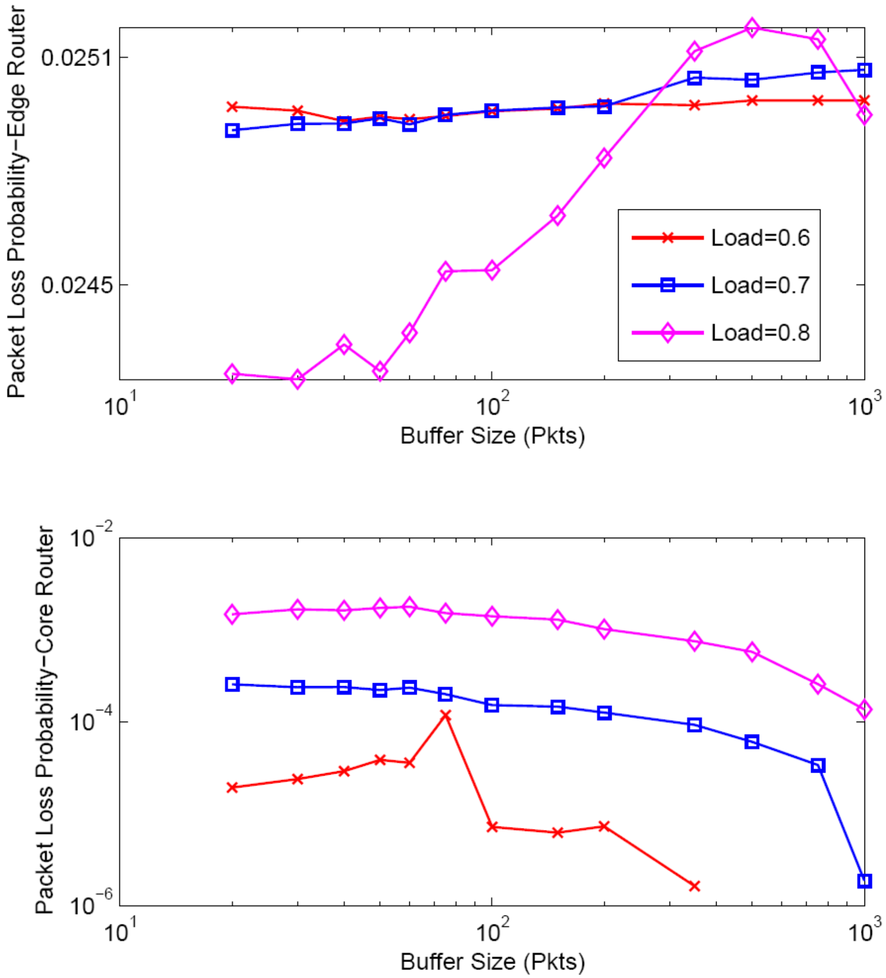


Simulations

- C = 100Mbps; C_a = 2Mbps; mean RTT = 50ms
- Packet size = 1KB; Mean flow size = 1.1MB
- Flow size distribution: Bounded Pareto; C*RTT = 625KB = 625 Packets



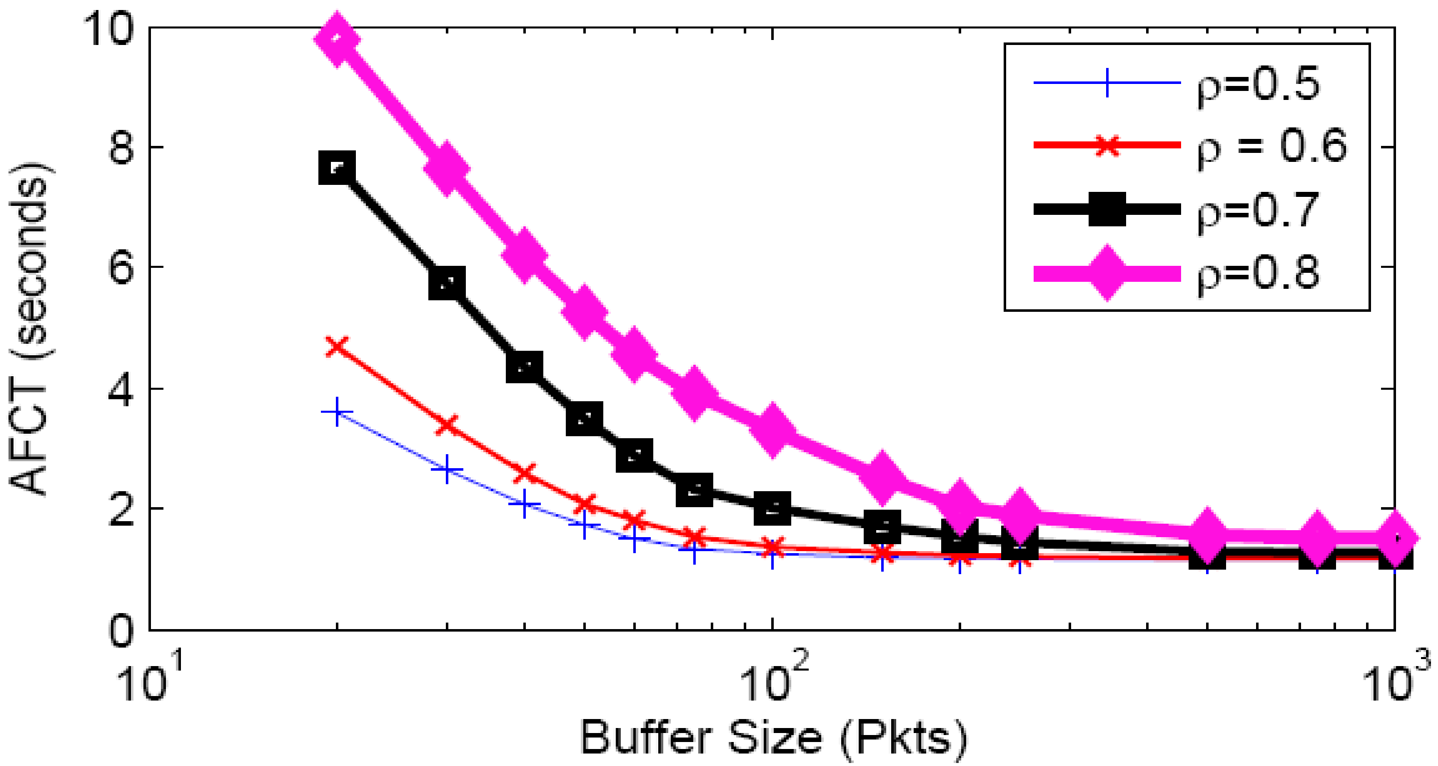
Loss probability



Data-center networks

- Obtain β(N,B,K) using a static analysis based on N long-lived flows.
- Use Little's law to obtain an expression for AFCT.
- Very few flows can congest the core router (K < 10).
  - Core router is highly congested even at mild loads.
  - Small buffers degrade performance significantly
- Simulation Parameters
  - C = 100Mbps
  - C_a = 30Mbps
  - mean RTT = 50ms
  - Packet size = 1KB
  - Mean flow size = 1.1MB
  - Flow size distribution: Bounded Pareto
  - C•RTT = 625KB = 625 Packets

Data-Center Networks



Which rule should we follow ?

- C = 10Gbps , RTT = 250ms
- When K < 10, core router always in congestion
  - We need 2CRTT amount of buffering!

Utilization	Buffer Required		
	2CRTT	O(√N)	log(Wmax)
95.00%	5Gb	25Mb	1Mb
99.00%	5Gb	25Mb	4Mb
99.90%	5Gb	50Mb	21.6Mb

Which rule should we follow ?

- C = 10Gbps , RTT = 250ms
- When K =10000, core router is rarely in congestion.
  - Buffers of size O(log(C_a RTT)) is sufficient!

Utilization	Buffer Required		
	2CRTT	O(√N)	log(Wmax)
95.00%	5Gb	25Mb	1Mb
99.00%	5Gb	25Mb	4Mb
99.90%	5Gb	50Mb	21.6Mb

Summary: Buffer Sizing

- Unified model to provide buffer sizing rules
- Model for file arrivals and departures
  - Able to capture the buffer sizing rule as a function of the ratio of core router speed to access speed
  - Hard to capture the above dependence using static models
- Internet type networks
  - Very little congestion on the core routers even at high loads
  - Very small buffers can be used
- Data-center networks
  - Routers are congested very often
  - Large buffers are needed to ensure very small AFCT

Conclusions

- Time-scale of interest determines the right modeling choice
  - End-user congestion control design
  - Router buffer design
- Detailed congestion-event models versus fluid models
  - Former useful to study packet loss, synchronization and impact on fairness and stability
  - Later useful for large-network analysis
- Static versus dynamic models of buffer sizing
  - Use static models to understand link utilization for a fixed number of flows
  - Incorporate efficiency formula in a dynamic model to understand the QoS measure of interest, namely, AFCT
  - Different conclusions in dynamic networks