TCP/Vegas

Yang Richard Yang

10/1/2001

Review AIMD Congestion Control
What Are the Desired Properties of a Congestion Avoidance Scheme?

- Efficiency (fully utilization)
- Fairness (resource sharing)
- Distributedness (no central knowledge for scalability)
- Convergence (fast convergence after disturbance, low oscillation)
  - responsiveness (speed to new state, lower or higher)
  - smoothness

Linear control

- The simplest control strategy

\[ x_i(t+1) = \begin{cases} 
    a_I + b_I x_i(t) & \text{if } y(t) = 0 \\
    a_D + b_D x_i(t) & \text{if } y(t) = 1 
\end{cases} \]
**AIMD State Space**

$\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_0$

- Fairness Line
- Efficiency Line

**Other Special Cases**

<table>
<thead>
<tr>
<th></th>
<th>Additive Decrease</th>
<th>Multiplicative Decrease</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Additive Increase</strong></td>
<td>AIAD</td>
<td>AIMD</td>
</tr>
<tr>
<td><strong>Multiplicative Increase</strong></td>
<td>MIAD</td>
<td>MIMD</td>
</tr>
</tbody>
</table>
TCP/Vegas

**Algorithms**

- **Retransmit**
  - first DUPACK & Timeout (finer grained timeout)
  - multiple losses: detect early, and no multiple reductions

- **Congestion avoidance**

- **Slow start**
  - increase every other RTT (congestion detection in the middle)
Issues: Rerouting

- If the propagation delay increases due to a new route, the flow should increase its window size to fill the pipe; however, the sender can interpret the increase as a signal of congestion and decrease its window size, which is undesired.

Issues: Persistent Congestion

- Remember the objective of Vegas is to keep \( w = x \cdot d + \delta \), where \( x \) is rate, \( d \) is delay, and \( \alpha \leq \delta \leq \beta \). Then if a new flow arrives and many flows are already sharing the link, the flow will observe a large \( d \). This large \( d \) is interpreted as propagation delay instead of queuing delay. Later, when some flows leave, the new flow will interpret the reduction of congestion as not enough window size and increase window size to maintain a persistent congestion.
Issues: Fairness

- TCP/Vegas grabs less bandwidth than TCP/Reno. The best way to think the problem is to think about a FIFO queue. The larger share of the queue occupied by a flow, the more bandwidth achieved by a flow. So how to delay TCP/Vegas in a TCP/Reno world?

- TCP/Vegas is largely insensitive to different round-trip time, and therefore has a different fairness semantics.