A Survey on congestion control with TCP network

Tarannum Khan[1], Sachin Upadhyay[2], Karan Dharni[3]

[1]Student Dept. of Computer science and engineering, Alpine Institute of Technology, Ujjain(India)
[2]Assistant professor Dept. of Computer science and engineering, Alpine Institute of Technology, Ujjain(India)
[3]Student Dept. of Computer science and engineering, Jaypee University of engineering and technology, Guna(India)

Abstract - *Congestion Control performs a completely important role in Computer Networks, Modern Telecommunication, Internet and each wired and wireless communications are being meant for excessive-velocity conversation of huge quantities of records. Due to lack of right Congestion control mechanism the congestion collapse of such networks would emerge as especially complex. A network with Streamed media traffic is a challenge for Congestion manages because of sensitivity. In this paper, a survey on diverse mechanisms of congestion control and avoidance has been completed.*

Keyword- Congestion Control, classification of Congestion Control, window based approach etc.

1. Introduction

The telecommunication industry started with a wired connection and then the latest advancement in exceptional technology has made it viable to communicate using wireless technologies. Invention of computer, software, hardware, microchips has changed the whole concept of communication. Today we see a blend of stressed out and wireless verbal exchange network using heterogeneous technology. The heterogeneity in communication networks has no longer handiest opened the door of various codes of conversation. Congestion is one of the fundamental troubles among them. A modern communiqué network is built the usage of some of properly-linked gadgets or nodes that have limited nearby capacity and resources. Currently, two special delivery paradigms are used [1] circuit switched transport and [2] packet switched transport.

Congestions an unexpected state of this communication network in which one or greater nodes reach their capacity restriction and as a end result they drop the approaching packet or support them for a later transmission, Induces delay in the presence of packets at the receiver. None of these results of congestion are ideal for media transport hence counter measures should be taken to prevent the occurrence of congestion in communication network.

2. Classification of congestion control schemes and their review

Congestion controls schemes can be categorized with appreciate to a mess of traits. In the subsequent we in short discuss diverse viable classification schemes for TCP-pleasant methods.

Window-Based vs. Rate-Based

One possible category criterion for TCP- pleasant schemes is whether or not they adapt their supplied network load primarily based on a congestion window or on their transmission rate.

Algorithms that have a place with the window-basically based class utilize congestion. Like TCP, each packet transmitted devours one space in the congestion window. The size of the congestion window is elevated inside the absence of congestion indications and reduced while congestion happens.

Rate-based totally congestion control achieves TCP friendliness via dynamically adapting the transmission rate in line with a few network remarks system that demonstrates congestion. It can be subdivided into Simple AIMD plans and form version-primarily congestion control. Straightforward AIMD plans emulate the conduct of TCP congestion control. This consequence in a price that displays the standard brief-time period noticed enamel-like behavior of TCP. This makes simple AIMD schemes fallacious for non-stop media streams. Model-based congestion control uses a TCP version such with the aid of adapting the sending rate to the common long term throughput of TCP, model-based congestion control can produce much smoother rate changes that are better suitable for the traffic which has been discussed in such schemes do not mimic TCP's short-term duration. However, the congestion control mechanism won't resemble TCP congestion control and great attention must be paid to the fee adjustment mechanism to ensure some form of opposition amongst TCP.

Unicast vs. Multicast

The design of suitable multicast congestion control protocols is some distance greater tough than the layout of unicast protocols. Multicast congestion control schemes preferably must scale to huge receiver sets and be able to cope with heterogeneous network conditions on the receivers. For example, if for all receivers the sender transmits packets at the same rate, the care has to be taken so that the transfer rate do not decrease when there is congestion in the network. This problem is known as the *loss path* multiplicity *problem* [3]. Whenever rate adjustment decisions are based not on congestion information from a specific receiver but on the overall congestion information present in the whole distribution tree, protocol performance may degrade in a drastic manner if there is some lacuna in the designing of the protocol.

Single-Rate vs. Multi-rate

A not unusual criterion for classifying TCP-friendly multicast congestion control protocols is on the basis of the fee on which they operate. This may be single rate or the multi rate. The unicast transport protocols are related to single rate schemes. In single-rate schemes the data is transferred to all the receivers at the same rate. This limits the scalability of the mechanism, since all receivers are restricted to the rate that is TCP-friendly for the bottleneck receiver.

Multi-rate congestion control protocols allow for a more flexible allocation of bandwidth for the many paths existing along the network. Such schemes scale better to large receiver sets when there is a possibility of the dissimilar or heterogeneous networks. A regular method to multi- rate congestion manages is to apply layered multicast: a sender divides the facts into several layers and transmits them to one-of-a-kind multicast groups. Each receiver can individually select to join as many groups as permitted by the threshold, which is the bandwidth bottleneck of the sender and the receiver. The quality of a receiver increases when it joins more number of groups. For a video transmission an increased number of received layers may improve the video quality, while for reliable bulk data transfer additional layers may decrease the transfer time.

In multicast, the group management & the routing mechanism deals with the congestion control activity in the indirect manner. In order for this mechanism to be effective, it is crucial to coordinate join and leave decisions of receivers behind a common bottleneck: if only some receivers leave a layer while others stay subscribed, no pruning is possible and congestion cannot be reduced. In addition, receivers do now not make efficient use of the multicast layers whilst they're no longer subscribed to a layer this is already present in their subpart of the routing tree. There data receiving rate could be increased without paying any extra cost. Therefore, receivers that share a bottleneck link should synchronize their decisions to join and leave layers. The *leave* latency is another issue of concern: pruning of the multicast tree upon receipt of leave messages for a layer can take considerable time, on the order of several seconds.

End-to-End vs. Router-Supported

Many of the TCP-friendly schemes proposed are designed for best effort IP networks that do not provide any additional router mechanisms to support protocol. So they can be easily deployable on the internet. These schemes are called *end-to-end* congestion control.. They may be in addition separated into *sender-based* and *receiver*-based totally strategies.

In sender-based approaches the sender uses information about the network congestion and then it adjust the window size to deal with such situation. The receivers only provide feedback, while the responsibility of adjusting the rate lies solely with the sender. Receiver-driven congestion control is usually used together with layered congestion control approaches. Here, the receivers decide whether to subscribe or unsubscribe from additional layers based on the congestion situation of the network.

The design of congestion control protocols and particularly fair sharing of resources can be considerably facilitated by placing intelligence in the network. Router supported congestion control schemes rely on additional functionality in the network. Particularly multicast protocols can benefit from additional network functionality together with comments. aggregation, hierarchical RTT measurements control of partnerships of receivers, or alteration of the routers' lining methodologies Generic router assist (GRA) as an instance, is a recent initiative that proposes trendy mechanisms positioned at routers to assist transport control protocols, which would greatly ease the design and implementation of effective congestion control protocols.

Furthermore, end-to-end congestion control has the disadvantage of relying on the collaboration of the end systems. Experience in the current Internet has shown that this cannot always be assumed: greedy users or applications may use non TCP to obtain bigger bandwidth. As discussed by Floyd and Fall in [1], Some shape of congestion control have to be enforced with the aid of routers so that it will save you congestion collapse. The authors present router mechanisms to identify flows that should be regulated: for instance, when a router discovers a flow which does not exhibit TCP-friendly conduct, the router may drop the packets. While ultimately fair sharing of resources in the presence of unresponsive or non-TCP-friendly flows can only be achieved with router support, this mechanism is difficult to deploy, since changes to the Internet infrastructure take time and are costly in terms of money and effort.

Classification Scheme

The classification distinguishes between single-rate and multi-rate congestion control at the top level and rate-based vs. Window-based congestion control.

Single-Rate Congestion Control Protocols

In this segment, single-rate congestion control protocols are specified. A more whole review may be found within the corresponding technical report.

Rate-Based Approaches

Many rate-based congestion control protocols mimic TCP's AIMD behavior to achieve TCP fairness, while others implicitly or explicitly adjust their rate according to a model of TCP traffic. A very obvious approach to TCP-friendly congestion control is to directly apply TCP's congestion control mechanism, but without the associated reliability mechanism. Early work in this area was presented in [4]. In that work this approach is used to adjust the rate of a unicast video stream to adjust with the congestion.

RAP — The Rate Adaption Protocol (RAP) presented in [5] is a simple AIMD scheme for unicast flows. Each data packet is acknowledged by the receiver. The ACKs are utilized to identify packet loss and construe the RTT. In intervals without congestion, the sending rate increases via 1 packet/RTT for this reason mimicking the AIMD nature of the transmission control protocol. To provide additional fine-grained delay-based congestion avoidance, the ratio of a short-term RTT average and a long-term RTT average is used to modify the inter packet gap between consecutive data packets. These fine-grained rate adjustments result in a smoother sending rate.

RAP achieves rates similar to TCP in an environment where TCP experiences no or few timeouts since RAP's rate reductions resemble TCP's reaction to triple duplicate ACKs. However the RAP does not take timeouts into account and is therefore more aggressive when TCP's throughput is dominated by timeout events.

LDA+ — unlike many of the other schemes, the Loss-Delay Based Adaption Algorithm (LDA+) does not devise its own feedback mechanism to control the sending rate however is based entirely on the Real-Time Transport Control Protocol (RTCP) feedback messages furnished by way of the Real-Time Transport Protocol (RTP) [6]. While LDA+ is essentially an AIMD congestion control scheme, it uses some interesting additional elements. The increase and decrease factors for AIMD are dynamically adjusted to the network conditions.

The amount of additive increase is then determined as the minimum of three factors, these three factors are as follows. Flows with a low bandwidth can increase their rate faster than flows with a higher bandwidth. Flows do not exceed the estimated bottleneck bandwidth. Flows do not increase their bandwidth faster than a TCP connection.

TFRC — The TCP-Friendly Rate Control Protocol (TFRC) [7] evolved from the TFRCP protocol. It is specified for unicast communication. Similar to TFRCP it is also associated with the sending rate. Several requirements for a loss rate estimator are formulated, and the authors agree on average loss interval method, the average loss interval method satisfies these requirements. The loss rate is measured in terms of loss intervals, spanning the number of packets between consecutive loss events. The authors provide additional mechanisms to prevent the loss rate from reacting too strongly to single loss events. The RTT is measured by the standard method of feeding back timestamps to the sender.

The sender then registers another reasonable rate from these parameters and changes the sending rate as needs be. To improve protocol performance in environments that do not fulfill the assumptions of the complex TCP equation, TFRC supports additional delay-based congestion avoidance by adjusting the inter packet gap (i.e., the time interval between consecutive data packets).

A major advantage of TFRC is that is has a relatively stable sending rate along with providing the sufficient challenge to the competitors. TEAR — TCP Emulation at Receivers (TEAR) [8] is a hybrid protocol that mixes components of window-based and fee primarily based congestion control. TEAR receivers calculate a fair receive rate which is sent back to the sender, who then adjusts the sending rate. To this quit, the receivers hold a congestion window. Since TCP's congestion window is placed on the sender, a TEAR receiver has to try to determine from the arriving packets whilst TCP could increase or lower the congestion window size. Additive increase and window reductions caused by triple duplicate ACKs are easy to emulate. Be that as it may, because of the absence of ACKs, timeout occasions can be assessed just generally.

Window based approaches

There are two main problems that have to be solved in order to use window-based congestion control for multicast. First, protocols should prevent drop-to-zero rate because of the path multiplicity problem. The other problem is related to the free slots in the window. Clearly it is not possible for the sender to receive ACKs for each packet from each receiver, since this would cause an ACK implosion.

In the following we will present several window-based congestion control approaches for multicast transmission. Specifically, we will concentrate on how these two previously mentioned issues will be tackled.

A Framework for Window-Based Congestion Control — Golestani and Sabnani propose to use a window-based approach where each receiver keeps a congestion window for each as in TCP. From the size of the window and the number of outstanding packets, each receiver calculates the highest sequence number it is able to receive without claiming an unfair amount of bandwidth. This information needs to be communicated to the sender without causing a feedback implosion. As an example of how this can be done, the authors show that a tree structure formed by the receivers or other intermediate systems can be used to aggregate the information: each node takes the minimum sequence number contained in all incoming messages and forwards this sequence number to its parent. When the aggregated information reaches the sender then it is allowed to send the packets. Every recipient keeps up its own congestion window, which circumvents the loss path multiplicity issue.

The observations made by Golestani and Sabnani form a theoretical background for window-based multicast congestion control. They need to be concretized by some actual algorithms, such algorithms are as follows:

RLA and LPR — The Random Listening Algorithm (RLA) proposed by Wang and Schwartz extends TCP selective ACK (SACK) by introducing some enhancements for multicast. Based on these loss indications the number of receiver's nwith a high congestion probability is tracked.

If there is any congestion then the window in divided in to the two parts, one is If the previous window cut was made too long ago (the authors propose an interval of twice the moving average of the window size times the smoothed RTT of the corresponding receiver) and other is If a generated uniform random number p is less than or equal to $1/n$.

MTCP — Multicast TCP (MTCP) [9] is a reliable multicast protocol. MTCP groups the session participants into a logical tree structure where the root of the tree is the sender of the data. A parent in the logical tree structure stores a received packet. Upon receiving a packet, a child (which may be a parent for other participants) transmits an ACK to its parent utilizing unicast.

NCA and pgmcc — Nominee-Based Congestion Avoidance (NCA) presented in [10] and pragmatic general multicast congestion control (pgmcc) [11] are two approaches to congestion control that share the same fundamental idea: they select as a group representative the bottleneck receiver with the worst network connection. This receiver acknowledges every packet received and therefore it allows TCP cycle algorithm to be used by the sender. It is important to note that in this approach congestion control and packet repair are treated independent of each other. Thus the approach can be used in combination with a large number.

The most challenging aspect of NCA and pgmcc is how to select the group representative. In both approaches, each receiver calculates the data rate. The information about the acceptable rate is conveyed back to the sender either piggybacked on NACKs (pgmcc) or accumulated in a tree structure. From those reports the sender selects as the representative the participant with the lowest acceptable rate and uses a TCP-like congestion control mechanism to this participant.

This approach seems very promising, since it closely mimics the behavior of unicast TCP and therefore should lead to fairness with regard to TCP flows if the proper representative is chosen. The author of pgmcc indicates that this may occur when a set of receivers has lossy links with a low RTT.

Multi rate congestion control protocols

In this segment, Multi rate congestion control protocols are specified.

Rate-Based Approaches

One of the first working examples of layered multicast transmission in the Internet was Receiver-Driven Layered Multicast (RLM) for the transmission of video [12]. Their work did not focus on TCP friendliness but on how to provide each receiver with the best possible video quality in dependence on the bandwidth available between the sender and that receiver. When the receiver does not experience congestion in the form of packet loss for a certain period of time, it subscribes to the next layer.

The use of RLM to control congestion is problematic since RLM's mechanism of adding or dropping a single layer based on the detection of packet loss is not TCP-friendly and can result in an unfair distribution of bandwidth among concurrent RLM sessions. Failed join experiments (i.e., a receiver joining a layer immediately has to leave again because the necessary bandwidth is not available) are therefore very costly in terms of the additional congestion they may cause. As mentioned earlier, in order for layered schemes to be efficient.

RLC — Vicisano, Crowcroft, and Rizzo address most of these problems are based on the RLC protocol [13]. They propose to dimension the layers so that the bandwidth consumed by each new layer increases exponentially. The time a receiver has to wait before joining a new layer. On the other hand, a layer is dropped immediately when congestion results in packet loss. This emulates the behavior of TCP since the increase in bandwidth is proportional to the amount of time required to pass without losing any packet. At the same time the reaction to congestion is a multiplicative decrease, since dropping one layer results decreasing the average rate

FLID-DL — To address some of the deficiencies of RLC, Byers *et al.* propose Fair layered protocol [14]. This protocol uses a fountain [15]. With digital fountain encoding, the sender encodes the original data and redundancy information such that receivers can decode the original data once they have received a fixed number of arbitrary but distinct packets. Since it is not necessary to ensure delivery of specific packets, this layering scheme is more reliable.

FLID-DL presents the idea of dynamic layering to decrease the join. The receive rate is reduced simply by not joining additional layers, whereas rate increase requires joining multiple layers. To reduce the total number of layers required by the mechanism, layers are reused after a quiet period where no data has been transmitted over the time. This scheme provides an elegant solution to avoiding the effect of long leave latencies, provided that it is sufficient for normal leave operations to take effect.

LTS and TFRP — Two similar congestion control protocols for the transmission of streams are Transmission Scheme (LTS) [16] and the TCP-Friendly Transport Protocol (TFRP) [25] both refrain from join experiments to probe for available bandwidth, using instead the simple TCP Eq. 1 to adjust the rate. Tan and Zakhor do not address the problem of how to measure the RTTs to the receivers in a scalable way. In LTS, the RTTs are measured simply by having the receivers send *RTT request messages* to the sender. To prevent rate oscillations, it is necessary to accurately measure and smooth loss and RTT values through filtering. MLDA — The Multicast Loss-Delay Based Adaption Algorithm (MLDA) [17]. it is a protocol for controlling congestion. It builds on the previously discussed LDA+ protocol, also using RTCP reports for the signaling between the sender and the receivers. MLDA retains the increase and decrease behavior of LDA+. The receivers report the rate to the sender, avoiding feedback implosion by using exponentially distributed timers. The sender continuously adjusts the bandwidth distribution of the layers to support the reported rates. Thus, MLDA combines the two concepts of sender- and receiver-based congestion control. To calculate the rate, the RTT has to be measured at the receivers. At certain points in time, a receiver measures the RTT using the well-known scheme. This accurate measurement is then continuously modified using the one-way delay between the sender and the receiver.

Window-Based Approaches

Rainbow — Rainbow [18] is a window-based congestion control scheme for the reliable transfer of bulk data. Like FLIDDL, the data is encoded using a digital fountain. Thus, it is not important what specific packets a receiver gets. The key idea of Rainbow is that receivers individually request the transmission of each data packet. Each receiver keeps a congestion window, and each request is marked with a label that essentially indicates the position of the request in the congestion window.

MANET is characterized by using its very dynamic topology due to the mobility of its wireless nodes. Such a completely bendy and infrastructure-much less network serves a huge variety of programs including rescue missions, military operations, and in some other situation. Retransmitting a misplaced packet through fast retransmit mechanism is followed through many TCP variations, inclusive of TCP NewReno, and is accompanied through invoking fast healing mechanism [19].

MANET is described by its extremely unique topology because of the mobility of its wireless nodes. Such a completely bendy and infrastructure-less network serves a wide variety of programs which includes rescue missions, army operations, and in some other situation. where infrastructure establishment is either very expensive or quite impossible [20].

A transport mechanism that is adapted to video flows was presented. It is called Q-AIMD for video quality AIMD. Q-AIMD permits fairness in video quality whilst transmitting a couple of video flows and improves the overall video best for all flows, mainly whilst the transmitted videos offer numerous kinds of content material with extraordinary spatial resolutions. Q-AIMD lessens congestion events through decreasing the video quality, and therefore the bit fee, every time congestion occurs. Q-AIMD has been evaluated with exclusive video contents and spatial resolutions and has proven a progressed general video fine as compared to different throughput-based totally congestion control mechanism.

In various factors consisting of the manner CCN routers are deployed, the recognition of contents, or the potential of hyperlinks have been taken into consideration in looking at the overall performance of AIMD when used over a CCN network. Results endorse the want to layout a proper congestion control to avoid much less famous contents from becoming hardly ever reachable in the tomorrow's Internet [21].

In [22] a system for AIMD in TCP has been propelled to examine issues of adjusting TCP AIMD algorithms over wireless networks. The framework offers a scientific analysis of existing AIMD- based TCP variants, classifies them into two predominant streams, and develops a generic expression that covers the rate adaptation strategies of both approaches. It similarly identifies a new approach in improving the performance of TCP and assists in the design of latest TCP variations. A tax-rebate approach was proposed as an approximation of the repayment scheme, and used to enhance the AIMD-primarily based TCP variations to provide unified answers for viable congestion control, sequencing control, and blunder control.

TCP Karak does not depend on any unequivocal criticism from the network center; it requires just the sender side adaptation. TCP Karak also works on top of any underlying routing protocol such as AODV [23].

3. Literature survey

Sebastian Kuhlmorgen, et al. [2017] this paper studies the performance of the gatekeeper with packet prioritization and an adaptive linear control algorithm. The simulation outcomes imply that the gatekeeper with priority queuing (PQ) can effectively take care of distinct packet priorities for multi-hop packets. Our gatekeeper-specific enhancements of the forwarding algorithm yield performance improvements in terms of reliability and latency compared to the plain DCC approach. Finally, we speak the issue of packet starvation resulting from the gatekeeper's PQ scheme that influences the performance of lower -priority packets [24].

Bo HU, et al. [2017] In this paper, proposes a Congestion Control method with Fairness (CCMF) for multipath transport. CCMF transfers the traffic among different paths and considers the bandwidth fairness when multipath flows share the same bottleneck with single-path flows. We introduce a parameter of F-factor to reflect the degree and variation trend of fairness. Thus, CCMF can regulate the sending rate according to the fairness factor F-factor and achieve end-to-end fairness transmission. The simulation results show that CCMF can improve the fairness degree compared with Coupled Algorithm [25].

Min Xiao et al. [2017] in this paper, proposes a delayed fractional-order congestion control version that is extra correct than the unique integer-order version whilst depicting the dual congestion control algorithms. The presence of fractional orders calls for the use of appropriate criteria which typically make the analytical work so harder. In view of the security theorems on not on time fractional request differential conditions, we observe the trouble of the stability and bifurcations for any such model with the aid of deciding on the verbal exchange delay because the bifurcation parameter. By studying the related function equation, some express situations for the nearby balance of the equilibrium are given for the behind schedule fractional order version of congestion manage algorithms. Moreover, the Hopf bifurcation conditions for fashionable delayed fractional-order structures are proposed. The presence of Hopf bifurcations on the balance is snared. The crucial values of the delay are recognized, wherein the Hopf bifurcations arise and a own family of oscillations bifurcate from the equilibrium [26].

Chengxi Gao et al. [2017] In this paper proposes DemePro, a DCN scheme for different line of situation. In face of congestion signal, DemePro also leverages ECN for congestion notification, whilst decouples packet marking from enqueuing, with a purpose to make certain fairness amongst a couple of offerings. We additionally query the effectiveness of the congestion signal derived from the single threshold for port buffer queue duration, via a fixed of experiments. Then, DemePro makes use of more than one thresholds for proactive congestion control, and packets are encapsulated to carry congestion quantity records, which will notify TCP senders for particular and excellent-grained congestion control. Experiments display that DemePro has higher overall performance than MQ-ECN [1] that is currently the nice DCN scheme for more than one-provider a couple of-queue situation, and DemePro can also well guarantee the fairness [27].

Rohan More et al. [2016] this paper proposes the significance of Jordan sequential network for prediction of future values,depending upon the current value and aggregate past values and also guarantees prediction of traffic flow with accuracy of about 92-98% using Jordan's Sequential Network. Thus, this paper focuses on prediction of traffic flow using Jordan's Neural network with maximum accuracy and analysis on various parameters to obtain the same [28].

Ramandeep Dhanoa et al. [2016] In this paper proposes a multistep Neural Network Prediction-based Routing (NNPR) protocol to predict as well as control network traffic before congestion actually happens. A distributed real time transaction processing simulator serves as the test-bed and a cloud-based scoring engine has been used to obtain results in real time; messages are then rerouted to prevent congestion. Various parameters which can cause congestion are studied. These include bandwidth, work size, latency, max active transactions, mean arrival time and update percentage. The performance of proposed protocol is compared with existing protocols. Through experimentation, it is demonstrated that NNPR consistently provides superior performance for all congestion loads [29].

4. Conclusion

This paper presents the have a look at of congestion control and elaborates various troubles related with it. We in short survey of various congestion control algorithms. It shows that at present there is no single algorithm that can resolve every problems of congestion control on computer networks. Further research work is needed in this direction.

References

[1] S. Floyd and K. Fall, "Promoting the Use of End-to-end Congestion Control in the Internet," IEEE/ACM Trans. Net., vol. 7, no. 4, Aug. 1999, pp. 458–72.

[2] J. Padhye et al., "Modeling TCP Reno Performance: A Simple Model and Its Empirical Validation," IEEE/ACM Trans. Net., vol. 8, no. 2, Apr. 2000, pp. 133–45.

[3] S. Bhattacharyya, D. Towsley, and J. Kurose, "The Loss Path Multiplicity Problem in Multicast Congestion Control," Proc. IEEE INFOCOM, New York, NY, Mar. 1999, vol. 2, pp. 856–63. .

[4] S. Jacobs and A. Eleftheriadis, "Providing Video Services over Networks Without Quality of Service Guarantees," W3C Wksp. Real-Time Multimedia and the Web, Oct. 1996.

[5] R. Rejaie, M. Handley, and D. Estrin, "Rap: An End-to-End Rate-Based Congestion Control Mechanism for Realtime

Streams in the Internet," Proc. IEEE INFOCOM, Mar. 1999.

[6] H. Schulzrinne et al., "Rtp: A Transport Protocol for Real-time Applications," RFC 1889, Jan. 1996.

[7] S. Floyd et al., "Equation-based Congestion Control for Unicast Applications," Proc. ACM SIGCOMM, Stockholm, Sweden, Aug. 2000, pp. 43–56

[8] I. Rhee, V. Ozdemir, and Y. Yi, "TEAR: TCP Emulation at Receivers – Flow Control for Multimedia Streaming," Tech. rep., Dept. of Comp. Sci., NCSU, Apr. 2000

[9] I. Rhee, N. Balaguru, and G. Rouskas, "MTCP: Scalable TCP-Like Congestion Control for Reliable Multicast," Proc. IEEE INFOCOM, Mar. 1999, vol. 3, pp. 1265–73.

[10] S. Kasera et al., "Scalable Fair Reliable Mulitcast Using Active Services," IEEE Net. (Special Issue on Multicast), vol. 14, no. 1, Jan./Feb. 2000, pp. 48–57.

[11] L. Rizzo, "Pgmcc: A TCP-friendly single-rate Multicast Congestion Control Scheme," Proc. ACM SIGCOMM, Stockholm, Sweden, Aug. 2000, pp. 17–28

[12] S. McCanne, V. Jacobson, and M. Vetterli, "Receiver-driven Layered Multicast," Proc. ACM SIGCOMM, Palo Alto, CA, Aug. 1996, pp. 117–30

[13] L. Vicisano, J. Crowcroft, and L. Rizzo, "TCP-like Congestion Control for Layered Multicast Data Transfer," Proc. IEEE INFOCOM, Mar. 1998, vol. 3, pp. 996–1003.

[14] J. Byers et al., "FLID-DL: Congestion Control for Layered Multicast," Proc. 2nd Int'l Wkshp. Networked Group Commun., Palo Alto, CA, Nov. 2000.

[15] J. Byers et al., "A Digital Fountain Approach to Reliable Distribution of Bulk Data," Proc. ACM SIGCOMM '98, Sept. 1998.

[16] T. Turletti, S. Parisis, and J. Bolot, "Experiments with a Layered Transmission Scheme over the Internet," Tech. rep. RR-3296, INRIA, France, Nov. 1997.

[17] D. Sisalem and A. Wolisz, "MLDA: A TCP-friendly Congestion Control Framework for HeterogenousMulticast Environments," 8th Int'l. Wksp. QoS, June 2000.

[18] K. Yano and S. McCanne, "A Window-based Congestion Control for Reliable Multicast Based on TCP Dynamics," Proc. ACM Multimedia, Oct. 2000. [30] Almobaideen, W., Al-Soub, R. and Sleit, A. (2013) MSDM: Maximally Spatial Disjoint Multipath Routing Protocol for MANET. Communications and Networks, 5, 316-322

[19] Tuan Tran Thai, Changuel, N., Kerboeuf, S., Faucheux, F., Lochin, E. and Lacan, J. (2013) Q-AIMD: A Congestion Aware Video Quality Control Mechanism. 20th International Workshop on Packet Video (PV), 1, 12-13

[20] Sreenivas, B., Bhanu Prakash, G. and Ramakrishnan, K. (2012) An Adaptive Congestion Control Technique for Improving TCP Performance over Ad-Hoc Networks. Elixir International Journal, 44, 7391-7395.

[21] Lai, C., Leung, K.C. and Li, V.O.K. (2013) Design and Analysis of TCP AIMD in Wireless Networks. 2013 IEEE Wireless Communications and Networking Conference (WCNC), 1422-1427.

[22] Almobaideen, W., Al-Khateeb, D., Sleit, A., Qatawneh, M. Al-Khdour, R. and Abu Hafeeza, H. (2013) Improved Stability Based Partially Disjoint AOMDV. International Journal of Communications, Networks, and System Sciences, 6, 244 250.