Managing Congestion in Data Center Network using Congestion Notification Algorithms

Rohit P. Joglekar[1], Prof. P. S. Game[2]

[1]*Student, Dept. of Computer Engineering, PICT, Pune, India*
[2]*Professor, Dept. of Computer Engineering, PICT, Pune, India*

---***---

Abstract - *With the rapid growth in Data Center Network (DCN), many problems are observed by the researchers. These problems are related to its architecture, congestion control and TCP throughput. Nowadays, there is vast improvement in the Data Center Network with respect to Congestion Notification. In today's scenario Data Center Ethernet (DCE) is mostly used as compared to other technologies. Ethernet is low in cost and primary protocol used in DCN. But this Ethernet technology may also lead to packet drop and low bandwidth utilization, consequently lead to congestion. To solve this problem various Ethernet protocols are being developed to prevent congestion at the switch such as Backward Congestion Control (BCN), Enhanced Forward Explicit Congestion Notification (E-FECN), Forward Explicit Congestion Notification (FECN), Quantized Congestion Notification (QCN), Approximate Fair Quantized Congestion Notification (AF-QCN) and Fair Quantized Congestion Notification (FQCN). Among those algorithms QCN is ratified as the formal standard. In this paper, we are going to cover all the aspects of recent research activities in DCNs, in terms of network congestion notification algorithms. We present a brief overview of the congestion problem along with previously proposed solutions.*

***Key Words*: Quantized Congestion Notification, Data Centre Network, Congestion Control.**

1. INTRODUCTION

Data center plays an important role in storing, accessing, processing data and consist of a large number of storage devices [1]. Along with storage devices, Data Center Networks has its servers to manage the data and the Ethernet switches for connectivity between those devices. Data center network is used in various sectors including government, business, education and financial sector [2]. Data Center Network is expanding with an exponential rate and with this development there observed many problems such as congestion, packet loss and TCP Incast [1]. Traditionally, TCP protocol is the only protocol that ensure the reliable transmission at the transport layer [3] while other protocol such as UDP does not have that facility. So the advancement in Ethernet protocol can overcome the penalizing effect on transport layer protocol. Therefore, congestion notification technique comes into the picture to reduce the load on the transport protocol. With advancement in the Ethernet link to 10 GBPS, there is improvement in bandwidth usage consequently.

This Survey paper will mostly going to focus on solutions provided by various algorithms.

This paper is organized as follows: We describe Congestion Control mechanisms with Congestion Notification algorithms in section 2. In section 3 we compares different congestion notification algorithms proposed for DCNs and finally conclude in Section 4.

2. CONGESTION CONTROL IN DATA CENTERS

Due to features like ease of use and low in cost Ethernet become most popular network protocol for communication in data center network. There are many other protocol developed by, the Internet Engineering Task Force (IETF) [4] and the IEEE Data Center Bridging Task Group of IEEE 802.1 Working Group [5]. Transport protocol, such as TCP is responsible for reliable transmissions in IP networks. So there is need for managing and controlling the reliability in transmission with the help of Congestion Notification algorithm. Thus, one area of the Ethernet extensions that can enhanced the performance of transport protocol without penalizing the throughput is Congestion Notification [6].

Congestion notification is a data link layer traffic management protocol that in initial stage, monitors the status of queues and then pushes the congestion at the edge of the network. At the edge of the network, rate limiters are works, which handles traffic to avoid frame losses at the Ethernet Switch [8]. Detecting and generating feedback is done at the switch and controlling is done at the source itself. Switch sends the congestion message to the source based on the calculated congestion measure. Rate regulators at the sources will adjust the rate of individual flows according to congestion feedback messages received from switches. In this survey paper, we are going to study and compare the congestion notification algorithms proposed for the Ethernet extensions in data centers. Project IEEE 802.Qau is one of the research standard that concerned with specifications of an Ethernet Layer or Layer 2 congestion notification mechanism for DCNs basically belongs to IEEE Data Center Bridging Task Group. Several congestion notification algorithms have been proposed, e.g., BCN [6], FECN [8], enhanced FECN (E-FECN) [9], QCN [10], AF-QCN [11] and FQCN [12]. Congestion notification algorithms with their system models are shown in below figures; BCN, FECN, E-FECN, QCN, AF-QCN and FQCN which are all queue-based congestion notification algorithms.

2.1 Backward Congestion Notification (BCN)

As shown in Fig.1 BCN mechanism was introduced by Bergasamo et al. at Cisco [6]. BCN is a rate-based closed-loop feedback mechanism. BCN works in three phases: Congestion Detection, Signaling, and Source Reaction.

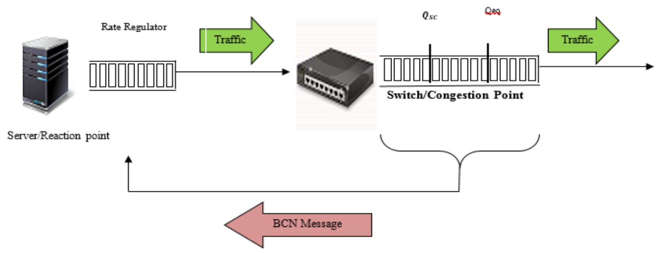


Fig -1: Backward Congestion Notification (BCN)

2.1.1 Congestion Detection:

As shown in above fig. 1, eq called as equilibrium queue length and SC called as severe congestion queue length are threshold values. First of all sampling is done at switch when the incoming packets arrived and the probability P is calculated. After sampling e, called as congestion measure is calculated.

2.1.2 Backward Signaling:

BCN contains three kinds of signals: PAUSE frames, BCN normal messages and BCN STOP messages. The arriving packets at the switch are sampled and for each sampled packet the feedback BCN message is generated [6] [12].

2.1.3 Source Reaction:

When a normal BCN message reaches the end station, e is calculated and updated value get set for rate regulator. The rate regulator tag (RRT) uses IEEE 802.1Q tag format [1] [6]. When a BCN message is received at the source, Additive Increase and Multiplicative Decrease (AIMD) algorithm is used to adjust values at the rate regulator.

2.2 Forward Explicit Congestion Notification (FECN) and Enhanced FECN (E-FECN)

FECN uses a close-loop explicit rate feedback control mechanism. FECN uses rate based load sensor to detect the congestion. At initial stage all flows are set to maximum value and probe is used to manage the congestion condition along the path from the source to the destination [9]. If the available bandwidth at switch is smaller than the value of the rate field in the probe, rate field get modified along the forward path. When the probe messages is received at the sources from the destination, the rate regulator adjusts the sending rate and allocate proper bandwidth to each link. At the congestion point, the switch measures the average arrival rate of each flow and the instantaneous queue length during the interval.

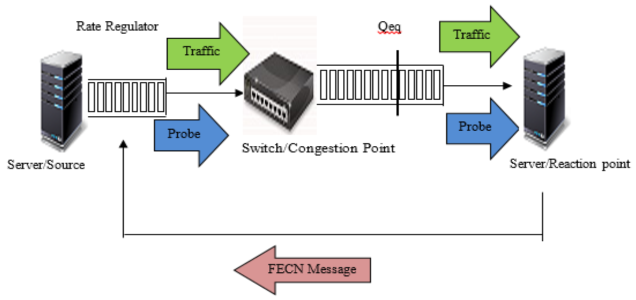


Fig -2: Forward Explicit Congestion Notification (FECN)

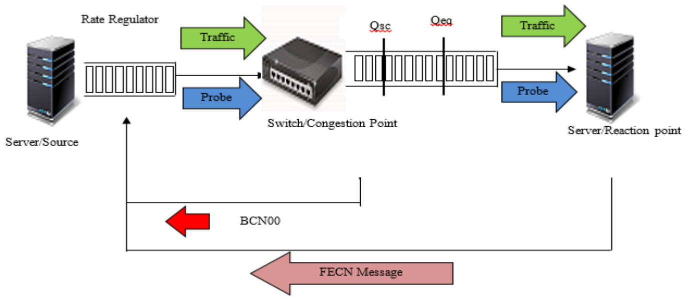


Fig -3: Enhanced Forward Explicit Congestion Notification (FECN)

E-FECN operations assume the same operations as those in FECN as shown in fig. 3. Here BCN00 message is used, which is adapted by switch under the situation of severe congestion.

2.3 Quantized Congestion Notification (QCN)

Quantized Congestion Notification (QCN) is developed for IEEE 802.1Qau to provide congestion control at the Ethernet Layer or Layer 2 by the IEEE Data Center Bridging Task Group [5]. QCN is accepted as a formal standard. The drawback of QCN is the rate unfairness of different flows when sharing one bottleneck link. Basically it is having two parts: Congestion Point (CP) and Rate Limiter (RL). At CP, the switch is attached to oversubscribed link. Switch samples incoming packets and send feedback about the severity level, if any, to the source point. At one traffic source, RL decreases its sending rate based on the congestion notification message received from CP and increases its rate voluntarily to recover lost bandwidth then probe for extra available bandwidth [10].

2.3.1 The CP Algorithm:

As shown in above fig. 4, Q is set as a threshold value. CP maintain the occupancy of buffer at that particular value called Equilibrium value. At time t, CP samples the incoming packets with a sampling probability p(t) and then find out the value of Fb.

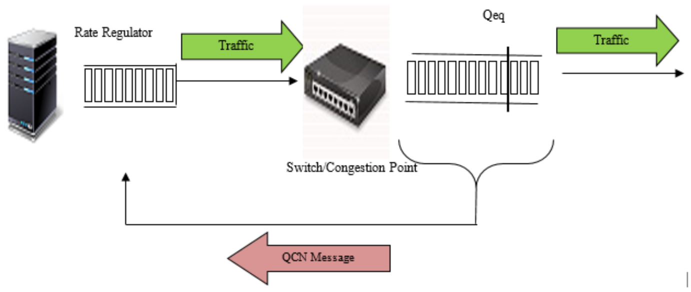


Fig -4 Quantized Congestion Notification (QCN)

If feedback is negative, either the buffer or the link or both are oversubscribed. Then congestion notification message containing the value of quantized feedback, denoted as ΨFb (t), is sent back to the source of the sampled packet; otherwise no feedback message is sent. At each sampling event, the sampling probability is updated. In the default implementation, the congestion feedback value is quantized to 6 bits and the maximum quantized value of feedback is 64, so the maximum sampling probability is 10.

2.3.2 The RL Algorithm:

When congestion message is received from the CP, RL adjust its sending rate. It either increase its sending rate to recover the lost bandwidth or decrease based on the basis of incoming feedback.

Rate decrease:
When feedback message is received, the current rate is set as the target rate. Here value of feedback is reduced to 50%, mince Fb (t) = 1/2. In the default implementation, the maximum quantized value of feedback is 64.

Rate Increase
Two modules, Byte Counter (BC) and Rate Increase Timer (RIT) are introduced in RL for rate increases. For counting the number of bytes transmitted by the traffic at source BC is used. Based on BC only, it will take a long time for a low rate source to increase its sending rate. This may leads to low bandwidth utilization if there is extra available bandwidth. Rather than limiting bandwidth rate it increases bandwidth utilization with the introduction of RIT. RIT also increases the source's sending rate periodically.

2.2 Fair Quantized Congestion Notification (FQCN)

Fair Quantized Congestion Notification (FQCN) sends the congestion message to each of the flow. Both QCN and FQCN are differs only in CP algorithm and RL is implemented same as there in QCN. QCN randomly samples the traffic packets at CP and sends feedback to the congestion link where culprit is observed. FQCN addresses following deficiencies: 1) It identifies congestion culprits on the basis of the overrated flow; 2) Through joint queue and per flow monitoring, it feedbacks individual congestion status to each culprit through multi-casting and this helps FQCN to achieve statistical fairness Congestion Culprits Identification.

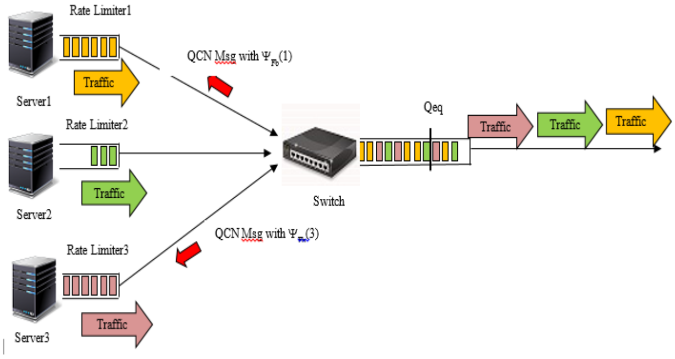


Fig-5 Fair Quantized Congestion Notification (FQCN)

If the calculated congestion feedback value is negative then with the use of multi-casting, the congestion notification message will be sent back to all identified overrated flows. For each overrated flow, the quantized congestion feedback value of the overrated flow is calculated.

Quantized congestion feedback value in each congestion notification message is proportional to ΨFb(t). ΨFb(i,t)is also quantized to 6 bits because Fb(t) is quantized to 6 bits. Fig. 6 describes the FQCN system model having three sources, sources 1 and 3 are identified as congestion culprits, and the congestion notification message is fed back to sources 1 and 3.

FQCN differs from both QCN and AF-QCN. QCN does not distinguish flow dependent congestion information and feedbacks the same congestion status to the source of the randomly sampled packet, while the congestion feedback value in the congestion notification message in both AF-QCN and FQCN is flow dependent. Thus, resolving congestion at switch is much faster with FQCN than that with QCN or AF-QCN. Moreover, the signaling overhead of FQCN is lighter than that of QCN and AF-QCN because congestion messages could also be received by low rate sources, which are not the real congestion culprits. AF-QCN can also identify congestion culprits, but it still feeds congestion information only back to the source of the randomly sampled packet.

3. COMPARISONS AND DISCUSSION

All Congestion Notification algorithm such as-BCN, FECN, E-FECN, QCN and FQCN summarized in Table 1. All of them are concerned with provisioning congestion notification in DCNs. We will discuss and compare the advantages and disadvantages of these congestion notification algorithms in the following aspects.

3.1 Fairness:

BCN achieve only proportional fairness but not max-min fairness. The fairness in FECN (EFECN) is calculated by the congestion detection algorithm at the switch. Similar to BCN, the feedback message is only sent to the source of the sampled packet in QCN, and therefore QCN only achieves proportional fairness rather than max-min fairness. One drawback of QCN is the rate unfairness with respect to different flows when sharing single bottleneck link. Such rate unfairness also degrades the TCP throughput in synchronized readings of data blocks across multiple servers. This drawback is overcome by Fair Quantized Congestion Notification (FQCN) [12], that improve the fairness of multiple flows sharing single bottleneck link. The main aim of FQCN is to feedback the global and per-flow congestion information to all culprits when the switch is congested [1].

3.2 Feedback Control:

From the system models shown in above figures, it is observed that BCN, QCN and FQCN use backward feedback control [10], FECN employs forward feedback and EFECN implements BCN00 messages to inform the source about the occurrence of congestion [1].

3.3 Overhead:

High and unpredictable overhead is what BCN can provide. QCN is having smaller overhead than that of BCN since there is only negative QCN message to reduce the sending rate. FECN is low in overhead but it can be predicted because the FECN message is sent periodically with a small payload of about 20 bytes [1]. E-FECN is having much larger overhead than that of FECN due to the BCN00 signal involved in the E-FECN algorithm. The Overhead of FQCN is medium and unpredictable but smaller than that of QCN in multiple flows sharing a single bottleneck link [9].

3.4 Rate of Convergence to Fair State:

BCN is slow in convergence to the fair state. FECN and E-FECN can achieve perfect fair state within a few RTT (Round Trip Time) with all sources get the same feedback [1].

3.5 Congestion Regulation:

The source rate in BCN, QCN and FQCN can be reduced more quickly than that in FECN [8]. The message in BCN and QCN is sent directly from the CP while the probe message in FECN has to take a round trip before it get return to the source. Using the BCN00 message in E-FECN, congestion adjustment speed is improved.

3.6 Throughput Oscillation:

BCN incurs large oscillations in throughput. FECN and E-FECN do not incur large oscillations in source throughput. In QCN and FQCN throughput oscillation is improved with the rate increase determined jointly by Byte Counter and Rate Increase Timer at the source.

3.7 Load Sensor:

Queue dynamics in BCN, QCN and FQCN send back to the sources, while FECN detect congestion. In addition to the rate-based sensors, queue monitor is also employed for severe congestion notification in EFECN.

3.8 Link Disconnection:

BCN, E-FECN, QCN and FQCN can employ the reactive feedback message to inform the source when any link in network get broken down and suddenly decrease or stop the transmission of packets[7][1]. While in case of FECN, there is no reactive feedback message and the probe might not return back to the source thus causing packet loss.

3.9 Fast Start:

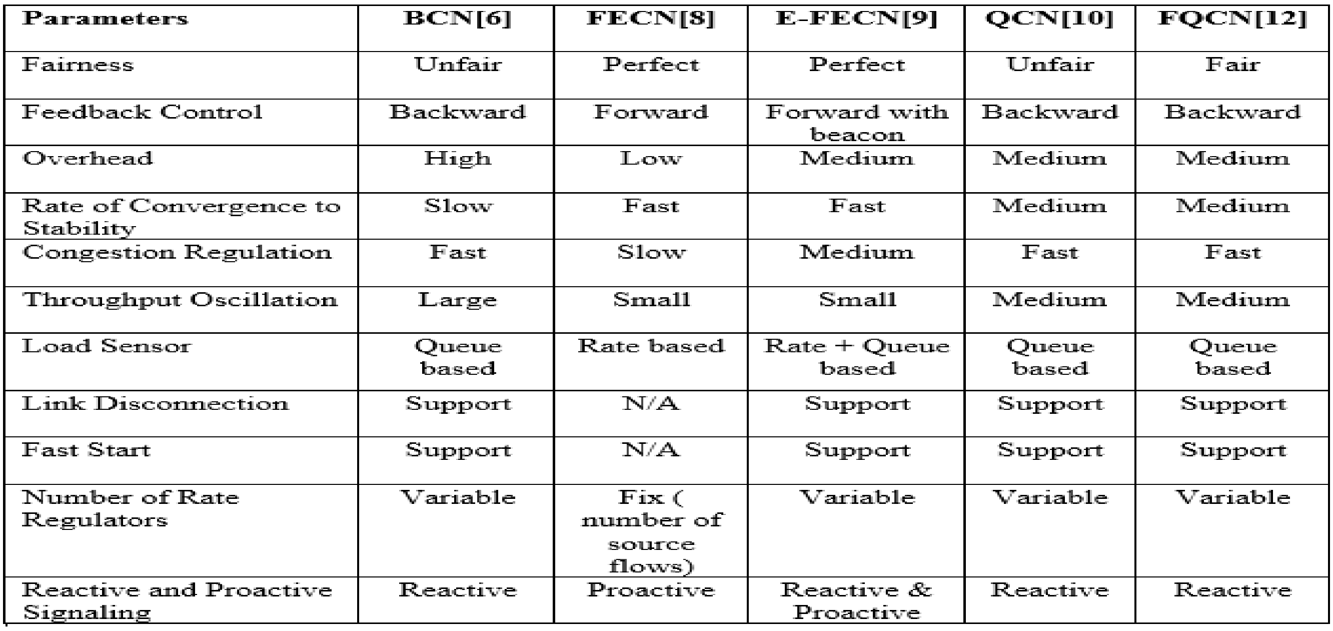
The sources are initialized with the full rate in BCN, QCN and FQCN and eventually ramp down if any negative feedback is received from a switch. In FECN, the sources are initially at lower rate and move to some threshold value, as successive probes get returned [5].

3.10 Number of Rate Regulators:

FECN requires regulators equal to the number of concurrent flows. Number of rate regulators in E-FECN varies due to the adaption of BCN000. In BCN, QCN and FQCN, the feedback message is only sent to the source of the sampled packet and therefore the number of rate regulators in them are varying in nature [1].

3.11 Reactive and Proactive Signaling:

BCN, QCN and FQCN use reactive signaling while FECN uses proactive signaling and E-FECN employs both these signaling methods. In proactive signaling probes are sent periodically, so at least one periodic interval is needed to respond to the sudden overload [12] advantages and disadvantages of the major. This paper will useful in getting overall view of the congestion notification algorithm along with difference in their behavior



Parameters	BCN[6]	FECN[8]	E-FECN[9]	QCN[10]	FQCN[12]
Fairness	Unfair	Perfect	Perfect	Unfair	Fair
Feedback Control	Backward	Forward	Forward with beacon	Backward	Backward
Overhead	High	Low	Medium	Medium	Medium
Rate of Convergence to Stability	Slow	Fast	Fast	Medium	Medium
Congestion Regulation	Fast	Slow	Medium	Fast	Fast
Throughput Oscillation	Large	Small	Small	Medium	Medium
Load Sensor	Queue based	Rate based	Rate + Queue based	Queue based	Queue based
Link Disconnection	Support	N/A	Support	Support	Support
Fast Start	Support	N/A	Support	Support	Support
Number of Rate Regulators	Variable	Fix (number of source flows)	Variable	Variable	Variable
Reactive and Proactive Signaling	Reactive	Proactive	Reactive & Proactive	Reactive	Reactive

Table 1: A COMPARISON OF BCN, FECN, E-FECN, QCN AND FQCN

4. CONCLUSIONS

With rapid deployment of modern high-speed, low-latency, large-scale data centers, many problems have been observed in data centers [14]. In this paper, the Ethernet layer congestion control issues in data centers have been studied. We studied the drawback and how it is been overcome by the different algorithms. In this survey paper, we have studied the most recent research activities in DCNs. We have discussed and compared some of the recently proposed congestion notification algorithm. In addition, we have also summarized the pros and cons of different algorithms.

ACKNOWLEDGEMENT

We express our profound gratitude to P. T. Kulkarni, Principal, Pune Institute of Computer Technology. We express our deep felt gratitude to Prof. G.P. Potdar, Head of the Department, Computer Engineering for his constant encouragement.

REFERENCES

[1] Prajjwal Devkota, A. L. Narasimha Reddy "Performance of Quantized Congestion Notification in TCP Incast Scenarios of Data Centers" 18th Annual IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, Vol.235-243,2010.

[2] Yan Zhang, Nirwan Ansari, "On Architecture Design, Congestion Notification, TCP Incast and Power Consumption in Data Centers" IEEE COMMUNICATIONS SURVEYS.

[3] Jinjing Jiang and Raj Jain, "Analysis of Backward Congestion Notification (BCN) for Ethernet In Datacenter Applications"in IEEE 802.1 Meeting, September 2005..

[4] D. Bergamasco,"Ethernet Congestion Manager,"in IEEE 802.1Qau Meeting,March 13 2007.

[5] Yan Zhang, Nirwan Ansari, "Fair Quantized Congestion Notification in Data Center Networks," IEEE transactions on communications,, vol. 61, no. 11, November 2013.

[6] M. Alizadeh, B. Atikoglu, A. Kabbani, A. Lakshmikantha, R. Pan, B. Prabhakar, and M. Seaman, "Data Center TransportMechanisms: Congestion Control Theory and IEEE Standardization, in Proc. 46th Annual Allerton Conference on Communication, Control, and Computing, Allerton House,UIUC, Illinois, pp. 12701277, Sep. 2008.

[7] Thor Olavsrud, "A Study of Fair Bandwidth Sharing with AIMD-Based Multipath Congestion Control," IEEE WIRELESS COMMUNICATIONS LETTERS, vol. 2, no.3, June 2013.

[8] J. Jiang, R. Jain and C. So-In, "An Explicit Rate Control Framework for Lossless Etherent Operation," in: in Proc. ICC, Beijing, China, May 19-23,,pp. 5914 5918, 2008.

[9] C. So-In, R. Jain and J. Jiang, "Enhanced Forward Explicit Congestion Notification (E-FECN) Scheme for Datacenter Ethernet Networks, in Proc. Symposium on Performance Evaluation of Computer and Telecommunication Systems, Edinburgh, UK,, pp. 542-546, Jun. 2008.

[10] Wanchun Jiang, Fengyuan Ren, Chuang Lin, Ivan Stojmenovic, "Analysis of Backward Congestion Notification with Delay for Enhanced Ethernet Networks," IEEE TRANSACTIONS ON COMPUTERS,2015.

[11] Abdul Kabbani, Mohammad Alizadeh, Masato Yasuda y, Rong Panz, and Balaji Prabhakar "AF-QCN: Approximate Fairness with Quantized Congestion Notification for Multitenanted Data Centers".

[12] Dizhi Zhou, Wei Song, Yu Cheng, "A Study of Fair Bandwidth Sharing with AIMD-Based Multipath CongestionControl,"IEEE WIRELESS COMMUNICATIONS LETTERS, VOL. 2, NO. 3, JUNE 2013.

[13] Yan Zhang, Nirwan Ansari, "On Mitigating TCP Incast in Data Center Networks," IEEE INFOCOM, 2011.

[14] MinJi Kim, Jason Cloud, Ali ParandehGheibi, Leonardo Urbina, Kerim Fouli, Douglas J. Leith, and Muriel Mdard, "Congestion Control for Coded Transport Layers," IEEE ICC - Communication QoS, Reliability and Modeling Symposium, 2014.