NWCRG
Internet-Draft
Intended status: Informational
Expires: April 29, 2020

N. Kuhn, Ed.
CNES
E. Lochin, Ed.
ISAE-SUPAERO
F. Michel, Ed.
UCLouvain

October 27, 2019

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

Abstract

There are discussions on how loss-based congestion controls consider lost packets that have been recovered by a coding mechanism. This document analyses to what extent transport protocols could ignore such signals and proposes best current practices on the interaction between congestion control and coding mechanism at the transport layer. Coding for tunnels is out-of-the scope of the document. Examples of interest for the proposed solution is to better deal with tail losses or with networks with non-congestion losses.

Table of Contents

1. Introduction

[RFC5681] defines TCP as a loss-based congestion control. Coding mechanisms can be deployed and could hide congestion signals to the sender.

Because loss-based and delay-based congestion controls are deployed in the current Internet, this memo discusses simple best practices on how coding and congestion control mechanisms could coexist.

The proposed recommendations apply for coding at the transport layer (coding for tunnels is out-of-the scope of the document). Examples of interest for the proposed solution is to better deal with tail losses or with networks with non-congestion losses.

2. Base solution description

The base solution can be described as follows:

- o 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).

- o The sender MUST be able to detect the "recovered packet signal". The base solution does not describe how the sender reacts to such signal.

The proposed solution applies for coding in the transport layer. The proposed approach is inline with the one in [I-D.swett-nwcrg-coding-for-quic].

The proposed solution does not applies for the interaction between coding under the transport layer (i.e. not end-to-end), such as coding for tunnels.

3. Sender-side coding solutions

This section presents solutions for a sender to add application coding.

3.1. Coded packets without considering CWND progression

In this solution, the coded packets are sent on top of what is allowed by a congestion window. Examples of the solution could be adding a given pourcentage of the congestion window as supplementary packets or sending a given amount of coded packets at a given rate. The redundancy flow can be decorrelated from the congestion control that manages source packets : a secondary congestion control can be introduced, such as in coupled congestion control for RTP media [I-D.ietf-rmcat-coupled-cc]. An example would be to exploit a lower than best-effort congestion control [RFC6297].

The advantage of such solution is that coding would help in challenges cases where transmission losses are persistent.

The drawback of such solution is that it may result in coding solutions being unfair towards non-coding solutions. This solutions may result in adding congestion in congested networks.

3.2. Coded packets driven by CWND progression

In this solution, the coded packets are sent within what the congestion window allows, such as in [CTCP]. Examples of the solution would be sending coded packets when there is no more data to

transmit or preferably send coded packets instead of the following packets in the send buffer.

The advantage of this solution is that it does not contribute in adding more congestion than the congestion window allows. Indeed, all traffic (source and redundancy) is controlled by one congestion control only and TCP metrics for fairness can be indifferently applied in this case.

The main drawback is the decrease of goodput if coded packets are sent but are not used at the client side.

4. Sender-side reaction to recovered packet signals

Delay-based congestion controls ignore packets that have been repaired with coding. There is no need to define best current pratices in this case. However, more discussions are required for congestion controls that use loss as congestion signals (potentially among other congestion detection mechanism).

4.1. The sender congestion control considers recovered packet signals as congestion-implied packet losses

In this solution, the sender reacts to recovered packet signals as to congestion-implied packet losses. That being said, this does not necessarily means that the packets have actually been lost. The server may have other means to identify that the packet was just out-of-ordered and ignore the recovered packet signals.

The advantages of the solution are (1) that coding mechanisms do not hide congestion signals, such as packets voluntary dropped by a AQM [RFC7567] and (2) packets may be recovered faster than with traditionnal retransmission mechanisms.

The drawback of this solution is that, if there is a high non-congestion loss rate, the congestion control throughput may decrease drastically. Reporting this amount of loss to a sender may reduce the application goodput when there is no actual congestion.

4.2. The sender adapts its window reduction to recovered packet signals

In this solution, the sender does not reduce the congestion window with the same amount when the "recovered packet signal" is received, i.e. when a packet has been lost but recovered. Example of this solution could be based on [RFC8511] or considering that recovering an isolated packet is not an actual sign of congestion.

The advantage of the solution is that in cases where there is no actual congestion, coding could help in improving the transmission without ignoring congestion signals.

The main drawback is the precised design of the solution and its interaction with AQM mechanisms [RFC7567]. Moreover there may be fairness issues since AIMD convergence may not be guaranteed.

4.3. The sender ignores recovered packet signals

This is the case for delay-based congestion controls. The interaction between delay-based congestion controls and the delay induced by a coding mechanisms is an open research activity. That being said, a potential approach would be that loss-based congestion control ignores the "recovered packet signal".

The advantage of this solution is that coding would provided substantial benefits in cases where there are transmission losses.

However, this solution hides potential congestion losses, making it unfair to other congestion controls.

5. Summary

This section provides a summary on the content in previous sections. The Figure 1 sums up some recommendations. It is worth pointing out that the "coding without congestion" considers that coded packets are sent along with original data packets, in opposition with the solution where coded packets are transmitted only when there is no more original packets to transmit. Moreover, the values indicated in this Figure consider a channel that does not exhibit a high loss pattern.

Sender-side reaction to recovered packet signals	Sender-side coding solutions: Coding adding congestion	Sender-side coding solutions: Coding without congestion
React as loss	fairness: ~ real-time: + bulk: ~	fairness: ++ real-time: + bulk: -
Adapt window reduction	fairness: ~ real-time: + bulk: +	fairness: + real-time: + bulk: -
Ignore signals	fairness: - real-time: + bulk: +	fairness: - real-time: + bulk: -

Figure 1: Recommendations

6. Acknowledgements

Many thanks to TBD.

7. Contributors

TBD

8. IANA Considerations

This memo includes no request to IANA.

9. Security Considerations

Security section goes here.

10. Informative References

[CTCP] Kim (et al.), M., "Network Coded TCP (CTCP)", arXiv 1212.2291v3, 2013.

[I-D.ietf-rmcat-coupled-cc]
Islam, S., Welzl, M., and S. Gjessing, "Coupled congestion control for RTP media", draft-ietf-rmcat-coupled-cc-09 (work in progress), August 2019.

[I-D.swett-nwcrg-coding-for-quic]
Swett, I., Montpetit, M., Roca, V., and F. Michel, "Coding for QUIC", draft-swett-nwcrg-coding-for-quic-03 (work in progress), July 2019.

[RFC5681] Allman, M., Paxson, V., and E. Blanton, "TCP Congestion Control", RFC 5681, DOI 10.17487/RFC5681, September 2009, <https://www.rfc-editor.org/info/rfc5681>.

[RFC6297] Welzl, M. and D. Ros, "A Survey of Lower-than-Best-Effort Transport Protocols", RFC 6297, DOI 10.17487/RFC6297, June 2011, <https://www.rfc-editor.org/info/rfc6297>.

[RFC7567] Baker, F., Ed. and G. Fairhurst, Ed., "IETF Recommendations Regarding Active Queue Management", BCP 197, RFC 7567, DOI 10.17487/RFC7567, July 2015, <https://www.rfc-editor.org/info/rfc7567>.

[RFC8511] Khademi, N., Welzl, M., Armitage, G., and G. Fairhurst, "TCP Alternative Backoff with ECN (ABE)", RFC 8511, DOI 10.17487/RFC8511, December 2018, <https://www.rfc-editor.org/info/rfc8511>.

Authors' Addresses

Nicolas Kuhn (editor)
CNES

Email: nicolas.kuhn@cnes.fr

Emmanuel Lochin (editor)
ISAE-SUPAERO

Email: emmanuel.lochin@isae-supaero.fr

Francois Michel (editor)
UCLouvain

Email: francois.michel@uclouvain.be