Coding and congestion control in transport
draft-kuhn-coding-congestion-transport-00

Nicolas KUHN
Emmanuel LOCHIN
François MICHEL
Introduction

• [RFC5681] : TCP is a loss-based congestion control
• Coding mechanisms could:
  • Hide congestion signals to the sender
  • Deal with tail losses or with networks with non-congestion losses efficiently

• This memo :
  • Simple best practices on how coding and congestion control could coexist

• [DISCLAIMER] The proposed recommendations apply for coding at the transport layer (coding for tunnels is out-of-the scope of the document)
Base solution

- The receiver MUST indicate to the sender that one or multiple packets have been recovered using a coding scheme
  - Such "repaired packet signal" could be based
    - on existing signals (even if the existing signal was not designed for that purpose, such as ECN) or
    - on new type of signals (such as a RECOVERED frame in QUIC)

- The sender MUST be able to detect the "recovered packet signal"
  - The base solution does not describe how the sender reacts to such signal
Sender-side coding solutions

• Coded packets without considering CWND progression
• Ex:

  ![Graph showing congestion, data, and repair windows]

  • Repair window can be fixed or based on another CC
Sender-side coding solutions

• Coded packets driven by CWND progression
• Ex:

• Repair window can be fixed or based on another CC
Sender-side reaction to recovered packet signals

• The sender congestion control considers recovered packet signals as congestion-implied packet losses
Sender-side reaction to recovered packet signals

• The sender adapts its window reduction to recovered packet signals (eg RFC 8511 for ECN signals)
Sender-side reaction to recovered packet signals

• The sender ignores recovered packet signals
## Summary

<table>
<thead>
<tr>
<th>Sender-side reaction to recovered packet signals</th>
<th>Sender-side coding solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Coded packets without considering CWND progression</td>
</tr>
<tr>
<td>React as loss</td>
<td>Fairness : ~</td>
</tr>
<tr>
<td></td>
<td>Real-time : +</td>
</tr>
<tr>
<td></td>
<td>Bulk : ~</td>
</tr>
<tr>
<td>Adapt window reduction</td>
<td>Fairness : ~</td>
</tr>
<tr>
<td></td>
<td>Real-time : +</td>
</tr>
<tr>
<td></td>
<td>Bulk : +</td>
</tr>
<tr>
<td>Ignore signals</td>
<td>Fairness : -</td>
</tr>
<tr>
<td></td>
<td>Real-time : +</td>
</tr>
<tr>
<td></td>
<td>Bulk : +</td>
</tr>
</tbody>
</table>
Questions

• Do we need this kind of work in the IRTF? IETF?
• Any comments?