CONGESTION CONTROL IN TCP/IP USING FUZZY LOGIC

Deepa Jose*

Dr.R.R.Mudholkar**

Abstract:

Congestion occurs in the network when arrival rate to a router is greater than its departure rate. In this paper, using fuzzy logic approach, we have proposed a modified TCP delay-based congestion avoidance mechanism which is based on traditional TCP-Africa algorithm. Here we present our fuzzy controller which is expected to act as a congestion controller in the routers. The proposed fuzzy controller uses queue delay and link capacity as input linguistic variables. The output of fuzzy controller is the window size to which the sending window must be adjusted. Both the current standard and most of the experimental congestion control methods are fundamentally loss-based, that is they rely on packet loss to detect that the network is above full capacity.

Index Terms: Fuzzy, TCP/IP ,Bandwidth, Congestion, Queue Delay ,Window Size

* Lecturer,MCA Dept, Y.M.T College of Management, Kharghar,Navi Mumbai.

** Reader,Electronics Dept, Shivaji University, Kolhapur.

INTRODUCTION:

This During past few years, computer networks have experienced an explosive growth in number of users and amount of traffic. The TCP (Transmission Control Protocol) congestion control was introduced into the Internet in the late 1980's by Van Jacobson[1].The rapid growth of the Internet and increased demand to use the Internet for time-sensitive voice andvideo applications necessitate the design and utilization of new Internet architectures to include more effectivecongestion control algorithms in addition to the TCP based congestion control. It has become clear (Braden et al., 1998) that the existing TCP congestion avoidance mechanisms and its variants, while necessaryand powerful, are not sufficient to provide good service in all circumstances. Basically, there is a limit to how much control can be accomplished from the edges of the network[2]. In the intervening years, TCP’s standard congestion control (Tahoe, Reno) has evolved in small ways, but no major redesign has taken place. Various problems have been shown, particularly in its achieved throughput on large bandwidth-delay product (BDP) networks and its propensity to cause high latency on network links with large available queues. In response to this, various experimental schemes have been proposed and implemented. In particular,CUBIC [Xu and Rhee, 2005] is now the default congestion control in Linux and Compound TCP [Tan et al., 2005] is now available in Microsoft Windows Vista. These algorithms have not as yet been ratified by the standards body.Both the current standard and most of the experimental congestion control methods are fundamentally loss-based, that is they rely on packet loss to detect that the network is above full capacity. It is presumed that the loss is caused by a full network queue. However, a second means of detecting incipient congestion proposed in the late eighties is to observe that the round trip time of a packet increases as network queues build [Jain, 1989]. This has the advantage of warning early rather than merely waiting until the network is over-utilised and packets are lost[3].

I. CONVENTIONAL AND FUZZY APPROACH:

TCP-Africa,is a new delay sensitive two-mode congestion avoidance rule for TCP that promises excellent utilization, efficiency, and acquisition of available bandwidth, with significantly improved safety, fairness, and RTT bias properties. This new protocol uses an aggressive, scalable window increase rule to allow quick utilization of available bandwidth, but uses packet round trip time measurements to predict eminent congestion events. Once the link is sensed to be highly utilized, the protocol reacts by resorting to the conservative congestion avoidance technique of standard TCP. An exponentially smoothed high accuracy round trip time estimate, aRTT is kept by the flow. This delay information, along with the minimum delay observed on the path minRTT, is used to estimate the queuing delay on the link[3]. W denotes the current congestion window of the flow.This delay metric, then, is used to enable or disable an aggressive window increase mode. We note that the conditions in which a delay based protocol would decrease its window are the same conditions where aggressive window increase rules are not desirable.

The function fast increase(W) is specified by a set of modified increase rules for TCP. Currently, we are using the congestion avoidance rules specified by HSTCP to specify the increase curve when the delay condition is met[4]. Alternatively, a multiplicative increase scheme could be used, such as fast increase(W) = 0.01×W.

The protocol utilizes a fuzzy logic controller that considers network related information to govern the application's sending rate while satisfying the user's needs. Using network information such as the available bandwidth, Packet Loss Rates (PLR), and Round Trip Times (RTT) a fuzzy inference system optimizes the application's send rate to meet the requested rate in a smooth manner without wasting network resources unnecessarily[5][6].

II. EXPERIMENT WORK:

The fuzzy logic controller is designed and its performance evaluated using MATLAB model simulations. The results indicate that the fuzzy controller solves the congestion collapse problem by reducing the number of undelivered packets into the network by nearly 100%.Fuzzy logic controllers, like expert systems, can be used to model human experiences and human decision making behaviors . In FLC the input-output relationship is expressed by using a set of linguistic rules or relational expressions. A FLC basically consists of four important parts including a fuzzifier, a defuzzifier, an inference engine and a rule base. As in many fuzzy control applications, the input data are usually crisp, so a fuzzification is necessary to convert the input crisp data into a suitable set of linguistic value that is needed in inference engine. Singleton fuzzifier is the general fuzzification method used to map the crisp input to a singleton fuzzy set. In the rule base of a FLC, a set of fuzzy control rules, which characterize the dynamic behavior of system, are defined. The inference engine is used to form inferences and draw conclusions from the fuzzy control rules. Fig. 1 shows the fuzzy logic controller architecture.

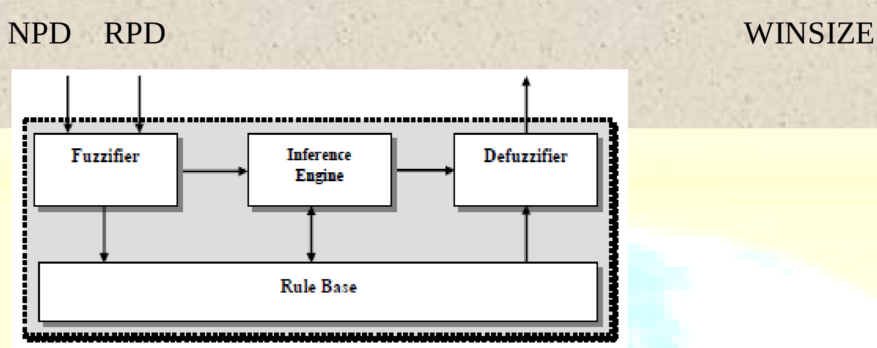

Fig 1 Fuzzy Logic Controller Architecture

Our approach to the Fuzzified TCP-Africa is Fuzzy-TCP Africa, a fuzzy logic controlled window . To implement Fuzzy- TCP Africa we use the dynamic network state dependant thresholds calculated using an FIE (Pitsillides et al., 1997). source utilization in the presence of dynamic network state changes, for example, the number of active sources. The FIE dynamically calculates the sender side window size based on two inputs: the number of packets dropped (NPD) and the rate of the change in packet drop(RPD). The FIE uses separate linguistic rules to calculate the sender window size based on the input from the network.

The inference rules applied are as in Fig 2:

1. If (NPD is Zero) and (RPD is Decreasing) then (WINSIZ is FastInc) (1)
2. If (NPD is Zero) and (RPD is Zero) then (WINSIZ is FastInc) (1)
3. If (NPD is Zero) and (RPD is Increasing) then (WINSIZ is smalinc) (1)
4. If (NPD is SL) and (RPD is Decreasing) then (WINSIZ is maintain) (1)
5. If (NPD is SL) and (RPD is Zero) then (WINSIZ is slowdec) (1)
6. If (NPD is SL) and (RPD is Increasing) then (WINSIZ is slowdec) (1)
7. If (NPD is L) and (RPD is Decreasing) then (WINSIZ is maintain) (1)
8. If (NPD is L) and (RPD is Zero) then (WINSIZ is slowdec) (1)
9. If (NPD is L) and (RPD is Increasing) then (WINSIZ is slowdec) (1)
10. If (NPD is SH) and (RPD is Decreasing) then (WINSIZ is slowdec) (1)
11. If (NPD is SH) and (RPD is Zero) then (WINSIZ is fastdec) (1)
12. If (NPD is SH) and (RPD is Increasing) then (WINSIZ is fastdec) (1)
13. If (NPD is H) and (RPD is Decreasing) then (WINSIZ is fastdec) (1)
14. If (NPD is H) and (RPD is Zero) then (WINSIZ is fastdec) (1)
15. If (NPD is H) and (RPD is Increasing) then (WINSIZ is fastdec) (1)

Fig 2: The inference rules applied

Usually, multi-input FIEs can offer better ability to linguistically describe the system dynamics. It is expected that the system is tuned to yield results closer to optimum, and improve the behavior of traditional TCP -Africa. The dynamic way of calculating the window size by the FIE comes from the fact that according to the rate of change of the packets, a different set of fuzzy rules, and so inference apply. Based on these rules and inferences, the window size is calculated more dynamically than the classical TCP Africa approach.

RESULTS AND DISCUSSION:

The selection of rule base is based on our experience and beliefs on how the system should behave (in other words, human expert's heuristic knowledge). Design of a rule base is two-fold: First, the linguistic rules (''surface structure'') are set; afterwards, membership functions of the linguistic values (''deep structure'') are determined. Usually, to define the linguistic rules of a fuzzy variable, Gaussian, triangular or trapezoidal shaped membership functions are used.

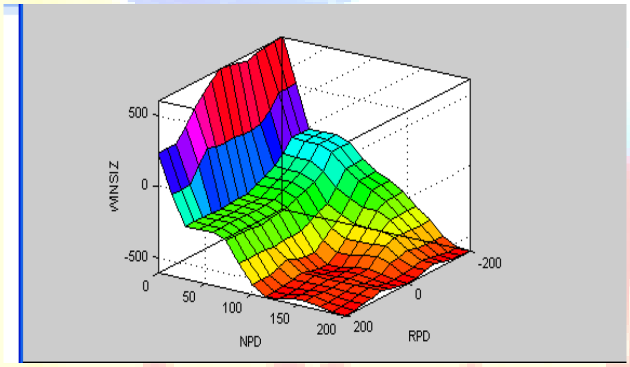

Fig 3. The input output relationship of fuzzified TCP- Africa

Since triangular and trapezoidal shaped functions offer more computational simplicity, we have selected them for our rule base.Then, the rule base is fine tuned by observing the progress of simulation, such as packet loss occurrences and rate of change inpacket loss .The Decision surface of winsize is decipted in fig.

In short, the results have shown that the fuzzy engine may indeed give better congestion from channel error conditions, and consequently assist the TCP error detection. However, improvements are certainly possible as the model here studied is rather modest. For instance, independent fuzzy outputs for each of the valuated conditions (congestion) could provide more flexibility in adjusting the engine. The membership functions can be optimized by using advanced learning/training techniques such as ANFIS , and self adaptive setting models can render our approach very robust. We have introduced and evaluated a fuzzy logic engine for supporting congestion control in TCP/IP networks. The architecture of the fuzzy enhanced controller has been explained, and its primary features discussed. The main conclusion is that efficiency can be obtained provided that the input data are taken precisely enough to reflect the actual changes inside the network

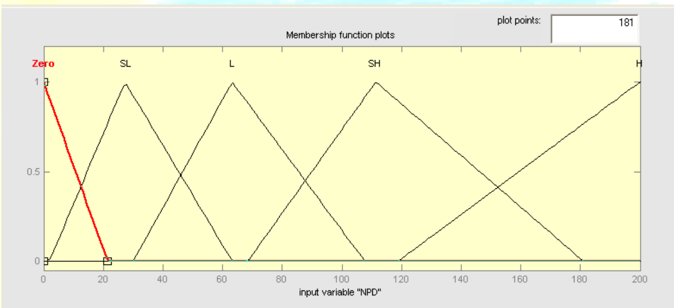

Fig 4: Membership functions of the linguistic values representing the input variable NPD

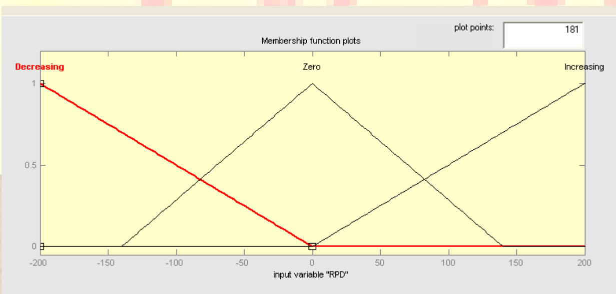

Fig 5: Membership functions of the linguistic values representing the input variable RPD.

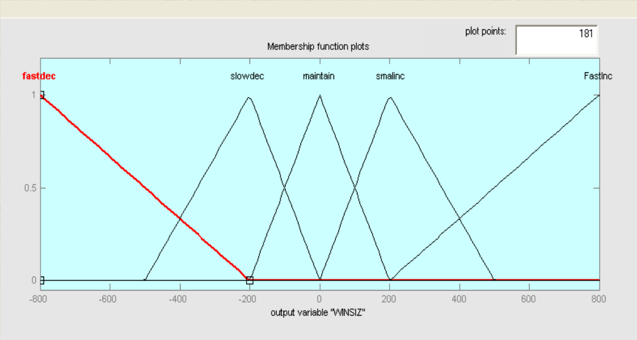

Fig 6: Membership functions of the linguistic values representing the output variable WINSIZE

Efficiency can be obtained provided that the input data are taken precisely enough to reflect the actual changes inside the network. . The proposed mechanism will be quite robust in dealing with steady scenarios, where abrupt changes are not too frequent, for instance a lossy channel. We proposed supporting schemes for accelerating our mechanism, which may boost the performance of our model as a whole. Therefore, the overall evaluation is positive in the sense that we applied an intelligent algorithm for inferring statistically the internal state of the network, and the outcome was surprisingly accurate. However, different scenarios and more elaborate inference models have to be checked to render our proposal even more generic.

REFERENCES:

- V. Jacobson, "Congestion avoidance and control," ACM SIGCOMM, pp. 314–329, 1988.
- S. Floyd, "Highspeed tcp for large congestion windows," RFC 3649, Experimental, Dec. 2003.
- J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: A simple model and its empirical validation," ACM SIGCOMM, 1998.
- C. Jin, D. X. Wei, and S. H. Low, "FAST TCP: motivation, architecture, algorithms, performance," IEEE INFOCOM, 2004.
- M. Kwon and S. Fahmy, "A comparison of loadbased and queue-based active queue management algorithms", in Proc. SPIE ITCom, vol. 4866, Aug. 2002.

- M. H. Yaghmaee, M. Menhaj, H. Amintoosi, "A fuzzy extension to the blue active queue management algorithm", IAEEE, Journal of Iranian Association of Electrical and Electronics Engineers, vol. 1, No. 3,Winter 2005, pp. 3-14.
- L.A. Zadeh, "Fuzzy sets", Information and Control, 1965. 338- 353
- W. Feng, et.al.,"Stochastic fair blue: a queue management algorithm forenforcing fairness", In Proc. of IEEE INFOCOM, Apr 2001.
- D. Chiu, R. Jain, "Analysis of the increase and decrease algorithms for congestion avoidance in computer networks", Computer Networks and ISDN systems, 17:1-14, 1989.
- Frank P. Kelly, "Fairness and stability of end-to-end congestion control," European Journal of Control, vol. 9, pp. 159–176, 2003.

BIOGRAPHIES:

Deepa Jose did her BCA from St.Teresa's College under Mahatma Gandhi University ,Ernakulam.She earned her MCA degree from Bharathidasan University, Trichy, in 2004 and is currently working towards the Ph.D degree.She is a Lecturer in Y.M.T College of Management Navimumbai , Maharashtra. She has about 6 Research papers published in National and International Journals. Her research interest includes Computer Communication Networks and Artificial Intelligence.

Dr. R. R. Mudholkar is currently working as an Associate Professor in the Department of Electronics, Shivaji University, Kolhapur (MS), INDIA. He completed his M.Sc. (1984), M.Phil. (1996) and Ph.D. in 2003. His areas of Research include Fuzzy Systems, ANN and GA based Fuzzy Systems. He has about 30 Research papers published in National and International Journals. Two students have awarded with Ph.D. and two M.Tech. under his guidance. Currently eight students are working in the areas of Fuzzy based Industrial process control and Cloud computing.