A NEW MODEL FOR CONGESTION DETECTION IN HIGH SPEED NETWORK WITH HIGH SPEED PROTOCOLS CREATING

T. Sheela, Dr. J. Raja

ABSTRACT

Continuously growing needs for Internet applications that transmit massive amount of data has led to the emergence of high speed networks. The primary factor hindering the flow of Traffic is Network Congestion. Data transfer must take place without any congestion. The Internet carries certain critical data and information which need to be delivered to the receiver at all cost. Thus Congestion Detection plays a key role in high speed networks. The Traditional TCP has been used over the years to detect and control Congestion. Various algorithms such as ECN, DECbit, RED Queue Mechanism are adopted by TCP to detect Congestion. In this paper, a rational mechanism utilizing XCP protocol is used. Adopting the techniques of a new congestion detection module with effective feedback mechanism and modified window adjustment is achieved. A Separate Congestion Detection Module is developed to detect and inform immediate Congestion Detection and Control. The feedback parameters are calculated based on arrival rate, service rate, traffic rate and queue size. Resulting in no drastic decrease in window size, better increase in sending rate because of which there is a continuous flow of data without congestion. Therefore as a result of this, there is a maximum increase in throughput, high utilization of the bandwidth and minimum delay. The result of the proposed work is presented as a graph based on throughput, delay and window size. Thus in this paper, XCP protocol is well illustrated and the various parameters for Congestion Detection are thoroughly analyzed and adequately presented.

Keywords: TCP/IP, Congestion Detection, Window Management, Feedback Control, Queue Management, Explicit Control Protocol.

1. INTRODUCTION

The High Speed Network refers to the networks supporting high data rates like high speed LAN's and Ethernets. As there is a growing demand for high-speed networks, data transfer must take place without any congestion[1][2][3]. A congestion avoidance method that can achieve high utilization, small queuing delay, freedom from oscillation and fairness in bandwidth allocation has been a major objective of network research in recent years. In this network, the occurrence of congestion is mainly due to the presence of multiple senders transmitting volume of data at varying arrival rate.

Dina Katabi describes that the flow product like bandwidth and delay increases, TCP becomes inefficient and prone to instability regardless of the queuing scheme. Different TCP variants (high speed protocols) like BICTCP, CUBIC, FAST, HSTCP, LTCP, and STCP were proposed to improve the performance of the congestion control algorithms in high-speed networks [13]. The Transmission Control Protocol is the most widely used protocol for data transmission. The Traditional Transmission Control Protocol utilizes a rational mechanism to control Congestion. In TCP, loss of packets by itself acts as a signal, which indicates the sender to lower down the congestion window to limit the number of packets to be sent, and thereby congestion can be reduced. Thus TCP initiates congestion control only if a packet is considered lost[6][7].

The primary disadvantage of the Existing Scenarios is that even though Congestion is avoided by their application, the Network Performance is degraded. Hence we propose a rational mechanism which would enable effective Congestion Detection and Control while rendering reasonable performance.

The Proposed Work presents XCP, the eXplicit Congestion Control Protocol, which is mainly for Congestion detection and control. It exhibits high bandwidth utilization, good throughput, it is stable for any link capacity, it have good fairness mechanism and better control over the responsiveness of the traffic. The XCP protocol is placed in transport layer and it's operating between TCP and IP protocol. The Proposed work enables to handle Congestion by adopting the following techniques namely Feedback Mechanism, Window Adjustment and Queue Management. By a combination of all these three techniques effective Congestion Avoidance and Control is achieved.

The rest of this paper is organized as follows. Section 2 explains the basics of XCP and its Overview. Section 3 describes the congestion detection module. Section 4 explains the results and discussion. Section 5 indicates the conclusion.

2. XCP BASICS

2.1 XCP overview

The new Explicit Control Protocol outperforms TCP and remains efficient, fair, scalable, stable and XCP generalizes Explicit Congestion Notification proposal. XCP is modeled and demonstrated as stable and efficient regardless of link capacity, round trip delay. XCP achieves fair bandwidth allocation, high utilization, small standing queue size, and near-zero packet drops with both steady and highly varying traffic. Additionally, XCP does not maintain any per-flow state in routers and requires few CPU cycles per packet, which makes it suitable for high-speed networks. This new eXplicit Control Protocol, XCP, generalizes the Explicit Congestion Notification proposal (ECN)[2][7]. In addition, XCP introduces the new concept of decoupling utilization control from fairness control. This allows a more flexible and analytically tractable protocol design and opens new avenues for service differentiation. Fig.1. represents the XCP overview.

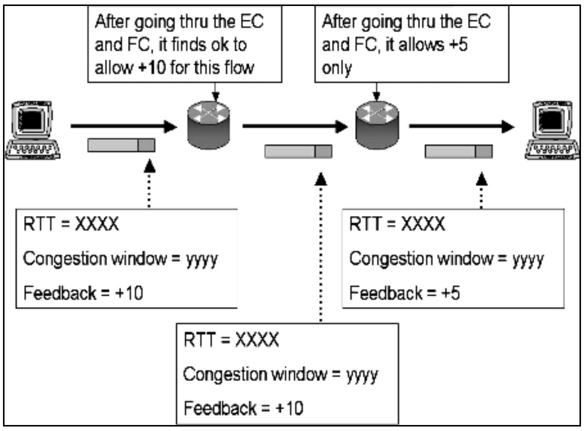

Fig .1- XCP Overview

3. CONGESTION DETECTION MODULE

The Congestion Detection Module accepts input traffic and enforces the corresponding algorithm. We classify load onto three categories,

- Normal Flow
- Medium Congestion
- Heavy Congestion

CASE 1 – Normal Flow

Whenever Normal Flow is detected, the Normal Flow Module is called, which acts as follows,

- Based on the destination IP, packets are routed to the adequate destination.
- Acknowledgment is sent back to all Senders.
- Throughput and Utilization are calculated.
- Control is rendered back to the Source Module.

CASE 2- Medium Congestion

Whenever Medium Congestion, i.e. Load ranging from 80-95% is obtained, the following processes are done,

- Queues are initialized to handle incoming traffic.
- The Queue parameters are calculated.
- Waiting Time, Resident time and Processing time are calculated.

The parameters to be calculated in the queue module are as follows,

Queue Size(r) $=\rho / (1-\rho)$ (1)

The queue size allocated for transfer.

Resident Time(T_r) $= T_s/(1-\rho)$ (2)

The Time of stay of the packets in the queue

Waiting Time(T_w) $= \rho T_s /(1-\rho)$ (3)

Processing Time(T_r) $= Ts+Tw$ (4)

The Net Time required to render a response. The queue management procedure adopted in XCP is shown. XCP uses the concept of BUFFERS. The packets are placed in buffers and are not dropped. Congestion occurs commonly in many of the High-speed networks. The occurrence of Congestion is mainly due to the presence of multiple senders transmitting volumes of data at varying arrival rates

CASE3 – Heavy Congestion

Whenever Heavy Congestion is detected, the load is ranging above 95%. Queue Management Module is called and the same procedure is adopted with more buffers. Also Window Adjustment and Feedback are rendered quickly. After analyzing the Input packets the corresponding actions are rendered. We now present the algorithm, which would better explain the working of this module. The congestion detection module is represented in Fig 2.

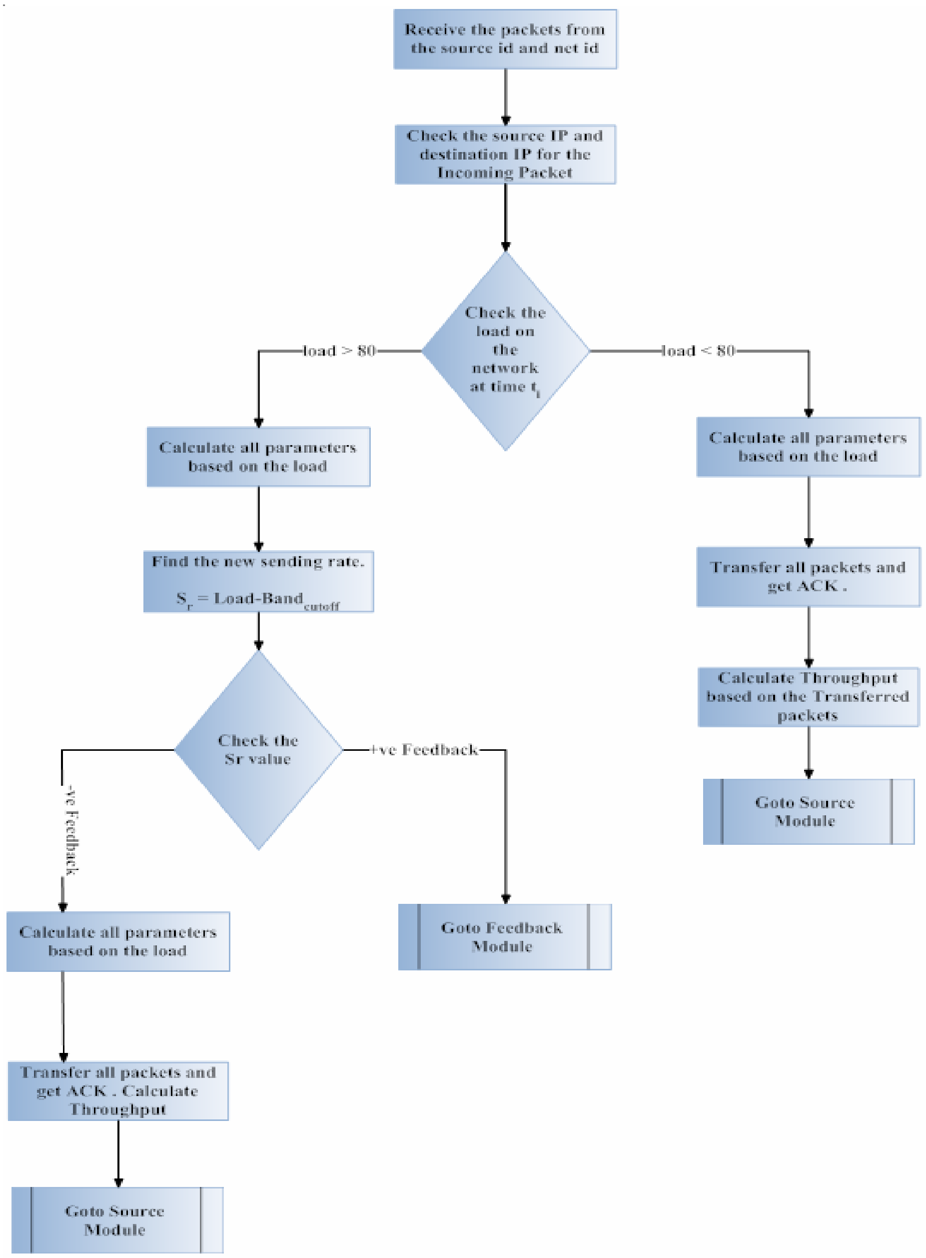

Fig.2. Congestion Detection Module

3.1 Algorithm

1. Receive packets from all senders. □(S1 to Sn).
2. Check the Source IP address and the Destination IP address.
3. Obtain load □□$_i$ at Time t_i. (I = 1 to n) depending upon time and the No of senders.
4. Analyze the Input Traffic
5. If it lies from (0-80%) then
    i. Calculate all parameters
    ii. Transfer all packets and get Ack./ *Normal flow*/
6. Else if it is >80 then
    i. Calculate the new Sending rate.
    ii. Check the + ive or – ive feed back
7. If the feedback is positive /* Medium Congestion*/(Case of 80% to 95%)
    i. Calculate all parameters
    ii. Transfer all packets and get Ack.
8. If the feedback is negative/* Severe Congestion*/(>95%)
    i. i. Calculate all parameters
    ii. ii. Go to feedback module

3.2 Evaluation

The above algorithm clearly explains the factors and the various procedures that are to be adopted in the Congestion Detection Module.

The factors that are to be calculated in the Congestion Detection Module are as follows,

Flow Rate : It may be defined as the total number of packets/sec sent by one sender at time t_i and is given by λ_i .

Load on the network : It is defined as the number of flows per second

For n flows, Load is given by $\sum \lambda_i$ I = 1 to n where n is the number of flows

T_s(Service Time) = Packet Size/Link Capacity (5)

The time required to service a given request

ρ (Utilization) = λT_s (6)

The amount of bandwidth and resources utilized.

The flowchart explains the various processes that are being adopted in the Congestion Detection Module.

The following Figure 3 depicts the dumb-bell topology consisting of senders, destinations and routers , in which the proposed mechanism has been tested and simulated using OMNeT++ version 4.0.

The Congestion Detection Module starts with accepting packets from multiple Senders. The Network architecture, which we propose, is a Simple Dumb-bell Topology which is represented in Fig 3.

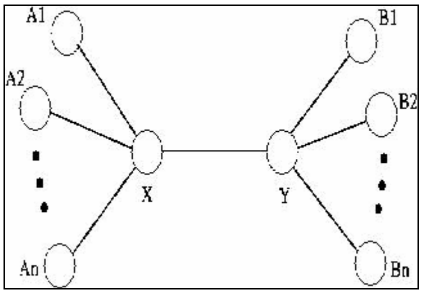

Fig.3. Dumb-bell topology

The Figure above illustrates a simple dumb-bell topology wherein multiple senders (A1-An) transfer packets to multiple nodes (B1-Bn). Here X and Y are routers. We also propose to have multiple routers which enable better performance.

4. RESULTS AND DISCUSSION

4.1 Throughput

The following Table 4.1 and Figure 4.1 indicate data rate and throughput. As the data rate increases the packet losses do not occur the throughput is maintained at around 100%. In the proposed mechanism during severe congestion the throughput level is maintained as 96% and it is represented in the Table 1. In the existing mechanism whenever congestion occurs the throughput drops significantly because of excessive packet loss in the network and it is described in the Fig.4.

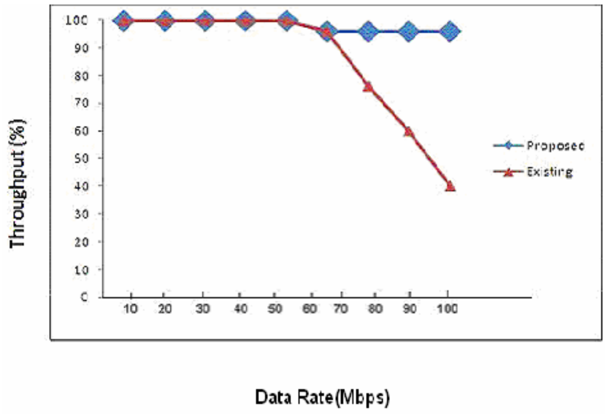

Fig.4. Data rate Vs Throughput

Table 1 Throughput

Data Rate	Throughput (Mbps)	Throughput In %
7	6.9	99
18	17.9	99.4
25	24.9	99.6
36	35.9	99.7
40	39.9	99.7
62	61.9	99.8
75	74.9	99.8
84	83.9	99.8
95	92.1	96
95.33	92.1	96

4.2. Utilization

The Utilization for both type of networks renders the following graphs. A Constantly increasing curve is obtained. As seen from the Figure .5 as the data rate increases the utilization also increases .When severe congestion occurs as a result of high data rate, the utilization increases to 1 and above. Utilization is calculated in Equation (6) and it is based on the service time and as service time is extremely less , large volume of data are transferred through the network in a faster manner resulting in high utilization of bandwidth.

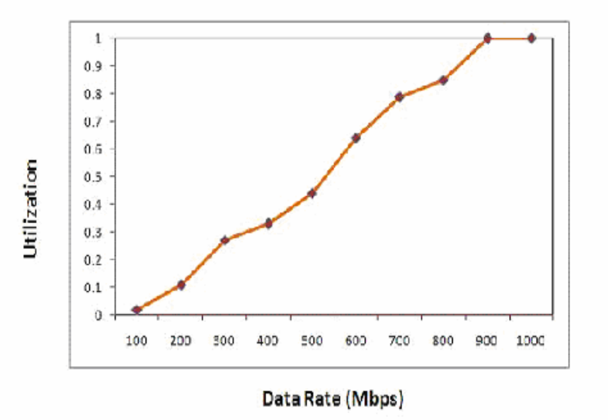

Fig.5. Data rate Vs Utilization

5. CONCLUSION

The proposed scheme achieves the maximum throughput with minimum delay in high speed network with the help of the effective Congestion detection module. In this paper, the algorithm for Window adjustment Procedure for XCP is proposed which achieves 96% efficiency when compared with High Speed Protocols like HSTCP, BICTCP, HTCP and cubic TCP. The window size is altered drastically based on the time and packet flow and the utilization of the bandwidth is reduced. But in XCP the procedure is based upon the bandwidth and feedback, where the input packets are being regulated by altering the source node and there by reducing the window size in unit step. So the sender can restrict the flow and can avoid congestion thereby resulting in effective Bandwidth utilization. Also the Feedback Mechanism in XCP enables to identify occurrence of congestion effectively. This mechanism is highly useful, particularly in Large Networks wherein many senders transfer continuously at varying data rates. Hence in comparison to the existing scenarios XCP provides a better mechanism for effective data transfer. This mechanism is highly useful, particularly in Large Networks wherein many senders transfer continuously at varying data rates. Hence in comparison to the existing scenarios XCP provides a better mechanism for effective data transfer. Also then by effectively calculating the Queue size, Window Size, Utilization and delay can still be minimized.

REFERENCES:

1. Sherali Zeadally and Liqiang Zhang (2004)Enabling giga bit network Access to end Users" IEEE proc., Vol. 92 No. 2,pp:340-353
2. Peng Wang and David .L .Mills,(2006) "Simple analysis of XCP equilibrium performance", Proc. 40th IEEE Ann.Conf, pp:585-590, IEEE'06
3. Li-song, shao ,He-ying , Zhang, Wenhua,Dou,(2006) "General Window Based Congestion Control Buffer Occupancy, Network Efficiency and Packet loss" proc of 4th Ann Comm.Networks, pp:195-201, IEEE'06
4. Steven.L.Low,achlan.L.H.Andrew Bartek (2005) "Understanding XCP: Equilibrium and fairness", Proc of IEEE Infocom, pp: 1-12.
5. Fernando Paganini, Steven.H.Low(2005) "Congestion Control for High Performance Stability and Fairness in General Networks" IEEE/ ACM NETWORKING, vol.13 .No.1, pp:43-56
6. Nishan R.Sastry and Simon.S.Lam(2005) "CYRF:A theory of window based unicast congestion control", IEEE/ACM TRANSACTIONS ON NETWORKING, vol. 13, pp:330-342 .
7. Yongguang and Lang(2005) "An implementation and experimental study of the explicit control protocol (XCP)", IEEE 24th Ann. Joint Conf of the computer & communication. Soc-2, pp: 1037-1048.
8. Sanjeeva Athuraliya and Steven.H.Low, (2001)"REM:Active Queue Management" IEEE Network, pp: 48-53.
9. Ramesh Johari and David Kim Hong Tang(2001) "End-to-End Congestion Control for Internet: Delays and stability", IEEE/ACM TRANSACTIONS ON NETWORKING,vol.9 No. 6, pp: 818-831.
10. Sally Floyd and Kevin Fall,(1998) "Promoting the use of end-to-end congestion control in internet" , IEEE/ACM TRANSACTIONS ON NETWORKING, vol. 7,no. 4, pp:1-15
11. David X, (2006)"BIC TCP: Motivation, Architecture, Algorithms, Performance", IEEE/ACM TRANSACTIONSONS ON NETWORKING,VOL. 14, NO. 6.
12. Kyle Halliday, Andrew Hurst and Jerom Nelson,(2004) "Analysis of next Generation TCP", pp:12-12.
13. Yee-Ting Li,Douglas Leith and Robert Shorten,(2004)" Experimental Evaluation of TCP protocols for High-Speed Networks.
14. Damon Wischik(2006) , "Buffer Requirements for High Speed Routers".
15. Khaled M.Elleithy and Ali S.Al-Suwaiyan, (2001)"Network Traffic Characterization for High –Speed Networks Supporting Multimedia.", pp:200-202, IEEE'01.
16. Satyanarayan Reddy K and Lokanatha C Reddy (2008)"A Survey on Congestion Control Mechanism in High Speed Networks", IJCSNS, International Journal of Computer Science Network Security, vol. 8, no.1, pp:208-212.
17. Satyanarayan Reddy K and Lokanatha C Reddy (2008)"A Survey on Congestion Control Protocols for High Speed Networks", IJCSNS, International Journal of Computer Science Network Security, vol. 8, no.7, pp:44-52.

18. Steven H.Low (2005) “A duality Model of TCP Queue Management Algorithms”, IEEE/ACM Transactions on Networking, vol.11, no.4, pp:520-546,
19. Huajun Liu(2005) “Research of the Queue Management Algorithms Journal of Communication and Computers, vol 2, No 5, pp:16-19
20. Jiakun Liu (2003)“Threshold Crossing Analysis of High Speed Network Traffic”, pp:915-918, IEEE,’03
21. C.Jin.D.X.Wei & S.H.Low(2004), “FAST TCP, R. Agrawal and R. Srikanth, Fast algorithms for mining association rules, in: Proc. VLDB, Santiago, Chile, September 1994.
22. D.J.Leith & R.N.Shorten,(2004) “H-TCP Protocol for Highspeed Long Distance Networks”, in 2nd Workshop on Protocols for fast Long Distance Networks, USA.
23. L.Xu, K.Harfoush, I.Rhee(2004), “Binary Increase Congestion Control (BIC) for Fast Long Distance Networks”, In IEEE Infocom,
24. C.Jin.D.Wei & S.H.Low,(2006) “Fast TCP Motivation,Architecture,Algorithms, Performance”,IEEE/ACM Transactions on Networking, Vol.14. no. 6,pp:1246-1259.
25. Yee.Ting, Li, douglas Leith & Robert.N,92007) “Experimental Evaluation of TCP Protocols for High Speed Networks”, IEEE/ACMTransaction Networking, pp:1109-1122.