EasyTrans: Enable Fast Iteration of Transport Protocol

Jie Zhang*, Chuan Ma*, Wei Wang†, Kai Zheng†, Yong Cui*

Tsinghua University*, Huawei Technologies†
Background and Motivation

• Transport protocol iterations accelerate
  • Emerging along with evolving applications

• Trouble in developing protocols
  • “Too coupled to use” or “Too extensible to perform well”

• Goal: EasyTrans
  • Enable easy development and retains good performance
EasyTrans: Architecture

- **Focus on** specific module and **ignore** unnecessary parts
- Various **calling modes** that **improve performance**
EasyTrans: Calling modes

- Four different calling modes
- High performance even with sophisticated computation
  - E.g. Machine learning

![Diagram showing calling modes](image)

**Fig. 2.** An example of events processing modes.
EasyTrans: Evaluation

- QUIC-based implementation
- Modules and Calling modes introduce slight overhead (<5%)

<table>
<thead>
<tr>
<th>Modules</th>
<th>Calling Modes</th>
<th>Goodput(Gbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>No module</td>
<td>-</td>
<td>1.895</td>
</tr>
<tr>
<td>Congestion control (a) and stream scheduling (b)</td>
<td>a: Unperiodic</td>
<td>1.799</td>
</tr>
<tr>
<td></td>
<td>b: Per-event</td>
<td></td>
</tr>
<tr>
<td>a and b</td>
<td>a: Periodic (T=10ms)</td>
<td>1.790</td>
</tr>
<tr>
<td></td>
<td>b: Per-event</td>
<td></td>
</tr>
<tr>
<td>a and b</td>
<td>a: Per-event</td>
<td>1.786</td>
</tr>
<tr>
<td></td>
<td>b: Per-event</td>
<td></td>
</tr>
</tbody>
</table>
EasyTrans: Future work

- Develop more modules:
  - ACK
    - Offers diverse strategies to packet acknowledgment
    - Challenge: Hard to decouple
  - Multipath
    - Make full use of the bandwidth
    - Challenge: Change the structure of EasyTrans
- Extensive evaluation
  - Performance
  - Fidelity