Congestion Control for Packet Switched Networks: A Survey

C.Socrates[*], P.M.Beulah Devamalar[**], R.Kannamma Sridharan[***]

[*]PG Scholar, PITAM, Anna University, Chennai, Tamil Nadu.
[**]Principal, PITAM, Anna University, Chennai, Tamil Nadu.
[**]Assistant Professor, PITAM, Anna University, Chennai, Tamil Nadu.

Abstract- Congestion has been considered as one of the basic important issue in packet switched network [1][9]. Congestion control refers to the mechanisms and techniques to control the congestion and keep the load below the capacity. This paper provides an overview of category provided by congestion control .It also includes how TCP uses congestion control to avoid congestion or alleviate congestion in network. Computer networks have experienced an explosive growth over the past few years and with that growth have come severe congestion problems. This paper also concentrates on avoidance of congestion. This scheme allows a network to operate in the region of low delay and high throughput. In this paper, a survey on various mechanisms of congestion control and avoidance has been done.

Index Terms- Congestion Control, Transmission Control Protocol, Active Queue Management, Congestion Avoidance.

I. INTRODUCTION

Congestion in a network may occur if the load on the network-the number of packets sent to the network is greater than the capacity of the network-the number of packets a network can handle..Network congestion occurs when a link or node is carrying so much data that its quality of service deteriorates. Typical effects include queuing, packet loss or the blocking of new connections. Congestion control is a method used for monitoring the process of regulating the total amount of data entering the network .so as to keep traffic levels at an acceptable value. This is done in order to avoid the telecommunication network reaching what is termed congestive collapse. Modern networks use congestion control and network congestion avoidance techniques to try to avoid congestion collapse. These include: exponential back off in protocols such as 802.11's CSMA/CA and the original Ethernet, window reduction in TCP, and fair queuing in devices such as routers Congestion avoidance techniques monitor network traffic loads in an effort to anticipate and avoid congestion a common network bottleneck.

II. EFFECTS OF CONGESTION

Congestion affects two vital parameters of the network performance, namely *throughput* and *delay*. The throughput can be defined as the percentage utilization of the network capacity. Throughput is affected as offered load increases. Initially throughput increases linearly with offered load, because utilization of the network increases. However, as the offered load increases beyond certain limit, say 60% of the capacity of the network, the throughput drops. If the offered load increases further, a point is reached when not a single packet is delivered to any destination, which is commonly known as *deadlock* situation. The ideal one corresponds to the situation when all the packet introduced are delivered to their destination up to the maximum capacity of the network. The second one corresponds to the situation when there is no congestion control. The third one is the case when some congestion control technique is used. This prevents the throughput collapse, but provides lesser throughput than the ideal condition due to overhead of the congestion control technique.

III. CONGESTION CONTROL MECHANISMS

Congestion control mostly applies to packet-switching network. A wide variety of approaches have been proposed, however the "objective is to maintain the number of packets within the network below the level at which performance falls off dramatically.

3.1TRANSMISSION CONTROL PROTOCOL (TCP):

There has been some serious discussion given to the potential of a large-scale Internet collapse due to network overload or congestion [2]. So far the Internet has survived, but there have been a number of incidents throughout the years where serious problems have disabled large parts of the network. Some of these incidents have been a result of algorithms used or not used in the Transmission Control Protocol (TCP) .The popularity of the Internet has heightened the need for more bandwidth throughout all tiers of the network. Home users need more bandwidth than the traditional 64Kb/s channel a telephone provider typically allows. Video, music, games, file sharing and browsing the web requires more and more bandwidth to avoid the “World Wide Wait” as it has come to be known by those with slower and often heavily congested connections. Internet Service Providers (ISPs) who provide the access to the average home customer have had to keep up as more and more users get connected to the information superhighway. Core backbone providers have had to ramp up their infrastructure to support the increasing demand from their customers below. Today it would be unusual to find someone in the U.S. that has not heard of the Internet, let alone experienced it in one form or another. The Internet has become the fastest growing technology of all time . So far, the Internet is still chugging along, but a good question to ask is “Will it

continue to do so?" Although this paper does not attempt to answer that question, it can help us to understand why it will or why it might not. Good and bad network performance is largely dependent on the effective implementation of network protocols. TCP, easily the most widely used protocol in the transport layer on the Internet (e.g. HTTP, TELNET, and SMTP), plays an integral role in determining overall network performance. Amazingly, TCP has changed very little since its initial design in the early 1980's. A few "tweaks" and "knobs" have been added, but for the most part, the protocol has withstood the test of time. However, there are still a number of performance problems on the Internet and fine tuning TCP software continues to be an area of work for a number of people [3].

Over the past few years, researchers have spent a great deal of effort exploring alternative and additional mechanisms for TCP and related technologies in lieu of potential network overload problems. Some techniques have been implemented; others left behind and still others remain on the drawing board. We'll begin our examination of TCP by trying to understand the underlying design concepts that have made it so successful.

This paper does not cover the basics of the TCP protocol itself, but rather the underlying designs and techniques as they apply to problems of network overload and congestion. For a brief description on the basics of TCP, .The End-to-End Argument The design of TCP was heavily influenced by what has come to be known as the end-to-end argument . The key component of the end-to-end argument for our purposes is in its method of handling congestion and network overload. The premise of the argument and fundamental to TCP's design is that the end stations are responsible for controlling the rate of data flow. In this model, there are no explicit signaling mechanisms in the network which tell the end stations how fast to transmit, when to transmit, when to speed up or when to slow down. The TCP software in each of the end stations is responsible for answering these questions from implicit knowledge it obtains from the network or the explicit knowledge it receives from the other TCP host

Basic congestion control schemes

- Slow start
- Fast retransmission and Fast Recovery(Reno)

3.1.1 SLOW START: Slow start reduces the burst affect when a host first transmits. It requires a host to start its transmissions slowly and then build up to the point where congestion starts to occur[5].The host does not initially know how many packets it can send, so it uses slow start as a way to gauge the network's capacity. A host starts a transmission by sending two packets to the receiver. When the receiver receives the segments, it returns ACKs (acknowledgements) as confirmation. The sender increments its window by two and sends four packets. This buildup continues with the sender doubling the number of packets it sends until an ACK is not received, indicating that the flow has reached the network's ability to handle traffic or the receivers ability to handle incoming traffic .Slow start does not prevent congestion, it simply prevents a host from causing an immediate congestion state. If the host is sending a large file, it will eventually reach a state where it overloads the network and packets begin to drop. Slow start is critical in avoiding the congestion collapse problem. But new applications such as voice over IP cannot tolerate the delay caused by slow start and in some cases; slow start is disabled so the user can grab bandwidth. That trend will only lead to problems.

3.1.2 FAST TRANSMIT AND RECOVERY (RENO): Fast retransmit and fast recovery are algorithms that are designed to minimize the effect that dropping packets has on network throughput. The fast retransmit mechanism infers information from another TCP mechanism that a receiver uses to signal to the sender that it has received packets out of sequence [4]. The technique is to send several duplicate ACKs to the sender. Fast retransmit takes advantage of this feature by assuming that duplicate ACKs indicate dropped packets. Instead of waiting for an ACK until the timer expires, the source resends packets if three such duplicate ACKs are received. This occurs before the timeout period and thus improves network throughput. For example, if a host receives packet 5 and 7, but not 6, it will send a duplicate ACK for packet 5 when it receives packet 7(but not packet 6).Fast recovery is a mechanism that replaces slow start when fast retransmit is used. Note that while duplicate ACKs indicate that a segment has been lost, it also indicates that packets are still flowing since the source received a packet with a sequence number higher than the missing packet[5].In this case, the assumption is that a single packet has been dropped and that the network is not fully congested. Therefore, the sender does not need to drop fully back to slow start mode but to half the previous rate.

3.2 ACTIVE QUEUE MANAGEMENT (AQM):

Dropping packets is inefficient. If a host is bursting and congestion occurs, a lot of packets will be lost. Therefore, it is useful to detect impending congestion conditions and actively manage congestion before it gets out of hand.

Active queue management is a technique in which routers actively drop packets from queues as a signal to senders that they should slow down.

3.2.1 RANDOM EARLY DETECTION (RED):

RED is an active queue management scheme that provides a mechanism for congestion avoidance. Unlike traditional congestion control schemes that drop packets at the end of full queues, RED uses statistical methods to drop packets in a "probabilistic" way before queues overflow[8][14]. Dropping packets in this way slows a source down enough to keep the queue steady and reduces the number of packets that would be lost when a queue overflows and a host is transmitting at a high rate. RED makes two important decisions. It decides when to drop packets and what packets to drop. RED keeps track of an average queue size and drops packets when the average queue size grows beyond a defined threshold. The average size is recalculated every time a new packet arrives at the queue. RED makes packet-drop decisions based on two parameters:

Minimum threshold Specifies the average queue size below which no packets will be dropped.

Maximum threshold Specifies the average queue size above which all packets will be dropped.

3.2.2 FRED (Flow based Random Early Detection): FRED acts just like RED, but with the following additions. FRED introduces the parameters min. and max., goals for the minimum and maximum number of packets each flow should be allowed to buffer. FRED introduces the global variable ,an estimate of the average per-flow buffer count; flows with fewer than the packets queued are over flows with more. FRED maintains count of buffered packets queue length for each flow that currently has any packets buffered. Finally, FRED maintains a variable strike for each flow, which counts the number of times the flow has failed to respond to congestion notification; FRED penalizes flows with high strike values.

3.2.3 BLUE: The key idea behind BLUE is to perform queue management based directly on packet loss and link utilization rather than on the instantaneous or average queue lengths[16]. This is in sharp contrast to all known active queue management schemes which use some form of queue occupancy in their managing the congestion. It maintains a single probability $m\ P$, which it uses to mark (or drop) packets when they are enquired. If the queue is continually dropping packets due to buffer overflow, BLUE increments m P , thus increasing the rate at which it sends back congestion notification. Conversely, if the queue becomes empty or if the link is idle, It decreases its marking probability. This effectively allows BLUE to "learn" the correct rate it needs to send back congestion notification.

3.2.4 ADAPTIVE CHOKe: Adaptive CHOKe enforces the concept of queue-based and flow information. It is desirable for AQM schemes to act without storing a lot of information otherwise it becomes a overhead and non-scalable[11]. This algorithm modifies the CHOKe algorithm to remove its drawback. This algorithm also calculates the average queue size of the buffer for every packet arrival. It also indicates two thresholds on the buffer, maximum and minimum It reduces both the packet loss rate and the variance in queuing delay.

S.NO	ALGORITHM	ADVANTAGES	DISADVANTAGES
1	RED	Early congestion. No bias against traffic. No global synchronization.	Difficulty in parameter setting. Insensitivity to traffic load and drain rates.
2	FRED	Good protection from misbehaving flows.	Per-flow state. Difficulty in parameter setting. Insensitivity to traffic load and drain rates.
3	BLUE	Easy to understand. High throughput.	No early congestion detection. Slow response.
4	A-CHOKe	To protect well-behaved flows from misbehaving flow and adaptive flows from non-adaptive flows. packet loss with well adaptive tuned parameters.	Heavy load and unresponsive flow.

TABLE:1. ADVANTAGES AND DISADVANTAGES OF AQM ALGORITHMS

IV. CONGESTION CONTROL TECHNIQUES

Congestion control refers to the mechanisms and techniques used to control congestion and keep the traffic below the capacity of the network. Congestion control techniques can be broadly classified two broad categories:

4.1 Open loop: Protocols to prevent or avoid congestion, ensuring that the system never enters a Congested State. This category of solutions or protocols attempt to solve the problem by a good design, at first, to make sure that it doesn't occur at all. Once system is up and running midcourse corrections are not made. These solutions are congestion don't change much according to the current state of the system. Such Protocols are also known as Open Loop solutions. These rules or policies include deciding upon when to accept traffic, when to discard it, making scheduling decisions and so on. Main point here is that they make decision without taking into consideration the current state of the network. The open loop algorithms are further divided on the basis of whether these acts on source versus that act upon destination

4.2 Close loop: Protocols that allow system to enter congested state, detect it, and remove it. The second category is based on the concept of feedback. During operation, some system parameters are measured and feed back to portions of the subnet that can take action to reduce the congestion. This approach can be divided into 3 steps:

- Monitor the system (network) to detect whether the network is congested or not and what's the actual location and devices involved.
- To pass this information to the places where actions can be taken.
- Adjust the system operation to correct the problem.

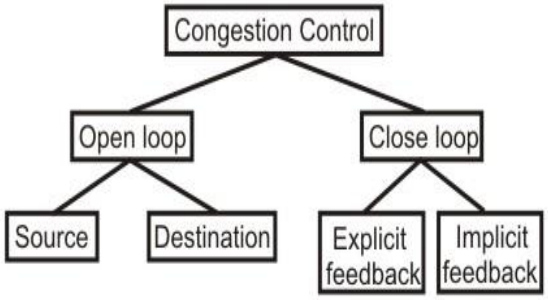

FIGURE 1:CONGESTION CONTROL CATEGORIES.

V. CONGESTION IN CONNECTIONLESS PACKET-SWITCHED NETWORKS

A network is congested when one or more network components must discard packets due to lack of buffer space. Given the above architecture, it is possible to see how network congestion can occur. A source of data flow on the network cannot reserve bandwidth across the network to its data's destination. It, therefore, is unable to determine what rate of data flow can be sustained between it and the destination .If a source transmits data at a rate too high to be sustained between it and the destination one or more routers will begin to queue the packets in their buffers. If the queuing continues, the buffers will become full and packets from the source will be discarded, causing losses of data. If the source is attempting to guarantee transmission reliability, retransmission of data and increased transmission time between the source and the destination is the result. As the load (rate of data transmitted) through the network increases, the throughput (rate of data reaching the destination) increases linearly. However, as the load reaches the network's capacity, the buffers in the routers begin to fill. This increases the response time (time for data to traverse the network between source and destination) and lowers the throughput.

Once the routers' buffers begin to overflow, packet loss occurs. Increases in load beyond this point increase the probability of packet loss. Under extreme load, response time approaches infinity and the throughput approaches zero; this is the point of congestion collapse. This point is known as the cliff due to the extreme drop in throughput.

5.1 AVAILABLE BIT RATE CONGESTION CONTROL IN ATM NETWORKS:

Asynchronous Transfer Mode (ATM) networks are characterized as connection-oriented cell switched networks. Cells are fixed in size (53 octets), and virtual circuits are established across the network to propagate the traffic for each connection. ATM provides several service classes to the connections established between sources and destinations: Constant Bit Rate (CBR), Variable Bit Rate (VBR), Available Bit Rate (ABR) and Unspecified Bit Rate (UBR).Of the four service classes, only ABR provides service degradation congestion control. With ABR, a source of cell traffic can specify minimum and maximum cell rates at connection establishment. If the connection is granted, then the source can transmit at a rate between the specified minimum and maximum cell rates. However, if the network becomes congested, the source may need to reduce its current cell rate to reduces the congestion. Several rate-based congestion control mechanisms were proposed to control source cell rates for the ABR service. The main proposals are summarized below.

5.1. BECN: In BECN, the intermediate switches return special resource management (RM) cells to sources of cells if they believe the source is causing congestion. Congestion is identified by monitoring cell queue lengths within the switches. On receipt of an RM cell, a source is obliged to halve its transmission rate. If no RM cells arrive after a recovery period, the source may double its cell rate (without exceeding the maximum cell rate specified at connect time). The recovery period is proportional to the current cell rate: as the cell rate drops, the recovery period shortens .BECN can be seen to be analogous to the Source Quench scheme, controlling the flow of a specific ATM connection.

5.2 PRCA (PROPORTIONAL RATE CONTROL ALGORITHM): PRCA has the source set every Congestion Experienced Bit on, but leave one in N off. If any RM cells arrive with the Congestion Experienced Bit off, then the destination sends RM cells back to the source, which prompt a source rate *i*ncrease. Again, Multiplicative Decrease and Additive Increase is used to control the source's cell rate. With this approach, the source cannot increase its rate unless RM cells are received. PRCA was found to have a fairness problem for long-path connections. If the probability of a cell having its Congestion Experienced Bit set by one ATM switch is p, then the probability of it being set over a p-switch path is p-n. Thus, long-path connections have a lower chance of receiving a rate increase RM cell than short-path connections. This is known as the beat-down problem.

5.3 EPRCA (ENHANCED PROPORTIONAL RATE CONTROL ALGORITHM): Enhanced PRCA (EPRCA) resulted from a combination of PRCA and the explicit cell rate scheme described above. Cell sources send a combination of RM cells and data cells with Congestion Experienced Bits. Switches along the path of a connection may calculate the Fair Share rate for the connection (and place the value in the RM cells), set the Congestion Experienced Bits if the connection is exceeding its fair share (or is causing congestion), or both. EPRCA allows switches to perform binary-feedback congestion control, explicit rate congestion control, or both [17]

VI. A RATE-BASED FRAMEWORK FOR CONGESTION CONTROL

Congestion information from the network, and to use this to adjust transmission rates which will optimize the overall network power. The obvious paradigm for such a congestion control system is to be rate-based.

6.1 OVERVIEW OF THE FRAMEWORK

The framework spreads the mechanisms of congestion control across both the Transport and Network Layers, measuring and using information on sustainable rates of traffic flow through the Network. The below steps are followed

The Transport Layer uses a rate-based flow control scheme, using rate information provided by the Network Layer. The flow

control and error control mechanisms are orthogonal, being performed by the Flow Control and Packet Retransmission functions, respectively

A source admits packets for each traffic flow (a single data stream from an application) into the network uniformly spaced in time. This reduces short-term congestion due to the traffic: this is performed by the Packet Admission function. Routers should attempt to preserve the time spacing of packets in a traffic flow

The routers in the Network Layer implement a congestion control scheme, part of which measures the sustainable traffic rate across the network for each traffic flow; this is performed by the Sustainable Rate Measurement function This sustainable rate measurement is passed to the destination machine and then via acknowledgment packets to the source machine.

Routers implement both congestion avoidance and congestion recovery mechanisms as the rest of the Network Layer congestion control scheme. Routers partition the available resources fairly amongst traffic flows, in both uncongested and congested operating regimes. Routers drop packets when in the operating region of the congestion cliff. These operations are performed by the Packet Selection, Packet Queuing and Packet Dropping functions [7].

6.2 ROUTER PACKET DROPPING FUNCTION:

Router may use any packet dropping function which meets the design decisions. Dropping functions which attempt to preserve packet spacing for each source may be useful. If the rate-based framework keeps the network operating at maximum power, packet loss will occur rarely, and the choice of functions for both Packet Queuing and Packet Dropping will have little effect on the framework.

6.3 ROUTER CONGESTION RECOVERY:

The congestion framework has congestion avoidance; there may be times when a router reaches the point of congestion collapse. For example, a new traffic flow may learn about its rate allocation before other large round-trip flows learn that their allocation has been reduced. For a small period of time, more packets will be admitted into the network than can be transmitted. The large round-trip flows will eventually learn about their new rates, but only after one round-trip. It should be possible to inform these flows about the problem in less than one round trip .This control mechanism must also not overburden the network with congestion control information .For this issue, the congestion recovery mechanism Rate Quench Packet Generation is used. When a router considers itself congested, it returns Rate Quench packets to the sources it believes are the cause of the congestion. These packets also have a Return Rate field, which the congested router sets to values which will alleviate congestion from each of the congesting sources. Upon receipt of a Rate Quench packet with a Return Rate field lower than the source's current rate, the source is obliged to immediately lower its rate to the value in the Rate Quench packet. Rate Quench packets can never raise a source's transmitting rate. In order to ensure stale Rate Quench packets are not used, a source must discard Rate Quench packets which have a sequence number less than the last valid Rate Quench packet received.

VII. CONCLUSION

This paper presents the study of congestion control and elaborates various issues related with it. As the congestion control is the most important factor of any packet switching network, the whole performance and accuracy of network is directly related to it, the congestion control becomes more important. We briefly 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] D. Cavendish, M. Gerla, and S. Mascolo, "A control theoretical approach to congestion control in packet networks," IEEE/ACM Trans .Network., vol. 12, no. 5, pp. 893–906, Oct. 2004.

[2] Sally Floyd and Kevin Fall. Promoting the Use of End-to-End Congestion Control in Internet. IEEE/ACM Transactions on Networking, August 1999

[3] Jeffrey Semke, Jamshid Mahdavi and Matthew Mathis. Automatic TCP Buffer Tuning, Computer Communications Review, ACMSIG COMM, Volume 28, Number 4, October 1998

[4] Van Jacobson. Modified TCP Congestion Control Avoidance Algorithm. end-2-end-interest mailing list, April 30, 1990

[5] W. Stevens. TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms, January 1997, RFC 2001.

[6] Sapna Gupta, Nitin Kumar Sharma and K.P. Yadav "A Survey on Congestion Control & Avoidance", Vol. 2 (9), 2012, 790-797

[7] Warren Keith Toomey, "A Rate-Based Congestion Control Framework for Connectionless Packet-Switched Networks" Thesis, Decenber,1997.

[8] V. Misra, W. Gong and D. Towsley: ``Fluid-based Analysis of Network of AQM Routers Supporting TCP Flows with an Application to RED" , In Proc. of ACM/SIGCOMM (2000).

[9] T. Azuma, and M. Fujita :``Congestion Controlin Computer Networks" , Journal of The Society of Instrument and Control Engineers, Vol. 41, No. 7, pp. 496--501 (2002--7)

[10] S. Low, F. Paganini and J. Doyle: "Internet Congestion Control:An Analytical Perspective" , IEEE Control Systems Magazine, Vol. 22, No. 1, pp. 28-4(2002)

[11] S. Athuraliya, V. H. Li, S. H. Low, and Q. Yin, REM: active queue management, IEEE Network, vol. 15, no. 3, pp.48-53, 2001.

[12] L. S. Brakmo and L. Peterson, TCP Vegas: end to end congestion avoidance on a global Internet, IEEE Journal on Selected Areas in Communications, vol. 13, no. 8, October 1995.

[13] V. Firoiu and M. Borden, A study of active queue management for congestion control, in Proc. IEEE Infocom, March 2000

[14] S. Floyd, R. Gummadi, and S. Shenker. "Adaptive RED: An Algorithm for Increasing the Robustness of RED's Active Queue Management", Preprint, http://www.icir.org/floyd/papers.html, August 2001.

[15] G. Sasikala, E. George Dharma Prakash Raj, "P-CHOKe: A Piggybacking-CHOKe AQM Congestion Control Method", IJCSMC, Vol. 2, Issue. 8, August 2013

[16] Jyoti Pandey Aashish Hiradhar, "A Survey on AQM Control Mechanism for TCP/IP Flow" IJARCSSE, Volume 4, Issue 4, April 20

[17] Sabato Manfredi, "Decentralized Queue Balancing and Differentiated Service Scheme Based on Cooperative Control Concept" IEEE TRANSACTIONS, VOL. 10, NO. 1, FEBRUARY 2014.

AUTHORS

First Author – C.Socrates, PG Scholar, PITAM, Anna University, Chennai, Tamil Nadu.
Second Author – P.M.Beulah Devamalar, Principal, PITAM, Anna University, Chennai, Tamil Nadu.

Third Author – R.Kannamma Sridharan, Assistant Professor,
PITAM, Anna University, Chennai, Tamil Nadu.