Evaluation of Different TCP Congestion Control Algorithms

Hilary Zhang
Zhengbing Bian
{hzhang|zbian}@cs.sfu.ca
Outline

- Introduction
- Our Approach
- Implementation Detail
- Results Discussion
- Reference
Why is Today’s Topic Important

- The algorithm for TCP congestion control is the main reason we can use the Internet successfully today.
- Without TCP congestion control, the Internet could have become history a long time ago.
Main Categories of Congestion Control Approach

- **Router-Centric vs. Host-Centric**
  - Router makes decision and informs the host
  - Host adjust sending rate

- **Reservation-Based vs. Feedback-Based**
  - Fixed capacity connection
  - Adjust rate according to feedback

- **Window-Based vs. Rate-Based**
  - Advertise window
  - Reservation of a fixed bandwidth
TCP Congestion Control

- Host-centric
- Feedback-Based

- Window-Based
  - *congestion window* – window is smaller when congestion is larger and vice versa
Review of TCP Congestion Control

- Slow Start, AIMD, Fast Retransmit, Fast Recovery
- Tahoe, Reno, NewReno, Sack, Vegas
- Evaluation measures
  - Effective resource utilization
  - Fair resource allocation
Our Approach

- Evaluate Reno, SACK, Vegas
- Effective Recourse Utilization
- Fairness between different delay links
- Competition between different versions TCP
- Effect of different Queuing algorithms
### Evaluation of TCP Congestion Control

#### Effective Recourse Utilization (1/2)

<table>
<thead>
<tr>
<th>Bandwidth Utilization</th>
<th>Reno</th>
<th>SACK</th>
<th>Vegas</th>
</tr>
</thead>
<tbody>
<tr>
<td>1% loss</td>
<td>88.3%</td>
<td>90.2%</td>
<td>98.5%</td>
</tr>
<tr>
<td>5% loss</td>
<td>34.5%</td>
<td>42.7%</td>
<td>75.0%</td>
</tr>
</tbody>
</table>
Effective Recourse Utilization (2/2)

Congestion Window size variation
Four lost in one window

Reno  SACK  Vegas
Evaluation of TCP Congestion Control

Fairness Between Different Delay Links

Simulation Topology

- S1 to R1: 10Mbps, 1ms
- R1 to S2: 1.5Mbps, 1ms
- S1 to K1: 10Mbps, 1ms
- R1 to K1: 10Mbps, 1ms
- R2 to S2: 10Mbps, 1ms
- R2 to K2: 10Mbps, Xms
- K1 to K2: 10Mbps, Xms

Evaluation of TCP Congestion Control
Bias on Long Delay Link

Bandwidth Occupation
(the same buffer size for all the three)

Reno
SACK
Vegas

Evaluation of TCP Congestion Control  Buffer size=10, Long delay=58ms
Queue Algorithm Effect

Fairness over Buffer size Changes

Evaluation of TCP Congestion Control
Competition Between Different Versions TCP

<table>
<thead>
<tr>
<th></th>
<th>S1</th>
<th>S2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Reno</td>
<td>SACK</td>
</tr>
<tr>
<td></td>
<td>Reno</td>
<td>Vegas</td>
</tr>
<tr>
<td></td>
<td>SACK</td>
<td>Vegas</td>
</tr>
</tbody>
</table>

Evaluation of TCP Congestion Control

Buffer size=15
Simulation Results

Bandwidth Occupancy

Reno-SACK

Reno-Vegas

SACK-Vegas

Retransmit:  
R-V 145/13170 vs. 0/5723  
S-V 142/13208 vs. 0/5682

Evaluation of TCP Congestion Control
Evaluation of TCP Congestion Control

Queue Algorithm Effect

Fairness over Buffer Size Changes

![Graphs showing ACK packets for Reno-Vegas with DropTail and SACK-Vegas with DropTail](chart)

Evaluation of TCP Congestion Control
Results Discussion

- Bandwidth Utilization of Vegas is High
- Reno and SACK Bias on Long Delay Link
- Vegas does not receive a fair share of bandwidth in the presence of Reno and SACK unless the buffer size are extremely small
- DropTail Buffer size will affect the fairness
- Other methods to cure the bias?
References

1. Kevin Fall, Sally Floyd, Simulation-based comparisons of Tahoe, Reno and SACK TCP, ACM SIGCOMM Computer Communication Review, v.26 n.3, p.5-21
2. S. Floyd, Congestion Control Principles, RFC2914, September 2000
6. Jeonghoon Mo, Richard J. La, Venkat Anantharam, and Jean Walrand, Analysis and Comparison of TCP Reno and Vegas
Thank you