Congestion Control in Computer Networks using Fuzzy Logic

C. CHRYSOSTOMOU, A. PITSILLIDES

Department of Computer Science,
University of Cyprus,
75 Kallipoleos Street, P.O. Box 20537, 1678 Nicosia
CYPRUS

http://www.cs.ucy.ac.cy/networksgroup

Abstract: Network congestion control is a complex problem that requires robust, possibly intelligent, control methodologies to obtain satisfactory performance. Designing effective congestion control strategies for computer networks is known to be hard because of the difficulty of obtaining realistic, cost effective, tractable analytical models. This renders the application of classical control system design methods, which rely on availability of these models, very hard, and possibly not cost effective. Computational Intelligence employing Fuzzy Logic Control methodology is reported to offer effective solutions for certain classes of control problems. It is particularly appealing in non-linear complex systems where satisfactory analytic models are costly or impractical to obtain, but where their behaviour is well understood and can be captured by linguistic models. Consequently, a number of researchers have looked at fuzzy logic in order to devise effective, robust congestion control techniques. In this paper, we discuss several control approaches currently in use, before we motivate the utility of Fuzzy Logic based control. Then, through a number of examples, we illustrate the power of the methodology by the successful application of fuzzy based congestion control in the two diverse networking technologies of ATM and TCP/IP.

Key-Words: Fuzzy Logic, Congestion Control, Active Queue Management, ATM, TCP/IP, Quality of Service.

1 Introduction
The problem of network congestion control remains a critical issue and a high priority, especially given the growing size and demand of networks, despite the many years of research efforts. Congestion may become unmanageable unless effective, robust, and efficient methods of congestion control are developed. This assertion is based on the fact that despite the vast research efforts, spanning a few decades, and the large number of different control schemes proposed, there are still no universally acceptable congestion control solutions. Current solutions in existing networks are increasingly becoming ineffective, and it is generally accepted that these solutions cannot easily scale up. Therefore, we propose that Computational Intelligence could have an essential role to play in designing such a complex control system. In Section 2 of the paper we discuss the problem of congestion in networks, and review recent approaches on congestion control in the diverse worlds of Internet and ATM (Asynchronous Transfer Mode). In Section 3, we present our Fuzzy Logic Methodology for congestion control applied in both TCP/IP and ATM. Section 4, presents some indicative simulation evaluation, and finally in section 5 we offer our conclusions and future work.

2 Congestion Control
2.1 How to Define Congestion Control
Congestion is a complex process to define. It is felt by a degradation of performance. Despite the many years of research efforts in congestion control, there is no unique approved, by the network research community, definition of congestion. Currently, there is an ongoing discussion between the active members of the networking community as to give the right definition for congestion [1]. However, it cannot be argued that the effect of network congestion is a degradation in the network performance. The user experiences long delays in the delivery of data, perhaps with heavy losses caused by buffer overflows. Thus, there is degradation in the quality of the delivered service, with the need for retransmissions of packets (for services intolerant to loss). In the event of retransmissions, there is a drop in the throughput,
which leads to a collapse of network throughput, when a substantial part of the carried traffic is due to retransmissions.

The difficulty of the congestion control problems has caused a lot of debate as to what are appropriate control techniques for the control of congestion, and depending on one’s point of view, many different schools of thought were followed, with many published ideas and control techniques.

In the next two subsections we highlight some aspects of congestion control control approaches proposed for ATM and TCP/IP. No attempt is made to be comprehensive. Rather our intention is to motivate the fuzzy logic based approach.

2.2 Congestion Control in ATM Networks

Congestion control in ATM based networks has been extensively researched. The complexity and immensity of the task was recognized early. Initially, there was a push for preventive control (i.e., open loop type controls). This view was influenced by a predominant view that controls must reside at the edges of the network. Note that many researchers, even at the early stage, did not adopt that view [2]. Progressively, there was a shift from that view that feedback is essential for effective control, and finally that controls inside the network should not be precluded, at least to supplement preventive controls. Several feedback based control schemes were proposed for delay tolerant traffic, including: EPRCA [3], ERICA [4], and Predictive Adaptive control [5-6].

The ATM community concentrated its efforts on a mechanism to allocate bandwidth dynamically within an ATM network, while simultaneously preventing data loss. This effort culminated in the introduction of a service category by the ATM Forum, called available bit rate (ABR). A feedback control framework was selected to achieve these aims [7]. The proposed framework allows downstream nodes to periodically send information to the traffic sources relating to maximum cell rates that they can handle. The cell rate information is carried by a stream of resource management (RM) cells. While these cells pass through the switching nodes, the cell rate information contents of these cells are dynamically updated by the intermediate systems. This introduces a framework for multivalued feedback, which can be quickly send back to the source (sent by congested switches, i.e. where is is sensed).

2.3 Congestion Control in TCP/IP Networks

The congestion control schemes employed by the TCP/IP protocol have been widely studied. It has become clear [8] that the existing TCP congestion avoidance mechanisms, while necessary and powerful, are not sufficient to provide good service in all circumstances. Basically, there is a limit as to how much control can be accomplished from the edges of the network. Some mechanisms are needed in the routers to complement the end-point congestion avoidance mechanisms. Thus, Active Queue Management (AQM) mechanisms have been introduced to assist the TCP congestion control.

The AQM approach can be contrasted with the “Tail Drop” (TD) queue management approach, employed by common Internet routers, where the discard policy of arriving packets is based on the overflow of the output port buffer. Contrary to TD, AQM mechanisms [8] start dropping packets earlier in order to be able to notify traffic sources about the incipient stages of congestion. AQM allows the router to separate policies of dropping packets from the policies for indicating congestion. The use of Explicit Congestion Notification (ECN) [13] was proposed in order to provide TCP an alternative to packet drops as a mechanism for detecting incipient congestion in the network. The ECN scheme requires both end-to-end and network support. An AQM-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. However, for ECN the adopted feedback scheme is based on a single bit, and thus it is not as rich as for ATM ABR.

Given that, many AQM schemes have been proposed to provide high network utilization with low loss and delay by regulating queues at the bottleneck links in TCP/IP best-effort networks, including random early detection (RED) [9], adaptive RED (ARED) [10], proportional-integral (PI) controller [11], and random exponential marking (REM) [12].

However, these AQM mechanisms still require a careful configuration of non-intuitive control parameters. Most of the AQM schemes have been designed by taking into account a simple network containing a single congested router. As indicated in Section 4, taking into account a tandem network with multiple congested AQM routers (which gives a more realistic picture of today’s TCP/IP networks), these schemes are often non-robust to dynamic network changes, and as a result, they exhibit greater delays than the target mean queuing delay with a large delay variation, and large buffer fluctuations, and consequently cannot effectively control the router queue.
3 Fuzzy Logic based Congestion Control

3.1 Fuzzy Logic

Fuzzy logic is one of the tools of what is commonly known as Computational Intelligence (CI). CI [14] is an area of fundamental and applied research involving numerical information processing. While these techniques are not a panacea (and it’s important to view them as supplementing proven traditional techniques), we are beginning to see a lot of interest not only from the academic research community [15], but also from industry [16]. Fuzzy Logic Control (FLC) may be viewed as a way of designing feedback controllers in situations where rigorous control theoretic approaches cannot be used due to difficulties in obtaining a formal analytical model, while at the same time some intuitive understanding of the process is available. The control algorithm defines a non-linear control law, and is encapsulated as a set of linguistic rules. FLC has been applied successfully [17] for controlling systems in which analytical models are not easily obtainable or the model itself, if available, is too complex and possibly highly nonlinear.

In recent years, a number of research papers using fuzzy logic investigating solutions to congestion control issues, especially to ATM networks, have been published. Given the complexity of ATM networks, rich variety of traffic sources that operate on them, and difficulty of obtaining formal models for in depth analysis, it is not surprising to see that FLC was favored by many researchers. For example, Pitsillides et al (e.g. [18-20], since early 90’s, have successfully used the concept of FLC for congestion control in ATM, as an alternative to the conventional counterparts. A survey is also given in [15]. Moreover, fuzzy logic is recently applied to TCP/IP best-effort networks [21-22], and also to TCP/IP Differentiated Services Networks [24], providing, in both cases, effective congestion control in diverse networking technologies.

3.2 Fuzzy Explicit Rate Marking (FERM) in ATM Networks

The proposed FERM scheme [20] is compliant with the ATM Forum Traffic Management Specification, version 4. Cell rates of data sources are adjusted by Explicit Rate (ER) information carried by Resource Management cells. The scheme, in the calculation of ER, monitors both the current queue length and its growth rate. Based on these two inputs, the fuzzy controller computes the Flow Rate Correction (FRC - values between -1 and 1), and an Explicit Rate

\[ ER_{next} = ER_{current} + (FRC \times \text{Link Cell Rate}) \]

for the sources feeding the ATM switch. Periodical ER calculations are performed by the Fuzzy Logic Controllers located in each ATM switch. If, within the current control interval, the ATM switch receives a RM cell traveling to the upstream nodes, it examines the ER field of the cell, and if this rate is greater than the calculated flow rate, it modifies the ER field with the calculated value and retransmits the RM cell. Fig. 1 shows the block diagram of FERM. As can be observed from the control surface of FERM (Fig. 2), it is a non-linear controller. For a certain queue length, it calculates different flow rate limits depending on the rate at which queue length varies.

3.3 Fuzzy Rate Marking (FEM) in TCP/IP networks

Our design of a fuzzy control system in TCP/IP networks [21-22, 24] is based on a fuzzy logic controlled AQM scheme to provide congestion control in best-effort networks. The system model of FEM is shown in Fig. 3, where all quantities are considered at the discrete instant \( kT \), with \( T \) the sampling period, \( e(kT) = q_{des} - q \) is the error on the controlled variable queue length, \( q \), at each sampling period, \( e(kT - T) \) is the error of queue length with a delay \( T \) (at the
previous sampling period), $p(kT)$ is the mark probability, and $SG_i$ and $SG_o$ are scaling gains.

The proposed fuzzy control system is designed to regulate the queues of IP routers by achieving a specified desired TQL, $q_{des}$, in order to maintain both high utilization and low mean delay. A fuzzy inference engine (FIE) is designed to operate on router buffer queues, and uses linguistic rules to mark packets in TCP/IP networks. As shown in Fig. 3, the FIE dynamically calculates the mark probability behavior based on two network-queue state inputs: the error on the queue length (i.e., the difference between the desired (TQL) and the current instantaneous queue length) for two consecutive sample periods (which can be interpreted as a prediction horizon). We have implemented FEM with marking capabilities, so that FEM routers have the option of either dropping a packet or setting its ECN bit in the packet header, instead of relying solely on packet drops (for the rest of the paper, by marking a packet it is meant setting its ECN bit). The decision of marking a packet is based on the mark probability, which is dynamically calculated by the FIE.

The FIE uses linguistic rules to calculate the mark probability based on the input from the queues. Usually multi-input FIEs can offer better ability to linguistically describe the system dynamics. We expect that we can tune the system better, and improve the behavior of the queue, by achieving high utilization, low loss and delay. The dynamic way of calculating the mark probability by the FIE comes from the fact that according to the error of queue length for two consecutive sample periods, a different set of fuzzy rules, and so inference apply. Based on these rules and inferences, the mark probability is calculated more dynamically than other AQM approaches [9-12].

This point can be illustrated by observing the visualization of the decision surface of the FIE used in the FEM scheme (see Fig. 4). An inspection of this decision surface provides hints on the operation of FEM. The mark probability behavior under the region of equilibrium (i.e., where the error on the queue length is close to zero) is smoothly calculated. On the other hand, the rules are aggressive about increasing the probability of packet marking sharply in the region beyond the equilibrium point. These rules reflect the particular views and experiences of the designer, and are easy to relate to human reasoning processes and gathered experiences.

The design of FEM aims to provide better congestion control and better utilization of the network, with lower losses and delays than other AQM schemes [9-12], especially by introducing additional input variables and on-line (dynamic) adaptivity of the rule base (self-tuned). The FEM controller has been evaluated in [21-22] considering both a single congested AQM router, and more realistic scenarios like a tandem network with multiple congested routers [24]. Based on these results, the FEM scheme outperforms a number of representative AQM schemes in terms of regulated queues, packet losses, and link utilization.

4.1 FERM evaluation

Extensive simulations, using OPNET Modeler [25], on a representative ATM network have been done [20], and a comparison is made between FERM and EPRCA. The results have been reported in [20]. Due to lack of space, we show here an indicative simulation result (see Fig. 5 and Fig. 6), where we can observe the transient responses of ATM LAN under FERM and EPRCA control, respectively. FERM offers excellent transient behaviour with good rise time, good settling time, and insignificant, if any, oscillations. Its transient behaviour is much better than EPRCA, in the sense that FERM attains steady-state much faster, and that it offers “smooth” control (no or negligible oscillations present).

FERM is also shown [20] to exhibit many desirable properties, like robustness, fairness, and scalability.
4.2 FEM evaluation

We have evaluated the performance and robustness of the recently proposed fuzzy logic based scheme, namely FEM AQM [21-22, 24], in a wide range of environments, and compare with other published results by taking some representative AQM schemes, namely A-RED [10], PI controller [11] and REM [12], using NS-2 [26] simulator. The performance evaluation examines the influence of both network and AQM parameters. In particular, the following have been examined: dynamic traffic changes, traffic load factor, heterogeneous propagation delays, introduction of short-lived TCP connections, different types of data streams, like FTP and Web-like, as well as unresponsive UDP traffic, use of single- and multiple-bottleneck links, and various target queue lengths in order to examine the sensitivity of the AQM algorithms. The performance metrics used to compare the AQM schemes are the useful throughput, loss rate, and the mean queuing delay and its standard deviation.

Due to lack of space, we show here some indicative simulation results. Fig. 7 shows the loss rate as the total traffic load increases, in a network topology with multiple bottleneck links, where it can be seen that FEM has the lowest drops (experiences minimal losses, much below 1%). FEM shows stable and low packet loss over large traffic load. A-RED has the largest drops with a large increase of packet loss with respect to higher loads. Fig. 8 shows the utilization of one of the bottleneck links with respect to the mean queuing delay. FEM outperforms other AQM schemes on both high utilization and low mean delay, thus it exhibits a more stable, and robust behavior, even though the traffic load is increasing. The other AQM schemes show a poor performance as the number of total traffic load increases, achieving much lower link utilization, and large queuing delays, far beyond the expected value. FEM also has the lowest variance in queuing delay, resulting in a robust behavior. On the other hand, the other AQM schemes exhibit very large queue fluctuations with large amplitude that inevitably deteriorates delay jitter.

5 Conclusions
In this paper we have presented a fuzzy logic control methodology that is applied in two diverse technologies: ATM and TCP/IP networks for congestion control. The design of the fuzzy knowledge base is kept simple, using a linguistic interpretation of the system behavior. We have successfully used the reported strength of fuzzy logic (a CI technique) and have addressed limitations of existing alternative mechanisms. This is clearly shown from the extensive simulative evaluation [20, 21-24]. Both Fuzzy Logic based controllers are shown to exhibit many desirable properties, like robustness, fast system response and fairness, with capabilities of adapting to highly variability and uncertainty in network.

From the results presented, using simple designs, we are optimistic that the Fuzzy Control methodology can offer significant improvements on controlling congestion in computer networks. Various enhancements of the proposed fuzzy based congestion control designs, such as adaptivity, as well as the formal evaluation of the properties of the controllers are currently being investigated.

References: