A Review: Fuzzy Logic in Congestion Control of Computer Network

Shilpa N. Ingoley
Sr. Lecturer,
Smt. S.H.M. Institute of
Technology, Ulhasnagar, India

Madhu Nashipudi
Asst. Professor
PIIT, New Panvel
Navi Mumbai Maharashtra, India

ABSTRACT
Congestion being a non-linear and dynamic problem in the Internet Fuzzy Controller can be use to deal with this problem. Congestion can be controlled at core and bottleneck router differently. To get better quality of service over the best effort network, multiple queue management algorithms can be used. Fuzzy logic helps to increase throughput, reduction in packet drop and delays.

General Terms
Transmission Control Protocol (TCP), Fuzzy Logic Controller (FLC), Fuzzy Inference System (FIS), MFs (Membership Functions)

Keywords
Congestion control, Fuzzy Logic(FL), TCP, Computational Intelligence (CI), Queue management.

1. INTRODUCTION
Congestion is an important issue in the computer network. Congestion takes place when the numbers of packets sent are more than the capacity of the network. The primary cause of network congestion is that data in networks are overloaded and available resources such as bandwidth, buffers, queues, processor time etc. are inadequate to handle such traffic loads. Congestion occurs on shared networks when multiple users contend for the access to the same resources. Whenever sum of demands is more than the available resources, then queues are built up. This occurs because incoming packet rate is greater than the departure rate at routers and switches. Queues are built up whenever waiting is involved. The available queue size starts increasing as time increases, before routers or switches take packets for processing. If this situation is not handled it will leads to congestion or Thrashing. Thrashing is a situation where large amount of computer resources are used to do minimal amount of work. In thrashing inappropriate amount of time is spend just accessing the shared resources.

Whenever network is congested it will leads to the performance degradation. Delay will be more and throughput will be less. When a packet is delayed, the source for not receiving an acknowledgement (ACK), retransmits the packet. This makes the delay and the congestion worse. When the load exceeds the capacity, the queues will become full and routers have to discard some packets. Discarding packets does not reduce the number of packets in the network. This happens because the source retransmits the packets, using timeout mechanism when packets fail to reach the destination. Multiple packet losses, low link utilization, low throughput, the blocking of new connections and high queuing delay will lead to the congestion collapse. Congestion Collapse is a condition which networks reach, when little or no useful communication is happening due to congestion. Network performance is largely depends on the effective implementation of network protocols.

1.1 Reasons for Congestion
Following are some of the reasons for congestion:

i) The primary cause of network congestion is that data in networks are overloaded and available resources are inadequate to contain such traffic loads. ii) Total sum of demand is greater than the available resource. iii) Explosive growth of Internet, number of Internet users is increasing day by day. iv) Internet is heterogeneous, which is connected by millions of asynchronous systems and controlling such system is a complex task. v) Resources are distributed, thus the difficulty of global assessment of network load. vi) Demands are unpredictable i.e. the- arrival rates and resource holding times (transmission, processing) are nondeterministic. vii) Design of system is made by considering for the worst case demands. viii) Support for the integrated service such as video, voice and data applications simultaneously. ix) Users want more bandwidth and speed to run new and real time application like VOIP, IP-TV, On-line games, Multimedia etc. x) Dynamic nature: Non-linearity and unpredictability for the variable size packets, busty network traffic and jitters.

1.2 Myths about Congestion Control

There are some myths about the congestion like infinite memory, high speed links and high speed processor will solve the congestion problem. Contrary to these beliefs, without proper protocol redesign, the above developments may lead to more congestion and thus will also reduce network performance.

Congestion is a dynamic problem. It cannot be solved with static solution alone. So there is a need for protocol designs that protect network in the event of congestion. The explosion of high-speed networks has led to more unbalance network causing congestion. In particular, packet loss due to buffer storage is a symptom and not a cause of congestion [2].

This paper is organized as follows: Section 2 discusses the commonly used congestion control technique like TCP and Queue management. To reduce congestion, Section 3 considers various fuzzy based congestion control algorithms at core router, edge router and for multiple queue management. Comparative study of fuzzy methods (as discussed in section 3) with conventional methods is illustrated in section 4. Finally section 5 concludes this paper.

2. CONGESTION CONTROL TECHNIQUES

Congestion control calculates the amount of data the sender can send to the destination on the network. It is concerning about controlling traffic entry into a communication network. Congestion control refers to techniques and mechanisms that can either prevent congestion, before it happens or removes congestion, after it has happened. Conventional congestion control scheme are divided into following *two* broad categories:

i) Open –Loop Congestion Control

This helps in preventing congestion before it happens. Congestion can be prevented at either the source or destination by using Retransmissions Policy, Acknowledgment Policy and Discard Policy.

ii) Closed-Loop Congestion Control

This helps to alleviate congestion after it happens. It uses Back Pressure, Choke Point, Implicit Signaling, and Explicit Signaling to remove congestion.

As per [15] there are two kinds of congestion control:

- End host based (which uses TCP) and
- Router based (which uses AQM)

2.1 Transmission Control Protocol (TCP)

TCP (RFC 793) is the most commonly used connection oriented and end-to-end protocol in the Internet. TCP has two major functions, first for maintaining reliable data transfer and second to control the congestion. TCP is a window based control, which uses the sliding window protocol. The sender/ transmitter keeps track of *receiver window (rwnd*) and *congestion window* (*cwnd*). The actual window size is minimum of these two. Since TCP is a self-clocked algorithm, the window size is modified at intervals according to the *roundtrip time* (RTT) or on the basis of ACK it receives. It uses *three* phases for congestion control:

1. Slow Start 2. Congestion Avoidance 3.Congestion Detection – Fast Retransmit/Fast Recovery

In slow start, the size of window increases by one MSS (Maximum Segment Size) each time whenever an ACK arrives. The algorithm starts slowly but increases exponentially. Slow start cannot continue indefinitely, so there is slow start threshold to stop (*ssthresh*) this phase. Once it reaches the threshold it goes into congestion avoidance phase. In this phase, size of window increases on the basis of RTT and not on the basis of arrived ACKs. Third phase is congestion detection, in which it considers RTO timer times-out or whenever it receives three duplicate ACKs as a signal of congestion. In both the cases it reduces value of threshold to half of the current window size. If a time-out occurs, it considers there is strong possibility of congestion and starts slow start phase again. If it received 3 duplicate ACKs, it considers there is weaker possibility of congestion and starts congestion avoidance phase. This is also called Fast Retransmit/Fast Recovery because it does not wait for retransmission timer to expire. There are various implementation of TCP: TCP Tahoe and Reno, TCP Vegas, Compound TCP, New Reno, TCP Hybla, TCP CUBIC etc.

TCP is optimized for accurate delivery rather than timely delivery. Therefore, TCP sometimes incurs relatively long delays (in the order of seconds) while waiting for out-of-order messages or retransmissions of lost messages. It is not particularly suitable for real-time applications such as Voice over IP, IP_TV, Online game, multimedia, music etc.

2.2 Active Queue Management

Discussion of congestion naturally involves queuing. Congestion in a network or internetwork occurs because routers and switches have queues-buffers that hold the packets before and after processing. Role of the router becomes important as it leaves some scope to control the congestion effectively. Each router in the network uses queue management (QM) and scheduling as two classes of algorithms that are related to congestion control. The QM algorithms try to control the length of packet queues by dropping packets when appropriate. Scheduling algorithms on the other hand, determine which packet to drop next and which is to send. Scheduling algorithms also used to manage the allocation of bandwidth among flows. [1] Network congestion induced by traffic leads to wasting all the resources that the packet consumed on its way from source to destination. Properly managed queues can minimize dropped packets and network congestion, as well as improve network performance.

The most basic technique of queue management is FIFO (first-in, first-out). In this packets are processed in the order in which they arrive in the queue. There are many QM schemes have been proposed to provide high network utilization with low loss and delay. QM algorithms such as Random Early Detection (RED), Adaptive RED (A-RED), Random Exponential Marking (REM), priority queuing etc., regulate queues at the bottleneck links in TCP/IP best-effort networks.

The QM approach can be contrasted with the “Tail Drop” (TD) (also called “Drop Tail” (DT)) QM approach employed by common Internet routers. In TD the discard policy of arriving packets is based on the overflow of the output port buffer. Contrary to TD, QM mechanisms start dropping packets earlier in order to be able to notify traffic sources about the incipient stages of congestion. QM allows the router to separate policies of dropping packets from the policies for indicating congestion. The use of Explicit Congestion Notification (ECN) was proposed in order to provide TCP an alternative to packet drops as a mechanism for detecting incipient stage of congestion in the network. BECN (backward explicit congestion notification) sends a frame in the backward direction to senders with the BECN bit set to inform senders to slow down. In FECN (forward explicit congestion notification), a frame can be send in the forward direction to receiving nodes with the FECN bit set. This bit will tell the forward nodes that they should inform the sender to slow down.

A QM-enabled gateway can mark a packet either by dropping it or by setting a bit in the packet’s header if the transport protocol is capable of reacting to ECN. The use of ECN for notification of congestion to the end-nodes generally prevents unnecessary packet drops. Priority Queuing provides better Quality of Service than FIFO queue because higher priority traffic, such multimedia can reach destination with less delay.

Active Queuing management (AQM) algorithm drops the packets before the Queue becomes full. It uses average queue length as a congestion indicator. Goals of AQM are to reduce the average queue length which helps in decreasing end-to-end delay and reducing the packet losses which results in more efficient resource allocation.

3. FUZZY LOGIC BASED CONGESTION CONTROL

Fuzzy Logic was proposed by Prof. Lotif Zadeh as a means of representing or manipulating data that is not precise but rather fuzzy. Fuzzy logic (FL) is aimed at a formalization of modes of reasoning which are approximate rather than exact. It also provides an alternative solution to non-linear control because it is closer to real world. Non-linearity is handled by rules, membership functions, and the inference process which results in improved performance.

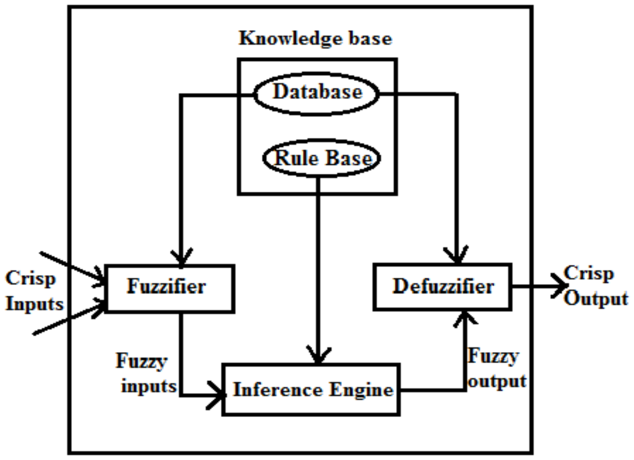

Fig 1: Basic structure of a fuzzy inference system

Fuzzy Controller: A fuzzy controller works similar to a conventional system: it accepts an input values, performs some calculations and generate an output value. Figure 1 shows the basic structure of Fuzzy System. It includes four main components. **A Fuzzifier**: It translates crisp (real valued) inputs into fuzzy values. **An Inference Engine**: That applies a fuzzy reasoning mechanism to obtain a fuzzy output. **A Defuzzifier**: Which translates this latter output into a crisp values. **A Knowledge Base:** It contains both an ensemble of fuzzy rules known as the rule base, and an ensemble of membership functions, known as the database.

3.1 Fuzzy Method for Core Router Buffer

Fuzzy logic controllers are expected to work in situations where there is a large uncertainty or unknown variation in the parameters of the system under control. Adaptive Drop Tail using Fuzzy Logic (ADT-FL) is proposed by [5]. Adaptive fuzzy logic controllers are capable of incorporating linguistic information from human operators or experts, whereas conventional controllers cannot. The ADT-FL controls the queue limit of the DT algorithm under varying traffic intensities and link bandwidth in order to save the router from congestion, subsequently preventing the network from congestion.

In this model there are two inputs, Traffic Intensity and Available Link Bandwidth which are denoted by Trafint and AVbw respectively. The Traffic Intensity and Available Link Bandwidth Trafint and AVbw are classified into 3 linguistic variables: {Low, Medium, High} which is represented as {L, M, H} as shown in Fig 2.

Output function of Queue limit -Qlim is classified into five linguistic variables: {Very_Low, Low, Medium, High, Very_High} which is represented as Qlim = {VL, L, M, H, VH} as shown in Fig 3. When a packet arrives inside the router interface buffer, the current value of Trafint and AVbw are obtained and Qlim value is calculated based on two inputs and set of rules.

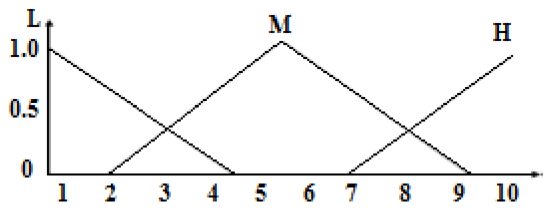

Fig 2: Input Membership Function – Trafint & AVbw

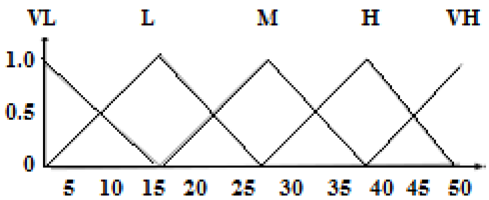

Fig 3: Output Membership Function – Queue Limit

The linguistic variables are defined as follows:

If (Trafint is Low) and (AVbw is Low) then (Qlim is L)
If (Trafint is Low) and (AVbw is Medium) then (Qlim is L)
If (Trafint is Low) and (AVbw is High) then (Qlim is VL)
If (Trafint is Medium) and (AVbw is Low) then (Qlim is M)
If (Trafint is Medium) and (AVbw is Medium) then (Qlim is M)
If (Trafint is Medium) and (AVbw is High) then (Qlim is L)
If (Trafint is High) and (AVbw is Low) then (Qlim is VH)
If (Trafint is High) and (AVbw is Medium) then (Qlim is H)
If (Trafint is High) and (AVbw is High) then (Qlim is M)

Low packet loss rate leads to increase in throughput and decrease in delay thereby increasing the quality of service. The performance of the proposed Adaptive Drop Tail fuzzy logic (ADT-FL) [5] is compared with the traditional Drop Tail and RED mechanisms. The proposed Adaptive Drop Tail based Fuzzy Logic (ADT-FL) is compatible with changing network and traffic conditions. When the incoming traffic bursts, the current internet router buffers fail to control congestion effectively where as ADT-FL gives better performance.

3.2 Fuzzy Method for Network Edge and Bottleneck Router

The work proposed by [4] uses fuzzy logic to control the dynamic and nonlinearities of congestion on edge router queue of the network. A new queue control strategy is introduced for traffic flows to prevent packet drops and optimize congestion control. The outcome will be to determine the level of queue utilization and provide adequate level of congestion control. Additionally, the fuzzy approach has the ability to capture the bursting nature of traffic under dynamic change of network operations such as arrival rate and window size. Such control is difficult to achieve under a linear control system due to the control parameters of traffic flows. The proposed fuzzy model utilizes four inputs and one output shown in Fig 4. This uses compositional inference, for which input and output linguistic variables are defined in Table-1. [4].

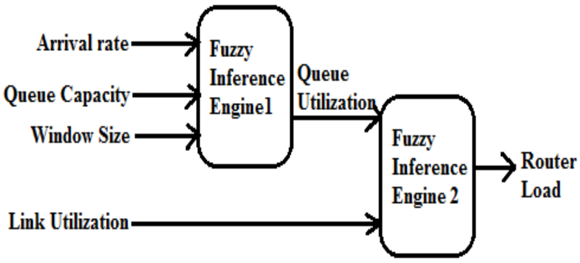

Fig 4: Proposed fuzzy model at edge routers

For performance evaluation, this intelligent fuzzy control based approach was compared against five well utilized conventional congestion controls – FIFO, RED, PQ, and WFQ & WRED. Results [4] shown in section 4(Fig 5 and 6), shows a significant improvement in queue delay time. Furthermore, edge router utilizes the bottleneck link in an efficient manner. The packet drop probability has also improved & stabilizes over time. The results clearly show that the proposed fuzzy congestion control outperforms well known conventional congestion control schemes.

Table 1. Linguistic variables for edge routers

Linguistic Variables	
Linguistic Variable	***Linguistic Value***
Arrival_ rate	Low Medium High
Queue_ capacity	Empty Moderate Full
Window_ size	Very small Small Medium Large Very Large
Link_ utilization	Underutilized Small Lightly overloaded Overloaded
Queue_ utilization	Underutilized Small Lightly overloaded Overloaded
Router_ load	Negligible Small Load Moderate Load Overloaded

3.3 Fuzzy-Priority Based Multiple Queue Management (FMQM) Algorithm

The current Internet is based on *best effort* service, where all traffic gets equal priority of bandwidth. Furthermore to support new Internet applications such as voice over IP, it is necessary to design effective Quality of Service approaches. A new fuzzy controller called Fuzzy-MQM [1] is used to avoid the congestion problem and to get higher performance. This structure has multiple queues that are designed for different packet priorities such as High, Medium and Low.

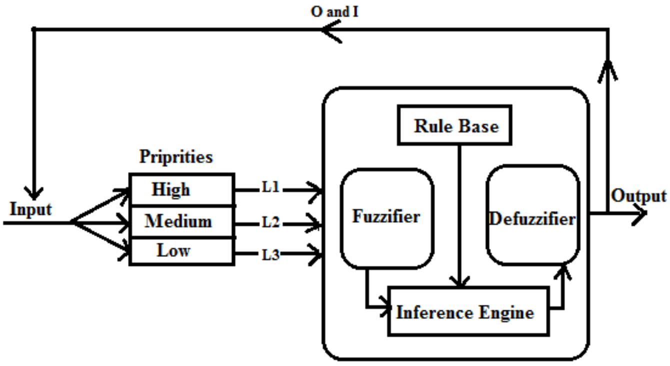

Fig 5: Fuzzy controller is used in FMQM algorithm

The Fuzzy-MQM controls the arrival and departure rates of packets with different priorities in order to save the router of congestion and overload. This subsequently prevents the network from congestion. In this algorithm if the arrival rate of packets with specific priority to a queue is high, then that queue quickly becomes full. In this situation, the controller shall send a message to the sender router concerning a decrease in the rate of sending packets with that specific priority. In addition, if a queue with specific priority be in threshold of filling, the process of sending packets from that queue shall be performed more than other queues.

Three queues are used in each input link attached to a router to keep packets with high, medium and low priority. The duty of the classifier is to separate the arriving packets based on their priority and to send them to the relevant queue. The scheduler is responsible for selecting packets that must be sent to the output link. When an input queue is full, the newly arrived packets are dropped. On the other hand, every packet has a specific deadline and the packet shall be sent before the deadline expires; otherwise, it will be dropped.

Table 2. Fuzzy Rule base for FMQM [1]

No	L 1	L 2	L 3	I	O
1	Empty	Empty	Full	High	Low
2	Empty	Half	Full	High	Low
3	Empty	Full	Empty	High	Med
4	Empty	Full	Half	High	Med
5	Empty	Full	Full	High	Med
6	Half	Empty	Full	Med	Low
7	Half	Half	Full	High	Low
8	Half	Full	Empty	Low	Med
9	Half	Full	Half	High	Med
10	Half	Full	Full	High	Med
11	Full	Empty	Empty	Med	High
12	Full	Empty	Half	Med	High
13	Full	Empty	Full	Med	High
14	Full	Full	Empty	Low	High
15	Full	Full	Half	Low	High
16	Full	Half	Full	Med	High
17	Full	Full	Empty	Low	High
18	Full	Full	Half	Low	High
19	Full	Full	Full	Low	High
20	Half	Half	Half	High	High
21	Empty	Empty	Empty	High	High

The fuzzy controller used in this algorithm, as shown in Fig.5, has three inputs, namely L1, L2 and L3 which are the status of high, medium and low priority queues. According to status (Empty, Half or Full) of each input queue the controller produces two types of outputs: I or O.

The fuzzy controller has a rule base, according to Table 2, which makes decisions on the basis of the rules in it. For example if the high priority queue is full, the medium priority queue is Empty and the low priority queue is full, then the inference engine searches the rule base and finds the Rule 13 of Table 2. The results of executing MQM and Fuzzy-MQM algorithms finds that the Fuzzy-MQM algorithm has lower dropped packets and deadline expirations and higher forwarded packets than the MQM algorithm.

4. COMPARATIVE STUDY ON THE BASIS OF RESULTS BETWEEN CONVENTIONAL AND FUZZY SYSTEM

Section 3 discussed the various congestion control methods using fuzzy logic. Following section gives the comparative result of the methods discussed in section III.

Table 3. Packet Loss for DT, RED and ADT-FL under varying Traffic Conditions [5]

$Traf_{int}$	AV_{bw}	DT $Q_{lim=10}$	Packet Sent	Loss%	RED	Packet Sent	Loss%	ADL-FL	Packet Sent	Loss%
7.1	3.66	2885	4876	42.01	2804	4924	39.8	2756	5147	36.56
8.13	7.32	3651	5771	46.26	3447	5780	42.48	3323	5830	39.85
8.49	4.62	3551	6026	41.77	3376	6019	38.97	3092	6305	32.48
8.72	2.33	5007	6200	67.72	3664	5099	56.08	3755	6101	44.45
10	1.92	4418	5502	66.37	4352	5502	65.42	4390	5613	64.89
8.89	2.19	3904	5117	61.67	3752	5074	58.66	3564	5133	53.17
9.79	1.94	4289	5494	64.02	4154	5386	62.76	4085	5402	60.79
7.65	1	4919	6013	69.21	4871	6031	67.73	4713	6026	64.21
8.14	3.74	3993	5143	63.45	3401	4551	59.65	3520	4795	57.99
6.9	1.29	3309	4459	58.99	3050	4200	57	2701	3979	51.37
8.47	1.45	4970	6011	70.47	4882	6032	67.97	3510	4660	60.41
10	3.25	4532	6413	54.64	4386	6517	50.71	4121	6335	48.20

Low packet loss gives high throughput, better quality of service and reduction in delay. Table 3 shows the comparisons between Drop Tail (DT), RED and fuzzy Adaptive Drop Tail (ADT–FL) at core router.

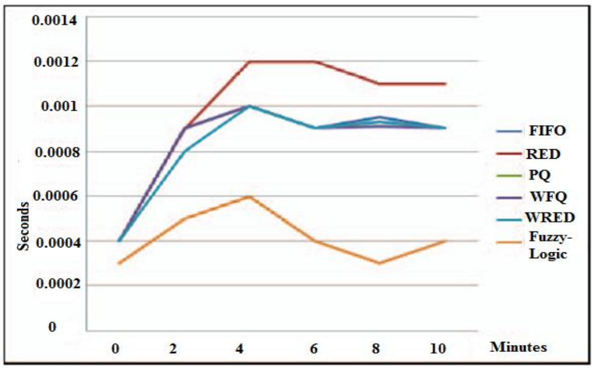

Fig 6: Queuing delay (time averaged) [4]

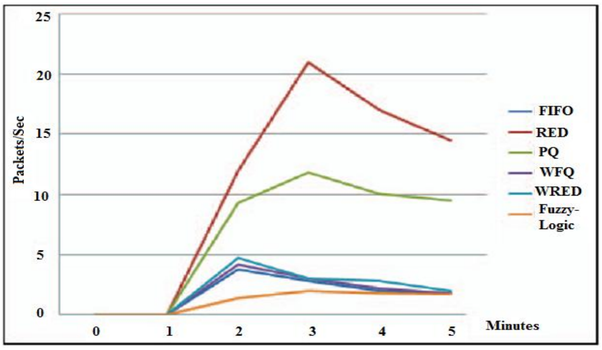

Fig 7: Packet drop at edge router (time averaged) [4]

The edge of the network plays an important role in converging flows and connecting networks together to allow packet to enter the core network. If congestion control at edge router is done using fuzzy controller which helps in reducing queuing delay and packet drop as shown in figures 6 and 7 respectively. In this fuzzy logic controller is compared against five well utilised conventional congestion control methods namely FIFO, RED, PQ, WFQ, WRED.

Queue management mechanisms plays vital role in congestion control. Multiple queue management provides better quality of service as it gives priorities to packets. Arrival rate of packets are dynamic so priorities queue management becomes very complex. Table 4 gives comparison between conventional Multiple Queue Management (MQM) and Fuzzy-MQM. Results in Table 4 shows higher number of packets forwarded, lower in number of dropped packets and deadline expiration than MQM.

Table 4. Numerical results of the Fuzzy-MQM and MQM algorithms [1]

		Fuzzy-MQM			MQM		
λ_i	Produce	Forward	Drop	Dead	Forward	Drop	Dead
2	10000	9800	70	40	9650	120	60
3	13300	11800	900	400	11350	1250	500
4	14700	11900	1850	850	10000	3300	1200
5	15900	11800	2750	1350	8350	5500	2300
6	16900	11600	3500	1700	6800	6200	3900
8	18000	11100	4400	2400	6000	7300	4600
9	18500	10700	4900	2700	5500	8250	4850
10	19000	10500	5300	2800	5000	8900	5000

5. CONCLUSION

Congestion control is important to prevent the network from overload and eventually from network collapse. There is a real challenge in the control of congestion in supporting video, voice and data applications simultaneously. Fuzzy logic techniques are expected to play an important role in non linear and non deterministic system.

This paper, reviewed existing literature on congestion control and use of fuzzy logic (FL) in controlling congestion in computer network. Discussed FL based methods show improvement in increased in throughput, reduction in delays and packet loss. Fuzzy Logic can be effectively use in controlling congestion at core as well as at bottleneck router. Output of one fuzzy system can be given as input to other fuzzy system. It can be also efficiently use in priority base fuzzy algorithms, which also gives better Quality of Service. Hope there will be more and more use of these techniques in future in controlling congestion in computer network.

6. ACKNOWLEDGMENT

We take this opportunity to thank the Principals and HODs-computer department of Smt. S.H.M.I.T., Ulhasnagar and P.I.I.T., New Panvel for their continuous motivation and support.

7. REFERENCES

[1] Behrouz Safaiezadeh, Amir Masoud Rahmani, Ebrahim Mahdipour " A New Fuzzy Congestion Control Algorithm in Computer Networks " (IEEE 2009 International Conference on Future Computer and Communication)

[2] Raj Jain "Congestion control in computer networks: Issues and Trends" (IEEE May 1990)

[3] Behrouz A. Forouzan "TCP/IP Protocol Suite " Fourth Edition (TATA McGraw- HILL Edition)

[4]Reginald Lal ,Andrew Chiou "Computational Intelligence Utilization in Simulation of Congestion Control in TCP/IP Edge Network" (IEEE 2010 fourth international conference)

[5] B.S. Kiruthika Devi, G. Preetha, S. Dina Nidhya, S. Mercy Shalinie "A Novel Fuzzy Congestion Control Algorithm for Router Buffers" (IEEE-International Conference on Recent Trends in Information Technology, ICRTIT 2011)

[6] Lotfi A. Zadeh "FUZZY LOGIC SYSTEMS: ORIGIN, CONCEPTS, AND TRENDS"

[7] Timothy J Ross "Fuzzy logic with Engineering application" second Edition –Wiley

[8] O. Al-Jaber, L. Guan, X. G. Wang, Irfan Awan, A. Grigg, X. Chi " Delay Restraining of Combined Multiple Input Cross Core Router" (IEEE - 22nd International Conference on Advanced Information Networking and Applications

[9] Anurag Aeron "Fine Tuning of Fuzzy Token Bucket Scheme for Congestion Control in High Speed Networks" (IEEE -2010 Second International Conference on Computer Engineering and Applications)

[10] Mehdi Mohtashamzadeh, Mohsen Soryani, Mahmoud Fathy " Fuzzy two time-scale congestion control algorithm" (IEEE - 2009 First International Conference on Computational Intelligence, Communication Systems and Networks)

[11] Xu Changbiao, Li Fengfeng "A Congestion Control algorithm of Fuzzy Control in Routers" (IEEE -2008)

[12] Darius Buntinas "Congestion Control Schemes for TCP/IP Networks "

[13] Habibullah Jamal, Kiran Sultan "Performance Analysis of TCP CongestionControl Algorithms" (IEEE - International Journal Of Computers And Communications)

[14] Hao Wang, Zuohua Tian, Qinlong Zhang " Sliding Mode Control with Fuzzy Reaching Law for Queue Management in the Internet" (IEEE -2010)

[15] Weijie Chen,Wanliang Wang,Yibo Jiang1, Guisen LI2, Qingli Chen1 "PFL: Proactive Fuzzy-logic-based AQM Algorithm for Best-effort Networks"(2010 International Conference on Computational Aspects of Social Networks.