Background TCP Data Transfer with Inline Network Measurement

Tomoaki TSUGAWA, Go HASEGAWA, and Masayuki MURATA

Osaka University, Japan

IEICE Transactions on Communications, 2006
Outline

- Introduction
- Background data transfer with TCP
- ImTCP-bg: ImTCP background mode
  - Outline of ImTCP
  - ImTCP-bg mechanisms
- Performance evaluations
- Conclusions
Some Internet services should be operated in the \textit{background} through prioritization mechanisms.

- **TCP-based approaches**
  - \textit{TCP Nice} and \textit{TCP-LP}
    - Decrease the congestion window size when the RTTs increase.
  - \textit{TCP Reno}
    - Continues to increase its congestion window size until packet loss occurs, regardless of increases in RTTs.
TCP Nice and TCP-LP
- Utilize the available bandwidth inefficiently.
- Trade-off between the degree of interference with other traffic and the utilization of link bandwidth.

A novel background TCP mechanism based on bandwidth measurement.
- Achieves both background transfer and bandwidth utilization.
Introduction (3/4)

Basic idea

- Utilizes *available bandwidth* information as a network congestion indicator.
- Sets the *maximum* value of the cwd of the sender by the measurement results.
- Employ an RTT-based mechanism that dynamically determines the degree to which the cwd can decrease.
Fig.1 Change of congestion window size in TCP Reno and ImTCP-bg.
Background data transfer with TCP (1/3)

- Adjusts the data transmission speed by changing the cwnd in response to network congestion.

- Congestion indicators
  - *loss-based mechanism*
  - *RTT-based mechanism*
Consider two objectives:

- No effect on the foreground traffic.
- Full utilization of the network link bandwidth.

Bandwidth-based mechanism

- Ideal background data transfer mechanism.
- The existing methods utilize numerous probe packets and require too much time.
Background data transfer with TCP (3/3)

_inline measurement TCP (ImTCP)_

- Does not inject extra traffic into the network.
- Obtains bandwidth information every 1-4 RTTs, can follow the traffic fluctuation.

The ImTCP mechanism is integrated into the proposed background TCP

- However, the RTT-based mechanism cannot be discarded.
ImTCP-bg: ImTCP background mode

- Consists of two congestion control mechanisms:
  - A bandwidth-based mechanism with inline network measurement.
  - An *enhanced* RTT-based mechanism for adjusting the congestion window size.
Outline of ImTCP (1/4)

Adjusts the interval of data packets, and then calculates the available bandwidth by observing the change of ACK intervals.

Fig. 2 Inline network measurement by ImTCP.
Uses a search range to find the value of the available bandwidth

- A range of bandwidth that is expected to include the current available bandwidth.
- Determined using the previous measurement results.
Outline of 1mTCP (3/4)

Steps of the proposed algorithm:
1. Set initial search range.
2. Divide the search range.
Outline of ImTCP (4/4)

3. Send packet streams and check if an increasing trend exists in the packet arrival intervals.
   
   The increasing trend indicates that the transmission rate > the current available bandwidth.

4. Choose a sub-range.

5. Calculate the available bandwidth.
ImTCP-bg mechanisms (1/2)

Bandwidth-based mechanism

- Smoothes the measurement results of ImTCP:
  \[ \tilde{A} \leftarrow (1 - \gamma) \cdot \tilde{A} + \gamma \cdot A_{cur} \]
- ImTCP-bg sender then sets the upper limit of the cwnd:
  \[ \text{maxcwnd} \leftarrow \tilde{A} \cdot RTT_{min} \]
Enhanced RTT-based mechanism

- ImTCP does not always provide reliable measurement results
- Detects the network congestion when equation \(( RTT' / RTT_{\text{min}} ) > \delta \) is satisfied.
- Decreases its cwnd according to the equation \( cwnd \leftarrow cwnd \cdot ( RTT_{\text{min}} / RTT' ) \)
Performance (1/10)

Fig.3 Network model.

Buffer size: 1000 packets.
Packet size: 1500 bytes.
Performance (2/10)

\( \gamma \) setting

Fig. 4 Change of congestion window size and its upper limit

(a) \( \gamma = 0.1 \)

(b) \( \gamma = 0.9 \)

\[ N_{\text{web}} = 20 \]
\[ \delta = 1.2 \]
Performance (3/10)

Fig. 5 Effect of parameter $\gamma$.

$N_{\text{web}} = 20$

$\delta = 1.2$
Performance (4/10)

- RTT threshold $\delta$

- $N_{\text{web}} = 20$
- $\gamma = 1/8$

(a) Average utilization of available bandwidth.

(b) Average queue length.

Fig. 6 Effect of parameter $\delta$. 
Performance (5/10)

Case of one background connection

Fig. 7 Results of one connection case.

(a) Average of throughput.

(b) Average of download time.
Performance (6/10)

Case of multiple background connections

Five background TCP connections join the network at 0, 50, 100, 150, and 200 seconds, and end data transmissions at 500, 450, 400, 350, and 300 seconds.

<table>
<thead>
<tr>
<th></th>
<th>TCP Reno</th>
<th>TCP Nice</th>
<th>TCP-LP</th>
<th>ImTCP-bg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average queue length [packet]</td>
<td>583.34</td>
<td>8.44</td>
<td>64.63</td>
<td>44.11</td>
</tr>
</tbody>
</table>

Table 1: Average queue length in the case of five connections.
Performance (7/10)

Fig. 8 Change of throughput with five connections.

(a) TCP Reno.  
(b) TCP Nice.
Performance (8/10)

(c) TCP-LP.

(d) ImTCP-bg.
Performance (9/10)

Fig. 9 Utilization of Available bandwidth with multiple connections.

(a) Number of connections: 1.

(b) Number of connections: 2.

Fig. 9 Utilization of Available bandwidth with multiple connections.
Performance (10/10)

(c) Number of connections: 5.

(d) Number of connections: 10.
Conclusions

- A new background data transfer mechanism that uses an inline network measurement technique.
- Provides a background data transfer without interfering with other traffic.
- Employs an enhanced RTT-based mechanism, which dynamically determines the control parameters.
Reference