

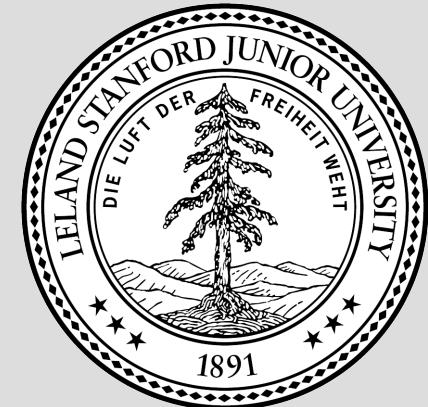
CS244

Advanced Topics in Networking

Lecture 10: Buffer Sizing

Nick McKeown

“Sizing Router Buffers”
[Appenzeller, et al. 2004]



Context

Guido Appenzeller

- At the time: CS PhD student
- Founded Big Switch Networks
- CTO at VMware for networking
- Sigcomm Test of Time Award, 2015



At the time of writing

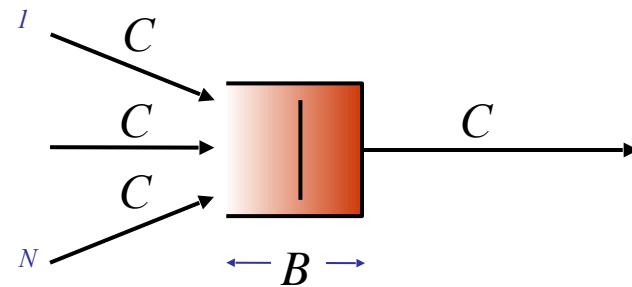
- Challenging to build ISP routers with big buffers
- 80% of world's SRAMs used for router and switch buffers/counters
- ISP routers sold with ~90% profit margin

Why should we care?

Background

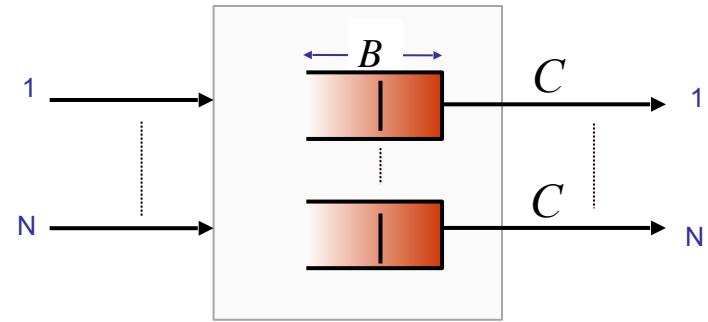
- No universal agreement on how big a router buffer should be
- Or why.
- Yet buffers are a major cause of variation in packet delay.
- Big buffers require large, slow DRAM memories...
- ...which complicate the design of large routers.
- It would be nice if we could use single chip switches and routers

Simple model of FCFS router buffer

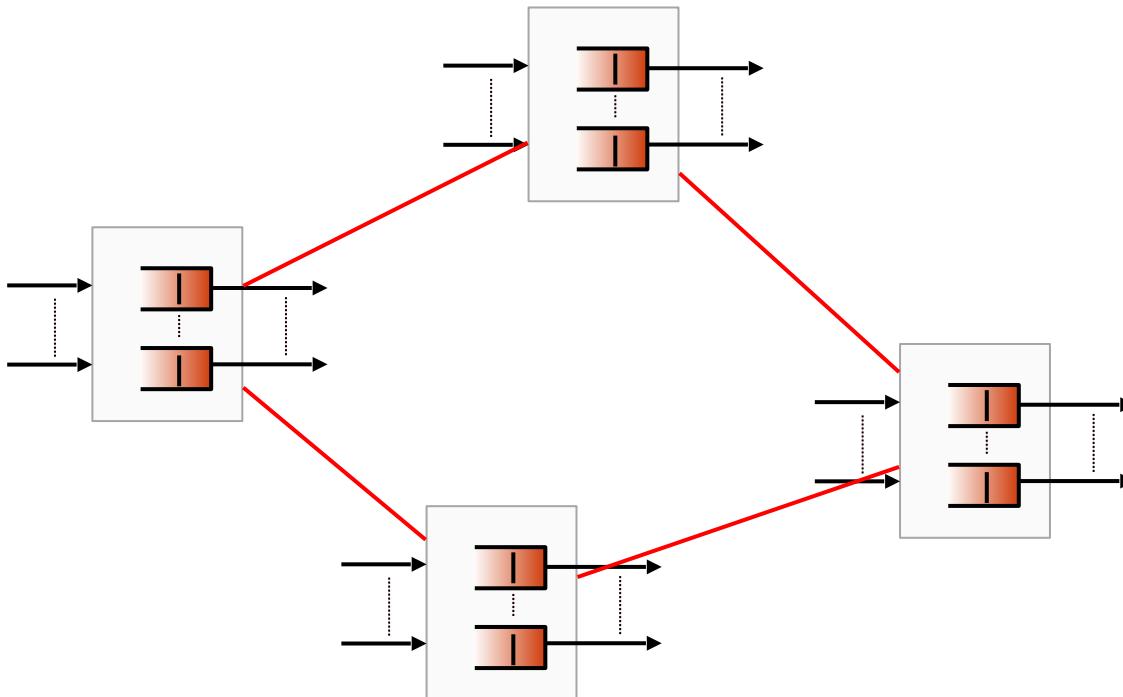


Q: Why does the router have a buffer (or queue)?
Q: What factors determine the buffer's size?

Simple model of a router



Simple Internet queueing model



Early Internet Models

Leonard Kleinrock
Professor at UCLA

1963: 1st theoretical study of packet switching using queueing theory

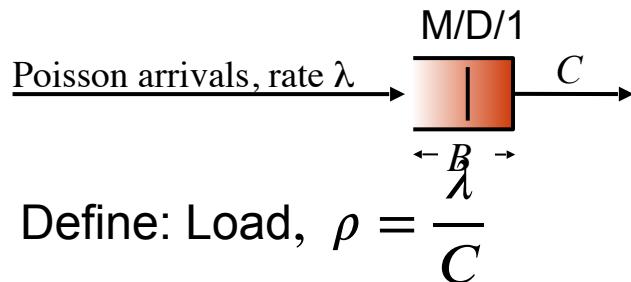


1969: 1st message sent over Internet.

UCLA IMP (Interface Message Processor)

Example model of single packet queue

Poisson Traffic, fixed size packets



Then: drop rate $< \rho^B$
e.g. $\rho = 0.8$, $B = 20\text{pkts}$ \rightarrow loss < 1%

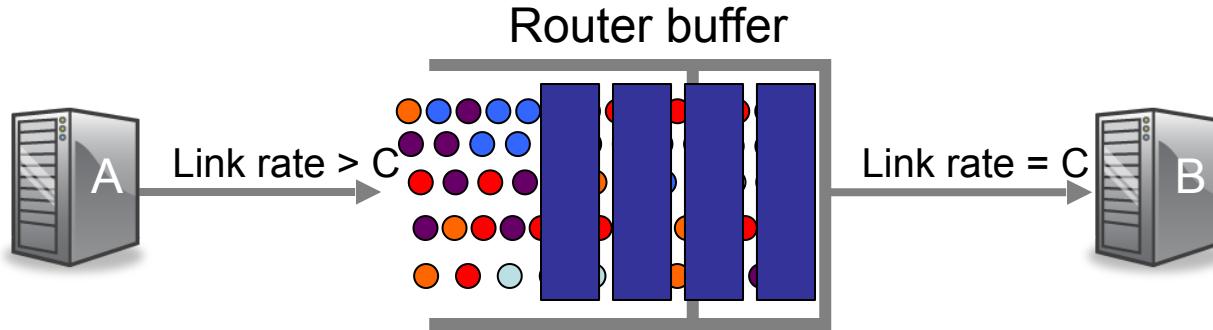
Observations:

- Packet drop rate is small
- Independent of C, RTT, number of flows, etc.

Q: How well does this model fit today's Internet?

The model assumes traffic is generated “open loop” by the source. Today’s Internet carries mostly TCP traffic, which uses a closed loop congestion control algorithm.

With multiple flows, RTT is less variable



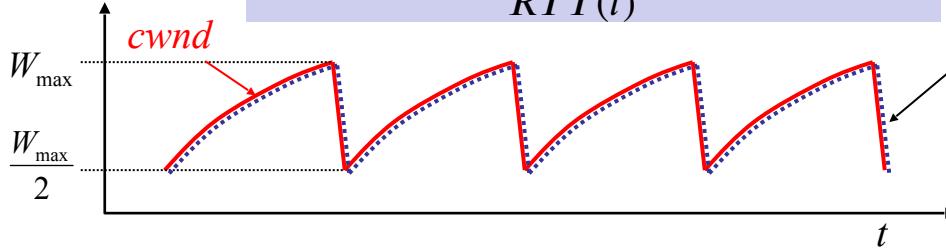
Observations:

- An arriving packet sees a usually-full queue
- RTT is pretty much constant (very different from single flow case)
- Drops are random events

One flow vs multiple flows

One flow

$$\text{Throughput} = \frac{W(t)}{RTT(t)} = \text{constant } 100\%$$



Buffer occupancy
and RTT

Multiple flows

$$\text{Throughput} = \frac{W(t)}{RTT} \propto W(t)$$

Buffer
Occupancy
and RTT

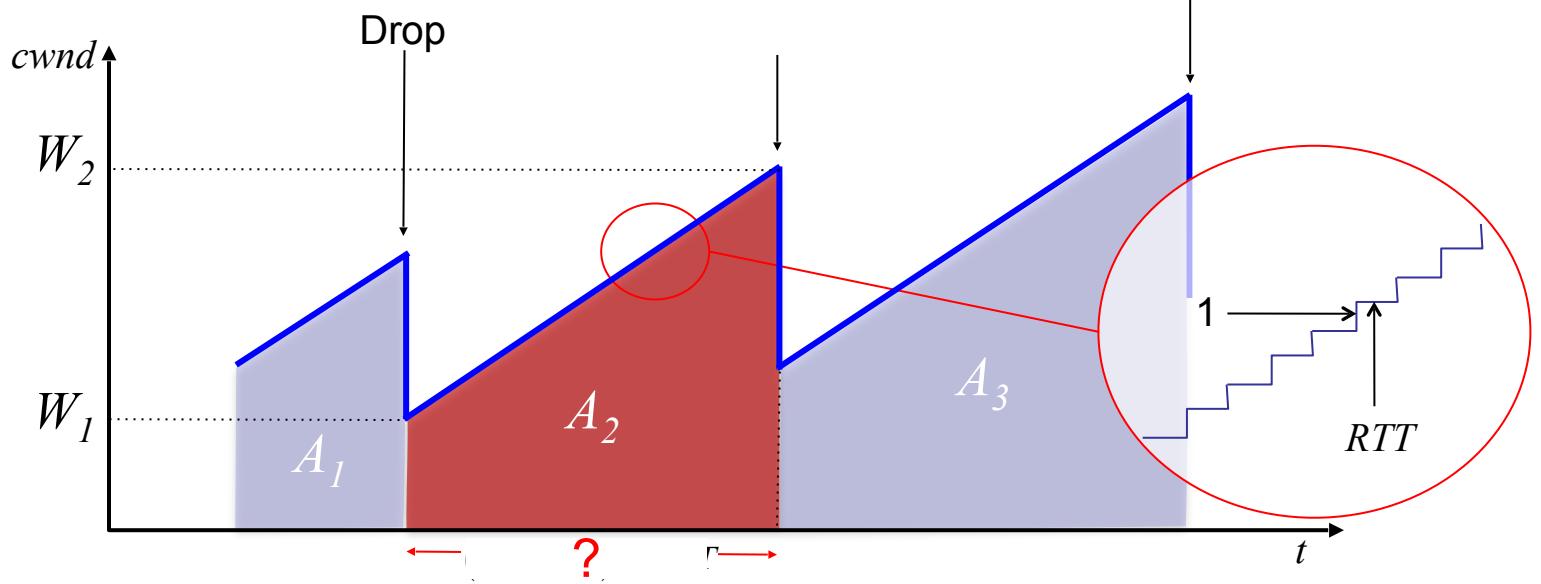
Buffer occupancy
and RTT

(Zoom in)
 $cwnd$

One of the flows

t

Geometric intuition for throughput equation



Expected Area, $E[A]$

$$\text{Area } A_2 = (W_2 - W_1)W_1 + \frac{1}{2}(W_2 - W_1)^2$$

$$\text{But } E[W_1] = \frac{1}{2}E[W_2]$$

Throughput, $E[T]$

$$\text{Segments sent during } A_2: T_2 = \frac{A_2}{(W_2 - W_1) \cdot RTT}$$

$$= \frac{\frac{1}{2}(W_2 - W_1)^2}{(W_2 - W_1) \cdot RTT}$$

Throughput: $E[T] = \frac{1}{4 \cdot RTT} E[W]$

Combining

$$\text{Drop rate, } p \approx \frac{1}{E[A]} = \frac{1 - \frac{1}{E[W]}}{\frac{3}{2} E[W]^2}$$

Combining...

$$E[T] = \frac{k}{\sqrt{p} \cdot RTT} \text{ segments/sec}$$

Note: To convert from packets/second to bits/second, multiply throughput by the packet size (e.g. MSS)

Interpreting the throughput equation

$$E[T] = \frac{k}{\sqrt{p} \cdot RTT} \text{ segments/sec}$$

- If RTT doubles, throughput halves.
- If packet drop rate increases from 1% to 4%, throughput halves.

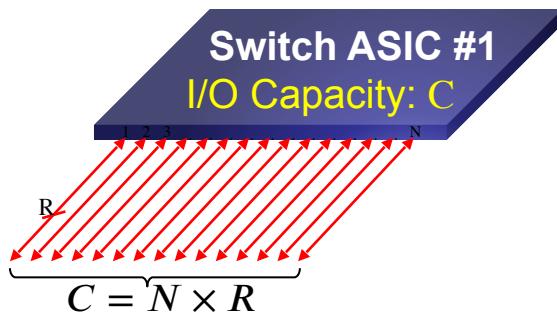
One reason to care about buffer size

With on-chip buffers we can build
higher capacity switch ASICs

Switch Chips are Limited by Serial I/O Capacity to the outside world

Single chip switch ASIC

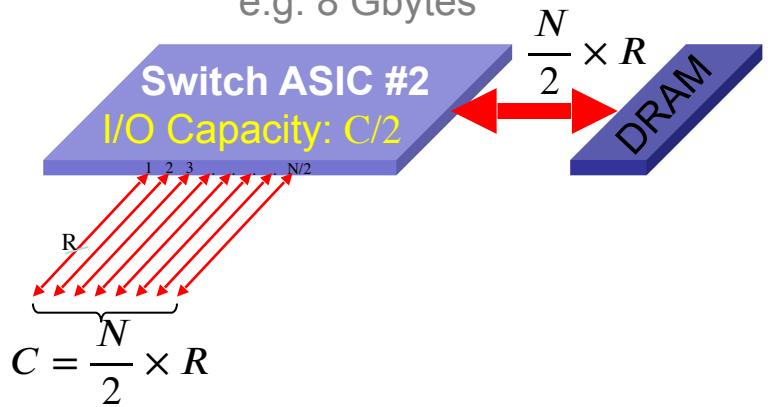
Small on-chip buffering
e.g. 64 Mbytes



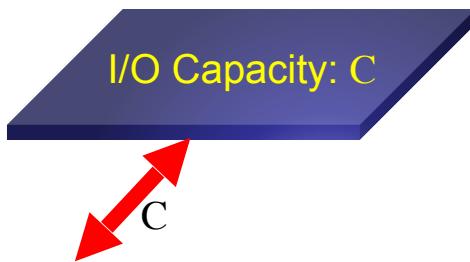
e.g. $12.8 \text{ Tb/s} = 128 \times 100\text{Gb/s}$

Switch ASIC with external memory

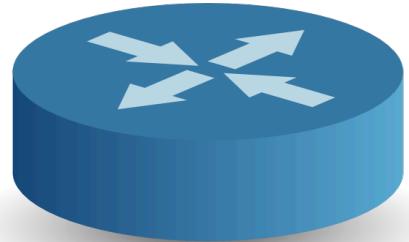
Large off-chip buffering
e.g. 8 Gbytes



Switch Chips are Limited by Serial I/O Capacity

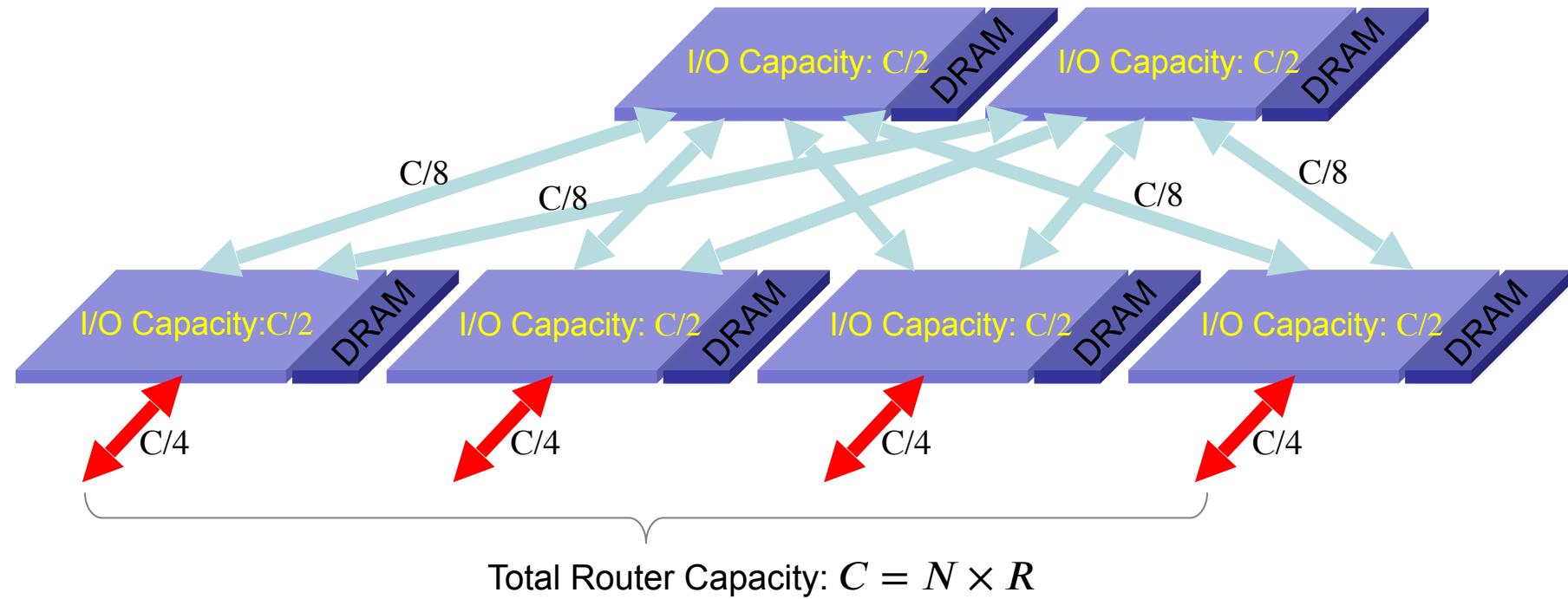


C



How many switch chips
with capacity $C/2$ do we
need to make a router
with capacity C ?

We need 6 ASICs with capacity $C/2$



It is worth understanding
where and when
small on-chip buffers suffice

A brief history of buffer size



Congestion Avoidance and Control*

Van Jacobson[†]
Lawrence Berkeley Laboratory
Michael J. Karels[‡]
University of California at Berkeley
November, 1983

partner networks have experienced an explosive growth over the past few years and with growth come never ceasing problems. For example, it is now common to see set gateway drop 10% of the incoming packets because of local buffer overflows. Investigation of some of these problems has shown that much of the cause lies in protocol implementations (not in the protocols themselves). The "obvious" way becomes a window-based transport protocol can result in exactly the wrong behavior when applied to network protocols. We give examples of "wrong" behavior and discuss the simple changes that can be made to prevent right things from happening. We also discuss the idea of achieving network transparency by using the transport characteristics to obey the "queue discipline" principle. We show how the algorithms derive from this principle effect. <http://www.cs.vt.edu/~mervin/ftp/rtgdocs/rtgdocs.htm>

In October of '86, the Internet had the first of what became a series of 'congestion collapses'. During this period, the data throughput from LBL to UC Berkeley (sites separated by 10 yards and two IMP hops) dropped from 32 Kbps to 40 bps. We were fascinated by this sudden factor-of-thousand drop in bandwidth and embarked on an investigation of why this was happening so bad. In particular, we wondered if the 4.3BSD (Berkeley Unix) TCP was behaving so bad if it could be tuned to work better under heavier network conditions.

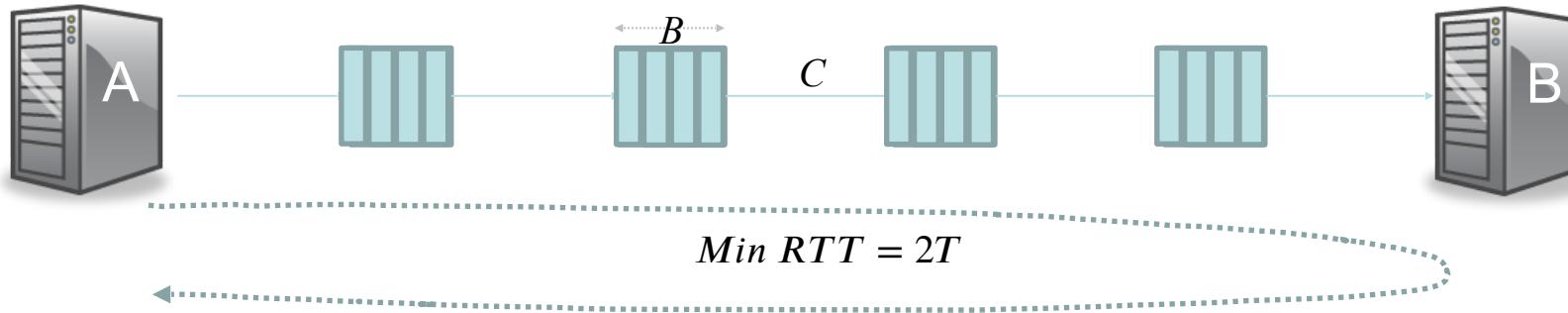
Note: This is a very slightly revised version of a paper originally presented at SIGCOMM '88 [12]. If you have referred to this work, please cite [12].
This work was supported in part by the U.S. Department of Energy under Contract Number DE-AC03-89ER20098.
His work was supported by the U.S. Department of Commerce, National Bureau of Standards, under contract NBSIR 85-100.

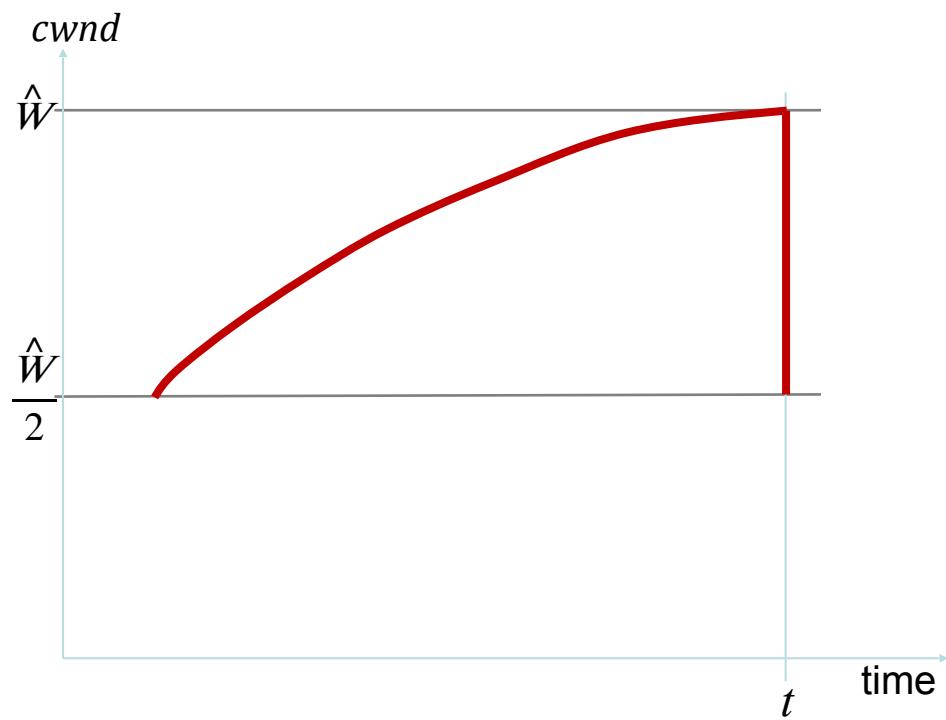
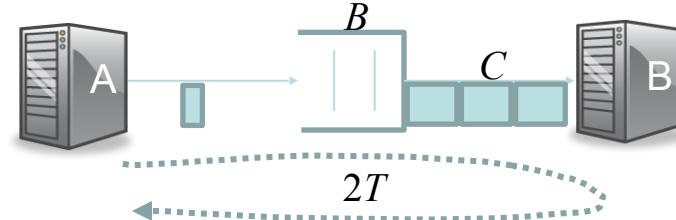
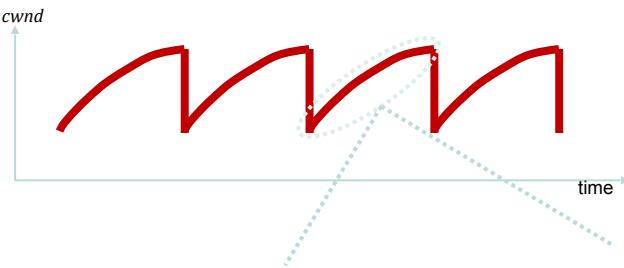
High Performance TCP in ANSNET

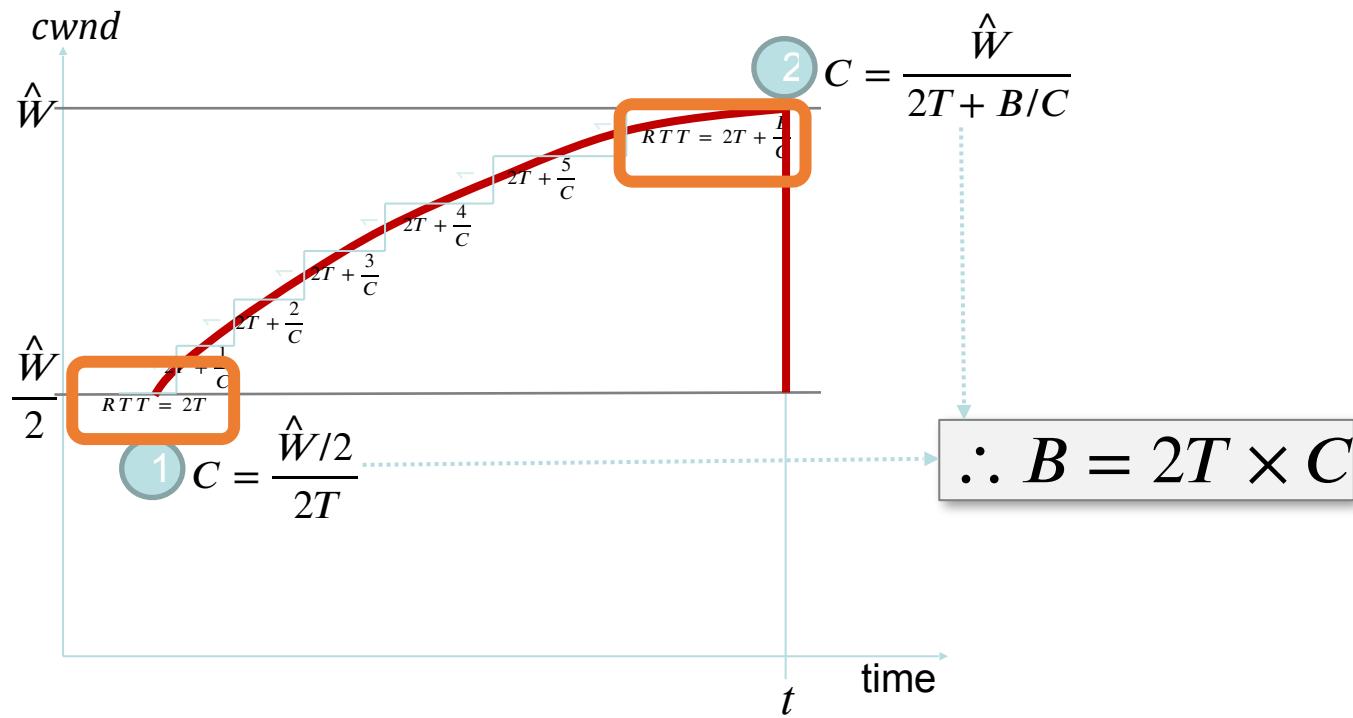
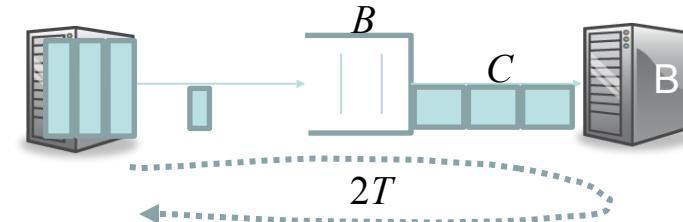
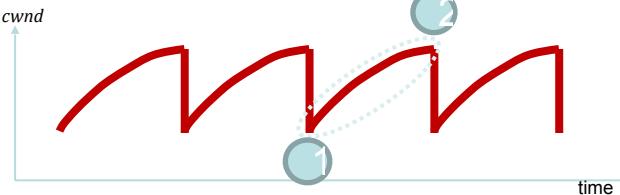
$$B = 2T \times C$$

“Buffer size should equal the bandwidth delay product”

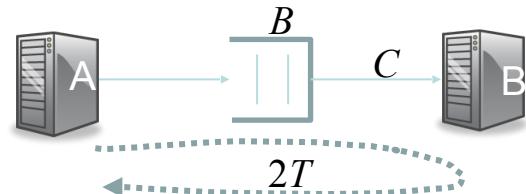
$$\text{Max RTT} = 2T + B/C = 4T$$



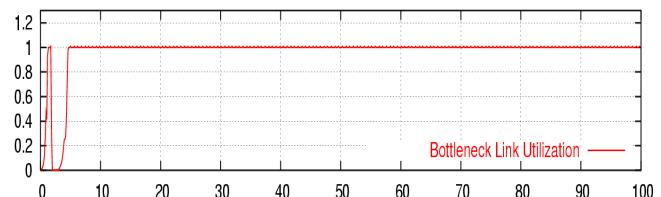
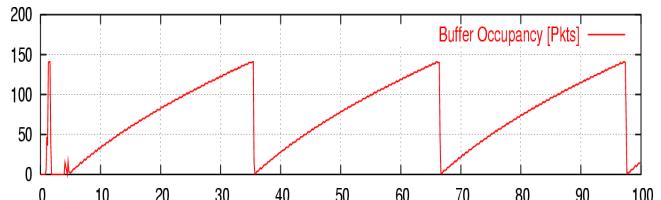
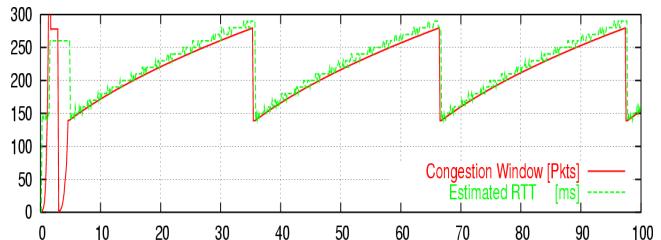




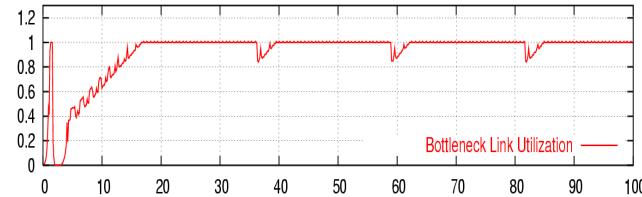
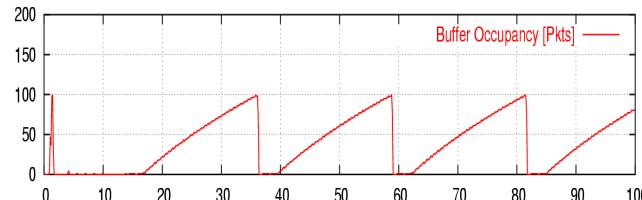
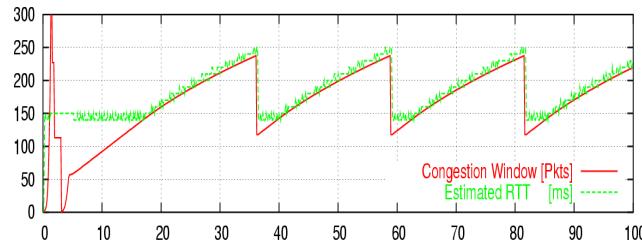
Time Evolution of a Single TCP Flow



$$B = 2T \times C$$



$$B < 2T \times C$$



Single TCP New Reno flow: 100% Throughput

1. If $\hat{W} \rightarrow \frac{\hat{W}}{2}$ then $B \geq 2T \times C$

Example:
 $2T = 100ms, C = 10Gb/s$
 $B \geq 1Gbit$

2. If $\hat{W} \rightarrow \frac{\hat{W}}{k}$ then $B \geq 2T(k - 1) \times C$

Example: $k = 1.5$
 $B \geq 500Mbits$

Example: $k = 1.14$
 $B \geq 140Mbits$

3. If $k = 1 + \frac{a}{2T}$ then $B \geq aC$

Example: $a = \frac{1}{100}$
 $B \geq 50Mbits$

i.e. if end host knows $2T$, buffer size is independent of RTT



Congestion Avoidance and Control

Van Jacobson[†]
Lawrence Berkeley Laboratory
Michael J. Karels[‡]
University of California at Berkeley
November, 1983

High Performance TCP in ANSNET

Sizing Router Buffers

Introduction

Computer networks have experienced an explosive growth over the past few years and with that growth have come several congestion problems. For example, it is now common to see internet gateways drop 10% of the incoming packets because of local buffer overflow. This is a problem that can be solved by using a more efficient and reliable transport protocol implementation (such as the protocols themselves). The *diverse* ways to implement a window-based transport protocol can result in exactly the wrong behavior in response to network congestion. We give examples of "wacky" behavior and describe some simple algorithms that can be used to make right things happen. The algorithms presented here are based on the principle of "congestion control" and not the "queue discipline" principle. We show how the algorithms derive from this principle and what effect they have on traffic over congested networks.

In October of '96, the Internet had its first became a series of "congestion col-

¹ This work was supported by the U.S. Department of Commerce, National Bureau of Standards, under Grant Number NBRAND00001.

High Performance TCP in ANSNET
Dmitri Vlasov dvlasov@ans.net
Advanced Networks & Services, Inc.
Cong Sang congsang@ans.net
Adventis
September 12, 1991

Abstract

The report considers as specific requirements and problems of the TCP performance in the ANSNET. The TCP performance in the ANSNET is analyzed. The TCP options like TOS demand flag and WAT are considered. The influence of the TOS demand flag on the TCP performance is analyzed. The reduced value of the maximum window has been found to be the main reason of the low TCP performance in the ANSNET. The influence of the TOS demand flag on the TCP performance is analyzed.

1 TCP Native Box

1.1 TCP Options Analysis

1.1.1 Maximum Window Size

1.1.2 Priority Precedence

1.1.3 Performance Metrics

2 Queues Requirements

2.1 Queuing Discipline

2.1.1 Effect of Queueing Capacity

3 Performance Testing

3.1 Testbed Configuration

3.1.1 Building Testbed Capacity

3.1.2 Configuration of Test Facilities

3.2 Test Results

3.2.1 High Speed Network

3.2.2 Low Speed Network

3.2.3 Broadband

3.2.4 ATM

3.2.5 Frame Relay

3.2.6 ISDN

4 Recommendations

4.1 Other Considerations

4.1.1 Congestion Control

4.1.2 Acknowledgment

5 Introduction

Sizing Router Buffers

Guido Appenzeller
Stanford University
gappenz@cs.stanford.edu

Ivan Keskaris
Stanford University
keskaris@cs.stanford.edu

Nick McKeown
Stanford University
nickm@cs.stanford.edu

ABSTRACT
In this paper we consider certain tradeoffs in the design of computer networks. One of the chief difficulties in designing a network is determining the size of buffers required at each node. In this paper we argue that the size of the link buffer is a key parameter which affects both the performance and reliability of the network. We show that such links can affect the performance of the network even if they are not the bottleneck of the link. For example, a 10Mbps router located next to a 1Gbps link will experience significant performance degradation due to the saturation of buffering memory by the incoming traffic. We also show that the performance of a network can be significantly improved by increasing the size of the link buffer. We further find that, for high volumes, the performance of a network can be significantly improved by increasing the size of the link buffer. We also find that the end-to-end delay of a 10Mbps link is significantly reduced when the link buffer is increased from 10ms to 100ms. We conclude that the size of the link buffer is a key parameter which must be considered in the design of a network.

Categories and Subject Descriptions

C.2 [Internetworking]: C.2.1 [Network Design and Evaluation]; C.2.2 [Network Protocols]; C.2.3 [Network Performance]

General Terms
Algorithm, Architecture, Design, Experimentation, Theory

Keywords
Computer network, buffer size, bandwidth, delay budget, TCP, IP, ATM, QoS

1. INTRODUCTION AND MOTIVATION

Background
Computer networks have become increasingly popular over the last decade. One of the major reasons for this popularity is the high degree of freedom that they offer. They allow us to connect multiple nodes in a local area, and when these nodes are connected to a single backbone, we can easily expand the network to include more nodes. In addition, they provide a wide range of services, such as email, file sharing, and video conferencing. However, there are many challenges associated with the design and implementation of computer networks. One of the most important challenges is determining the size of the link buffer. This is because the link buffer is a key component in the design of a network, and its size can significantly affect the performance and reliability of the network. For example, a link buffer that is too small may result in packet loss, while a link buffer that is too large may result in unnecessary latency. Therefore, it is important to carefully consider the size of the link buffer when designing a network.

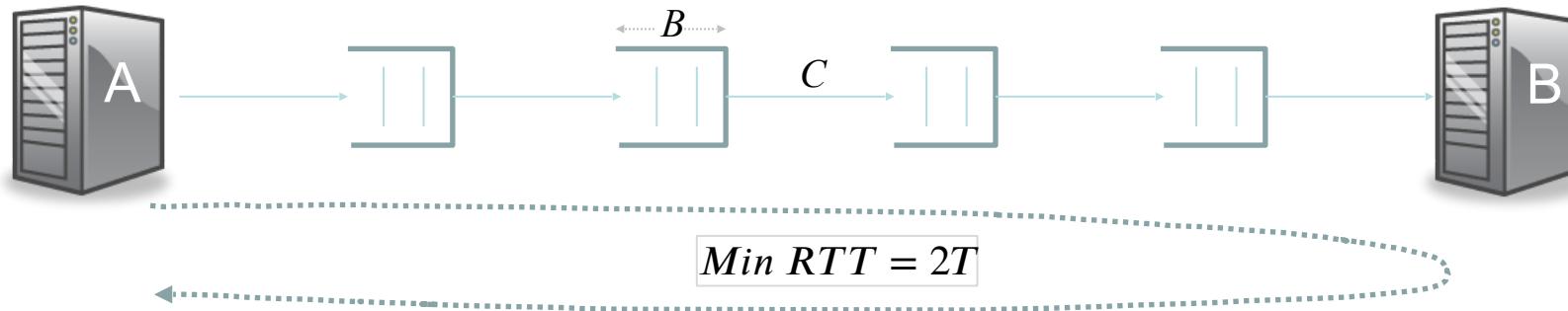
Problem Statement
The problem of determining the size of the link buffer is a difficult one. There are many factors that must be taken into account, such as the type of traffic, the number of nodes, and the link speed. In addition, the link buffer must be able to handle both normal traffic and abnormal traffic, such as link failures and node failures. Therefore, it is important to carefully consider the size of the link buffer when designing a network.

Our Contribution
In this paper, we propose a new approach for determining the size of the link buffer. Our approach is based on the idea that the link buffer should be sized to accommodate the maximum amount of traffic that can be generated by the nodes connected to the link. We also propose a method for calculating the maximum amount of traffic that can be generated by the nodes connected to the link. This method takes into account the link speed, the number of nodes, and the type of traffic. We also propose a method for calculating the maximum amount of traffic that can be generated by the nodes connected to the link. This method takes into account the link speed, the number of nodes, and the type of traffic.

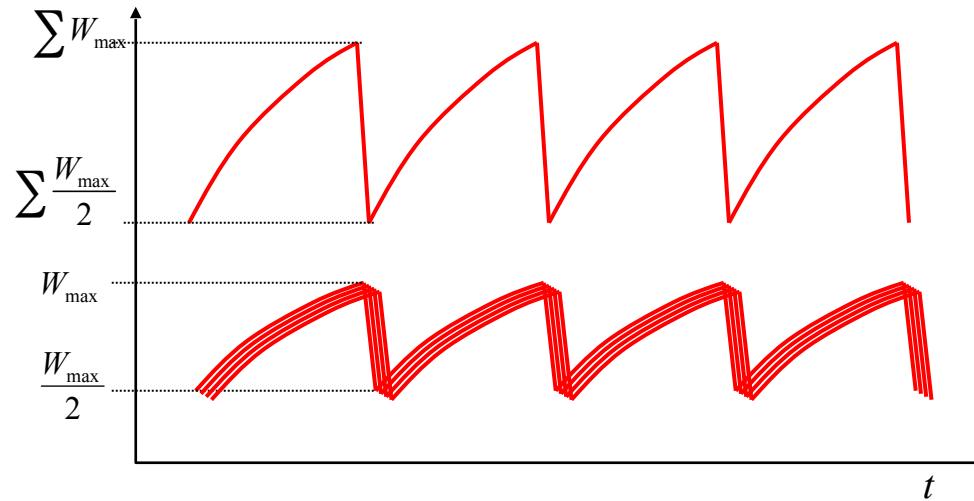
Organization
The rest of this paper is organized as follows. In Section 2, we introduce the basic concepts of computer networks and the link buffer. In Section 3, we discuss the performance of a network with different link buffer sizes. In Section 4, we propose a new approach for determining the size of the link buffer. In Section 5, we evaluate our approach using simulation results. Finally, in Section 6, we conclude the paper.

$$B \geq \frac{2T \times C}{\sqrt{N}}$$

where N is the number of long-lived flows

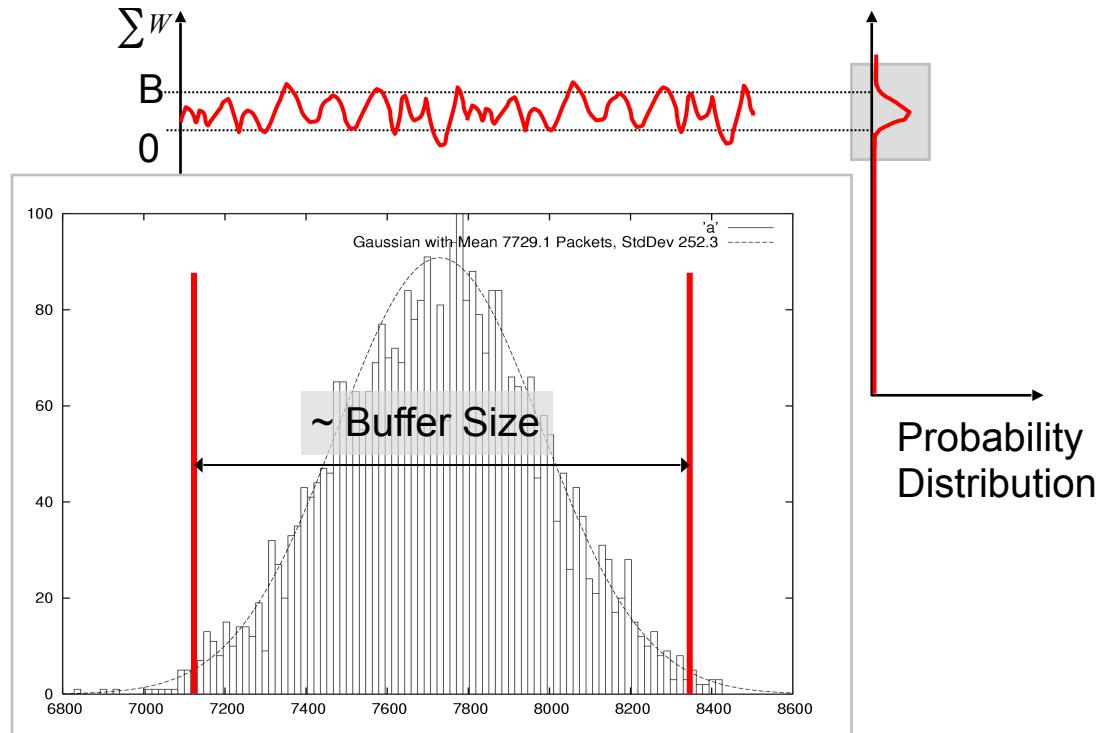


Synchronized Flows



- Aggregate window has same dynamics
- Therefore buffer occupancy has same dynamics
- Rule-of-thumb still holds.

Many TCP Flows



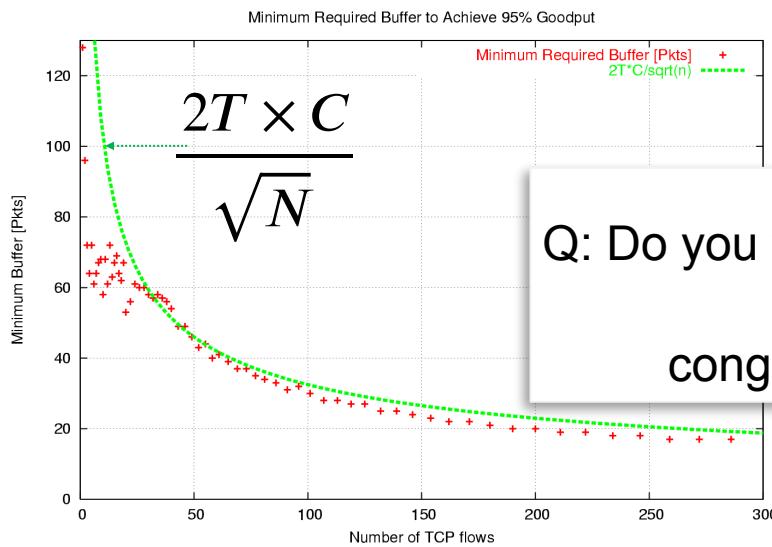
Many AIMD flows: 100% Throughput

$$B \geq \frac{2T \times C}{\sqrt{N}}$$



Example: $2T = 100ms$, $C = 10Gb/s$, $N = 1$
 $B \geq 1Gbit$

Example:
 $2T = 100ms$, $C = 10Gb/s$, $N = 10,000$
 $B \geq 10Mbit$



Q: Do you think $B \propto \frac{1}{\sqrt{N}}$ will hold for other
congestion control algorithms...?

You said

Nikhil Athreya

I think the assumptions made in the original paper are beginning to change. While they acknowledge that their work is mostly based off of TCP, they argue that single-packet sources (e.g. DNS) and constant-rate UDP sources (e.g. online games) can be modeled with short flows. However, a quick Google search tells me that (<https://www.caida.org/research/traffic-analysis/tcpudpratio/>) that the amount of UDP flows either has surpassed TCP flows or is at least comparable to it. I'm unsure how this would affect the results in this paper; perhaps because the results for short flows shows that average queue length is only dependent on link load and flow length, this shouldn't be a problem, but I'd be interested to hear more about it.

You said

Kevin Baichoo

It seems like from the backbone router vendor's will disavow this smaller buffer sizes, as otherwise they can't justify the prices for their equipment.

Esther Goldstein

Do buffer sizes today go by the rule-of-thumb equation, or has the equation been refined in a different way than what was proposed in the paper?

Isabel Victoria Papadimitriou

It seems that this is working under an assumption that TCP is TCP, and router manufacturers have to deal with that regarding buffer size. Why isn't it the other way around ("this is the buffer size, make congestion control work"), or is it possible to think about some sort of joint optimization of buffer size and paradigm?

1988

1994

2004

2006

2020

Congestion Avoidance
and Control
VJ & MK

High Performance
TCP in ANSNET
CV & CS

Sizing Router
Buffers
GA, IK, NM

Routers with
Very Small Buffers
ME, YG, AG, NM, TR

$$B = O(\log W)$$

Congestion Avoidance and Control¹

Van Jacobson,²
Lawrence Berkley Laboratory
Michael J. Karels,³
University of California at Berkeley
November, 1988

Introduction

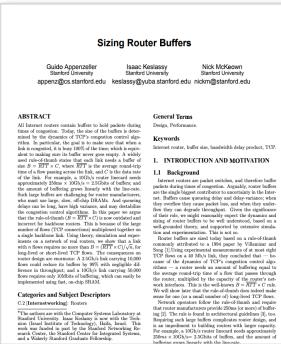
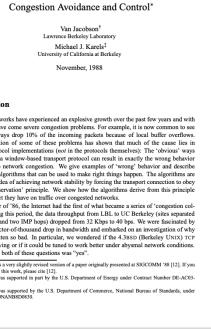
Computer networks have experienced an explosive growth over the past few years and with this growth has come a corresponding increase in network traffic. As a result, the number of Internet gateways drop 10% of the incoming packets because of local buffer overflows. Our investigation of some of these problems has shown that much of the cause lies in the transport protocol. This paper presents a detailed analysis of the ways to implement a window-based transport protocol that results in exactly the wrong behavior in response to network拥塞. We also present a number of simple algorithms that can be used to make things better. The algorithms are based on the principle of "fair sharing" of bandwidth among competing flows, and they obey a "packet conservative" principle. We show how the algorithms derive from the principle and how they work.

In October of '86, the Internet had its first of what became a series of separation collapses.⁴ During this period, the file download from LBL to UC Berkeley (a file separated by a single byte) was interrupted 100 times. The reason for this was that this sudden factor-of-thousand drop in bandwidth and emboldened an investigator of why things were happening to start sending more and more data. The result was that the link was misbehaving or it could be forced to work better under abnormal network conditions. The answer to both of these questions was "yes".

¹This paper was originally presented at the NOICON '88 [12]. If you wish to reference this paper, please cite [12].

²This work was supported by the U.S. Department of Energy under Contract Number DE-AC03-76SF00098.

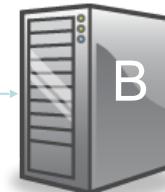
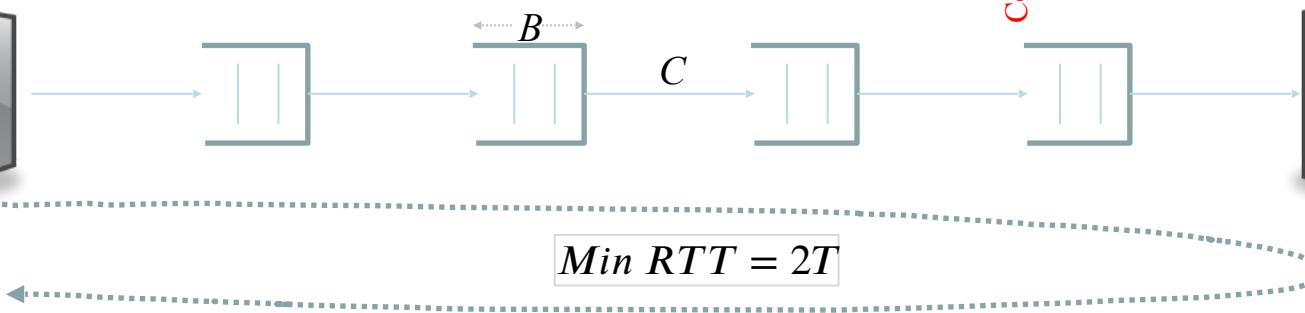
³This work was supported by the U.S. Department of Commerce, National Bureau of Standards, under Grant Number NBSRNSD883.

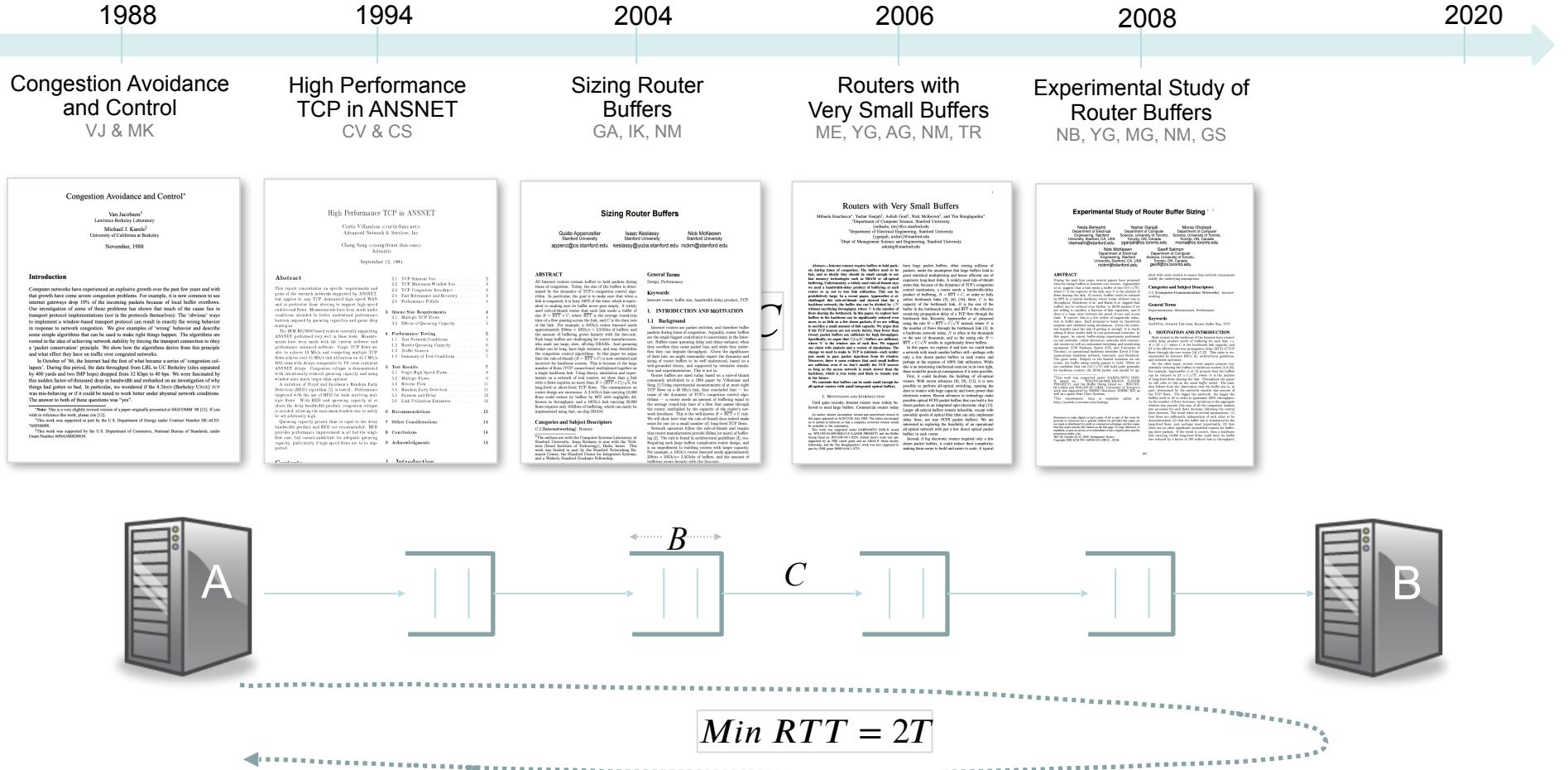


Assumptions
Consequences

1. Paced Traffic
2. Link utilization < 80%

Only 20-50 packet buffers.





Buffer Sizing Experiments

Small Buffers

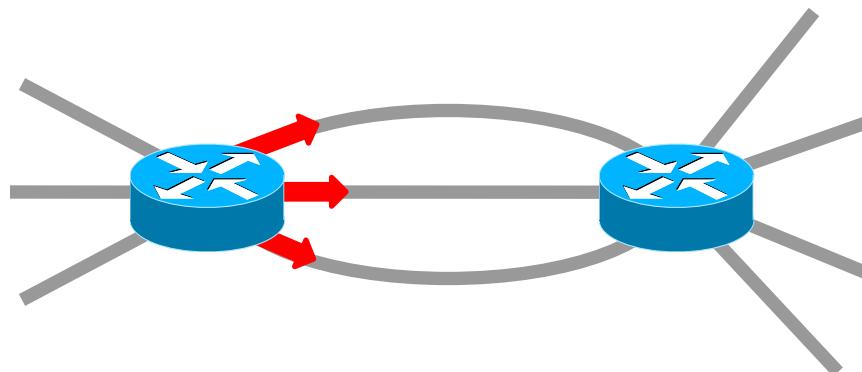
- Stanford University dorm network
- University of Wisconsin
- Internet2
- Level 3 Communications

Tiny Buffers

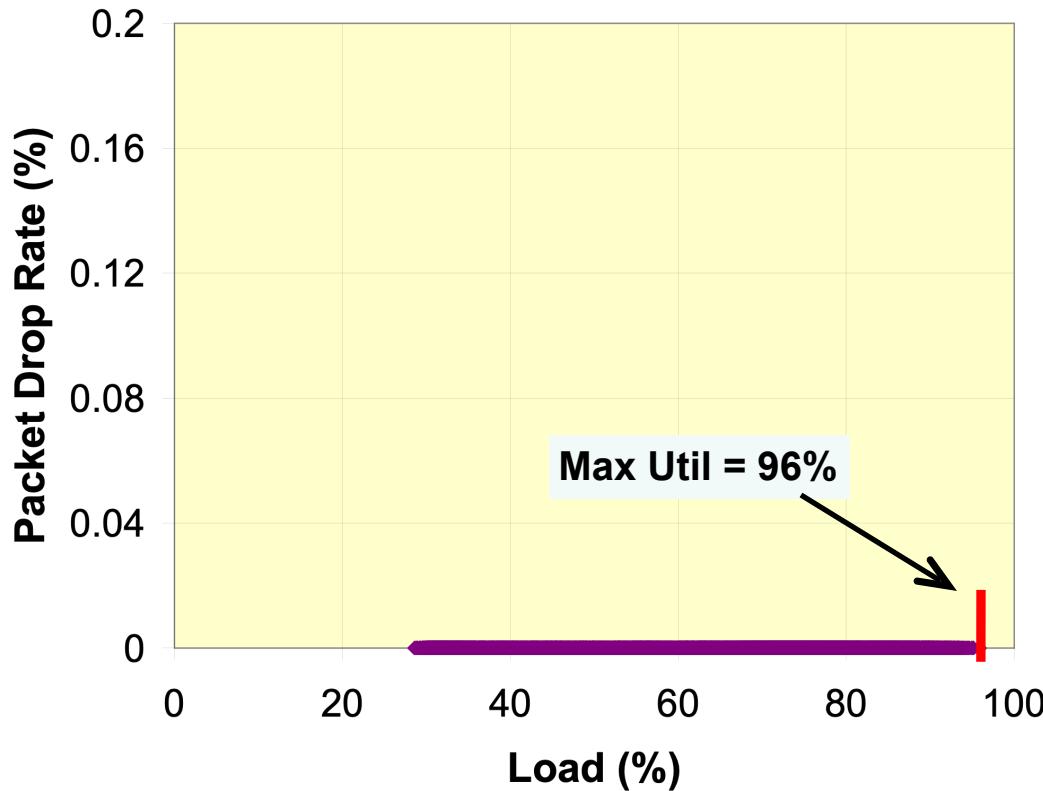
- Internet2
- Sprint Advanced Technology Lab
- University of Toronto

Level 3 Communications Experiments

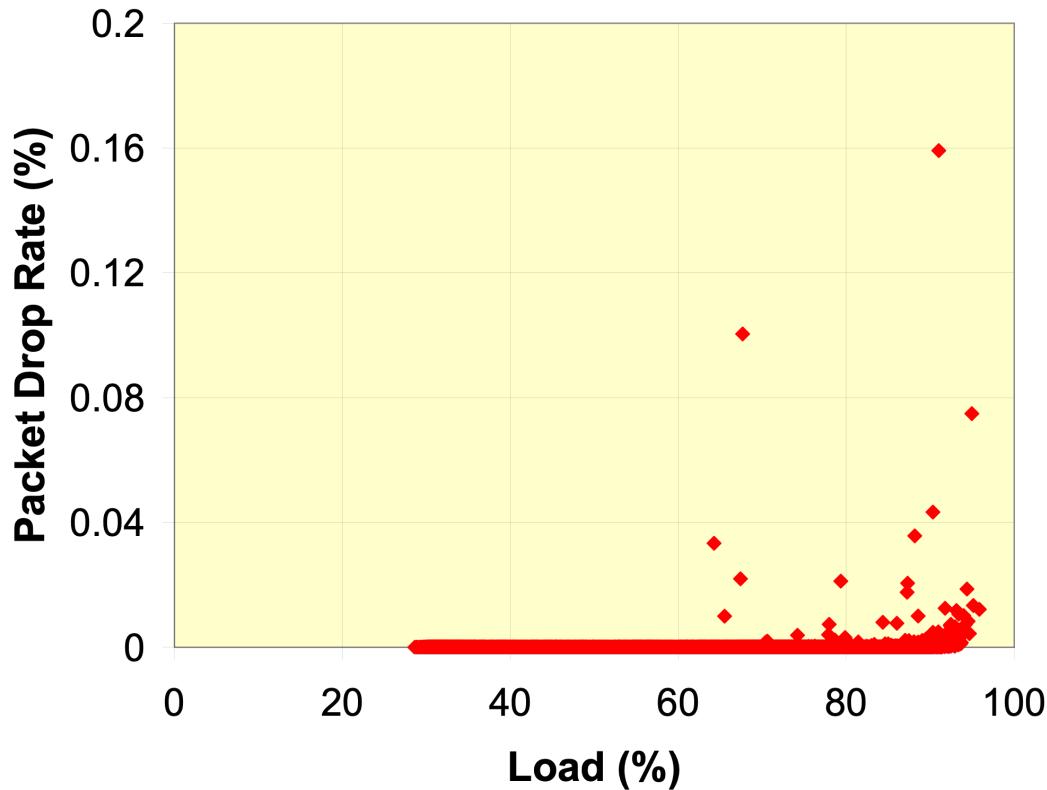
- High link utilization
- Long duration (about two weeks)
- Buffer sizes 190ms (250K packets), 10ms (10K packets), 2.5ms (2500 packets), 1ms (1000 packets)
- Load balancing over 3 links (2.5 Gb/s each)



Drop vs. Load, Buffer = 190ms, 10ms



Drop vs. Load, Buffer = 1ms

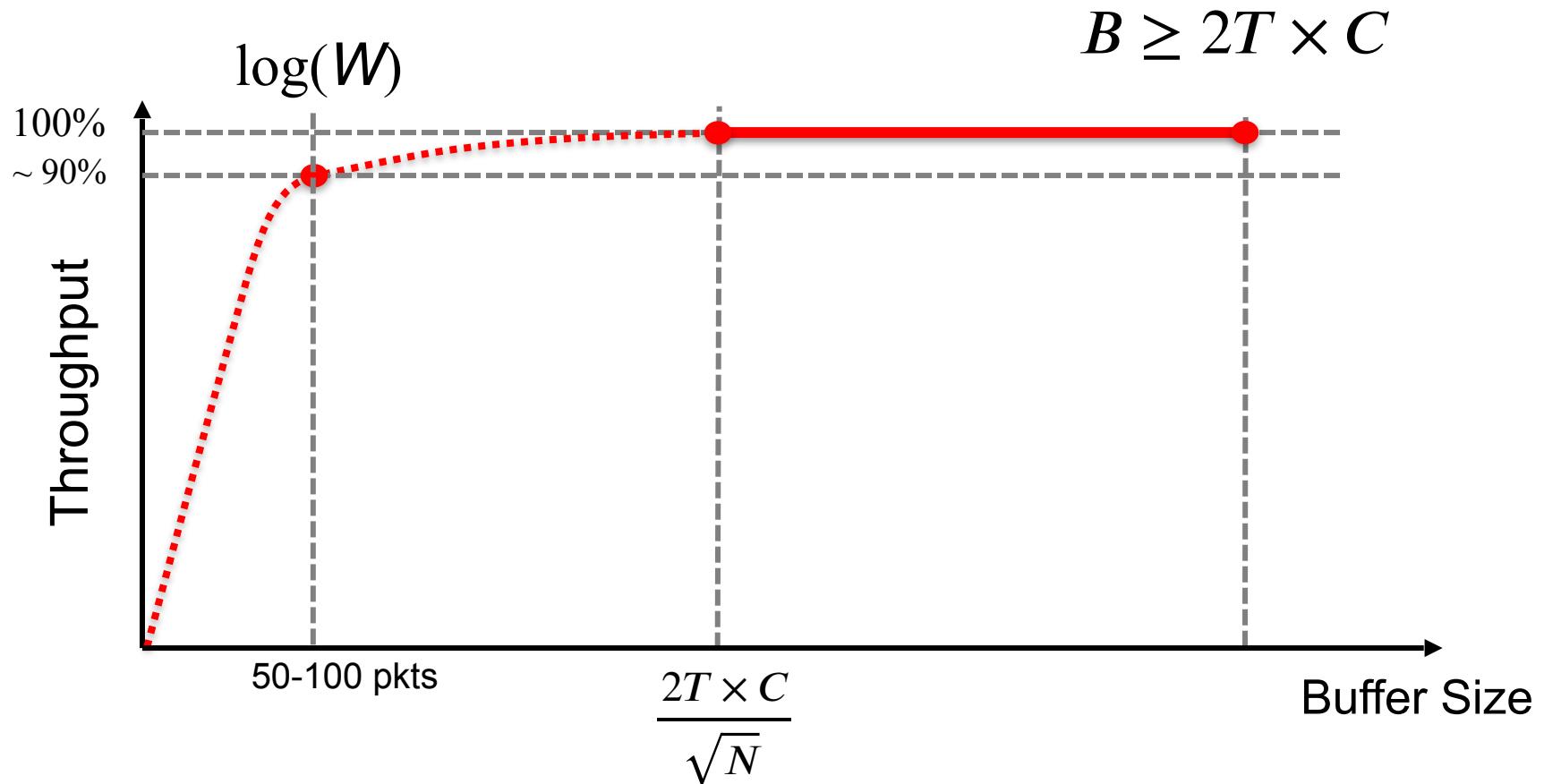


Internet2 Experiments



Neda Beheshti
2010

Summary of throughput and buffer size



End.