FLoCAD: Fuzzy Logic Controlled Video Streaming with Congestion Avoidance using Delay and Packet Loss.

E. Jammeh, M. Fleury, and M. Ghanbari

ESE Department, University of Essex, UK

Keywords: Video networking, congestion control, fuzzy logic.

Abstract

Accuracy in determining congestion level is a big factor in designing efficient rate adaptive networked multimedia. This paper proposes measuring one-way delay of video packets to serve as incipient network congestion indicators. The delay is potentially used in conjunction with packet loss rate as input to a fuzzy logic controller. The controller efficiently adapts the sending rate of the application to the level of network congestion. The paper demonstrates that using delay to signal imminent congestion and packet loss to signal full-blown congestion is a means of accurate and timely adaptation to varying network conditions.

1 Introduction

Networked video communication is reliably achieved by determining the available bandwidth and adapting the video rate to it. It is paramount that the network state be accurately and timely determined. Packet loss has traditionally been used by TCP to signal congestion with remarkable success in avoiding excessive Internet congestion. However, gauging the network congestion level from packet loss has limitations, given that it signals full-blown congestion rather than imminent congestion. Furthermore, packet loss as an indicator of congestion becomes completely inadequate as the delay-bandwidth product of a network increases, because by the time congestion is detected, severe congestion might already have occurred [1]. For transport of encoded video, any packet loss is significant, as the effects are felt in subsequent pictures until the next intra-coded picture at the end of a group-of-pictures. The limitation of packet loss as a congestion signal has been identified as a performance bottleneck in TCP and many enhancements to the protocol have been proposed, for example [4] [11] [18]. However, in video streaming over the Internet, UDP is normally applied, as TCP's reliability mechanisms have the potential for unbounded delivery latency, leading to missed display and/or decode deadlines. This paper newly applies delay-based congestion control to video transport.

The main contribution of this paper is to demonstrate that packet delay is a suitable congestion control input to a fuzzy logic congestion controller. The one-way delay (OWD) of a stream of packets is the sum of the path delay d_p and the queuing delay d_q. Queuing delay is a function of the network load or congestion level. Maximum queuing delay occurs when the buffer at the bottleneck link is full and packets start being dropped: full blown congestion.

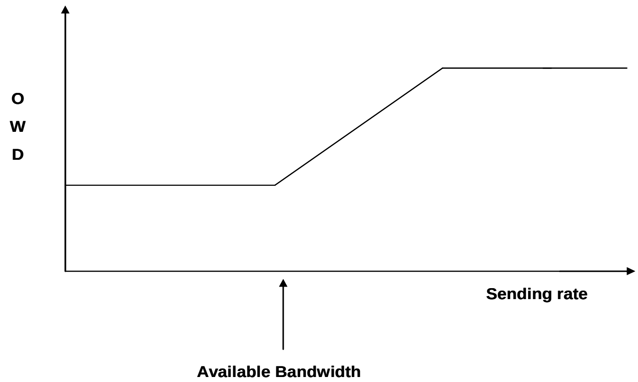

Figure 1: The relationship between delay, sending rate and the available network bandwidth.

In Figure 1, the OWD of packets is at a minimum when the sending rate is less than the available network bandwidth and OWD starts increasing above the minimum when the sending rate exceeds the available bandwidth at the bottleneck link. A further increase in the sending rate results in an increase in OWD until a maximum is reached, which is a function of the buffer size at the bottleneck link. The OWD of packet *i*, d_i, is given by,

$$d_i = d_p + d_q, \quad (1)$$

where d_q is given by,

$$d_q = \frac{q \times L}{C}, \quad (2)$$

where q is the instantaneous queue size at the bottleneck link, L is the packet length and C is the capacity of the bottleneck link.

Minimum OWD occurs when there is no queuing delay, equation (3), and the maximum OWD occurs, equation (4), when the queue size at the bottleneck buffer is at a maximum, owing to network congestion:

$$d_{\min} = d_p \quad (3)$$

$$d_{\max} = d_p + \frac{q_{\max} \times L}{C}. \qquad (4)$$

The region of increasing delay between d_{min} and d_{max} is represented by congestion level CL in (5)

$$CL = \frac{d_i - d_{\min}}{d_{\max} - d_{\min}} \qquad (5)$$

$$CL = \frac{d_p + \frac{q \times L}{C} - d_{\min}}{d_p + \frac{q_{\max} \times L}{C} - d_{\min}} = \frac{q}{q_{\max}} \qquad (6)$$

Following from (5), equation (6) shows that the congestion level can be used as a buffer fullness measure.

Fuzzy logic control is a convenient tool for handling un-modeled network congestion states. It allows the intuitive nature of congestion reduction to be applied through linguistic variables. Fuzzy congestion control is a sender-based system for unicast flows. The receiver returns a feedback message that indicates averaged delay. This allows the sender to compute the network congestion level. The sender then applies a control signal to a transcoder's quantization level, as a reflection of the anticipated congestion. Thus, congestion control is achieved through delay feedback, especially at tight links, representing the point of minimum available bandwidth on the network path. Rather than a single, invariant throughput equation to emulate TCP's behaviour [11], fuzzy logic control, provides a flexible method of adapting to evolving Internet traffic patterns, as precise mathematical representations are difficult to devise for such a complex, non-linear control problem [13].

2 Related work

The main intention of delay-based control is to avoid the need to rely on detrimental packet losses, and, therefore, strictly delay-based control is a form of congestion avoidance [8][9], as when congestion is established packet loss is frequent, with effects that endure until the next decoder synchronization point. In [5], delay-based congestion avoidance, using the delay gradient [17], was employed for interactive applications such as video conferencing to find the minimum possible delay without overly restricting throughput. In the process it was found that output oscillations were reduced along with delay variance. The method also avoided the 'phase effect' [6], whereby packet-loss probe-based congestion control introduces unfairness between streams across the same link, as the same stream may repeatedly suffer packet loss at the congested link owing to an ordering effect.

In the TCP world, TCP Vegas [4] seeks to avoid the TCP loss penalty by reacting to changes in round-trip time (RTT). However, the correlation of RTT with potential packet loss from congestion remains unresolved [3], and, hence, OWD may be a better indicator. Congestion control for high delay-bandwidth product networks [9][11][18][19] may well be delay-based, because of the need to anticipate congestion well in advance of packet loss feedback. In these networks, packet loss feedback simply arrives too late to be of value. For TCP, delay-based and loss-based congestion control are not incompatible, and in [11] loss- and delay-based control are combined. We also believe that some form of loss-based moderation of a purely delayed-based approach will become necessary, if only when despite all efforts at control, loss still occurs.

A survey of congestion control through computational intelligence, [12] observes that not much work has been reported on deploying natural algorithms within the Internet. ATM networks, which employ access control to virtual circuits, are one domain to which fuzzy logic has been more extensively applied [7]. For example, in [10], fuzzy control was applied to Available Bit-Rate (ABR) class traffic, as there is uncertainty about the input, and the queue service rate is variable. Feedback to the fuzzy controller arises from: output queue length; rate of change of queue length; and number of lost ATM cells. Because of its resemblance to ATM admission control, the authors of [14] have explored fuzzy logic to improve the performance of the Random Early Discard (RED) Internet router queue algorithm, including Explicit Congestion Notification (ECN), and research in [16] considers DiffServ buffer occupancy for each class of layered video packets.

3 Methodology

3.1 FloCAD system

Figure 2 is a high-level view of the proposed FloCAD system. It shows a fuzzy logic congestion controller which takes as its inputs the congestion level, *CL*, its rate of change, packet loss rate, *PL*, and the current sending rate quality of the video, *RS*, to determine the rate of change of the current video sending rate, *Ctrl*. In practice, a modular fuzzy logic controller would be applied in which a two-input fuzzy logic controller, with inputs *CL* and ∂CL, produce an output, which, with *PL* serves, as inputs to a second controller. Therefore, *PL* acts as a modification of the *Ctrl* output produced purely by delay. The output of this controller is subsequently modified in a third fuzzy logic controller by *RS*.

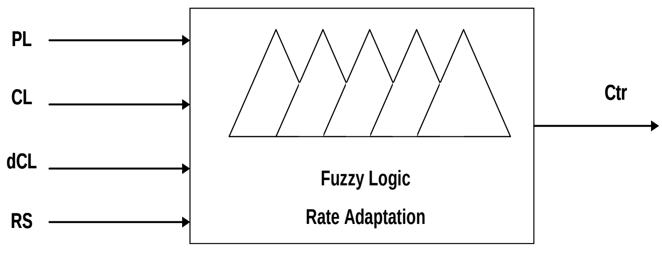

Figure 2: Block diagram depicting the inputs and outputs to a fuzzy congestion controller.

The output of the fuzzy logic rate adaptation unit controls the quantization level of a frequency-domain rate controller [2], while a video decoder at the client decodes the received video stream in real time.

3.2 Delay-based implementation

This paper demonstrates that delay by itself is an adequate means of congestion control. The average OWD (d_{av}) is computed for each frame from the instantaneous OWD of all N packets of a frame,

$$d_{av} = \frac{\sum_{i=1}^{N} d_i}{N} \qquad (7)$$

where N is the number of packets in a frame and d_i is the OWD of the i^{th} packet in the frame. Measurements of the minimum and maximum OWD are continuously updated during the streaming session. The current values are then used to compute the congestion level, *CL*. The rate of change of the *CL* is simply found from (8):

$$\partial CL_{n+1} = CL_n - CL_{n+1} \qquad (8)$$

The measured congestion level and its rate of change act as inputs to the controller which fuzzifies them into suitable linguistic variables. The controller employs a simple Mamdani inference model and centroid-of-area defuzzification. Triangular membership functions are used at the input, as shown in Figures 3 and 4. As is normal procedure, triangular functions are chosen for real-time computation, though sharp transitions are present. The linguistic variable employed in Figures 3 and 4 are listed in Table 1. The output of the fuzzy controller is a control value (*Ctrl*) that is used to control the sending rate. For a current sending rate R_n with a computed control signal of *Ctrl*, a new sending rate R_{n+1} is computed as follows:

$$R_{n+1} = R_n + \partial R_n \qquad (9)$$

where

$$\partial R_n = (1 - R_n') \times Ctrl \qquad (10)$$

and R_n' is the normalised current sending rate. Normalisation to a range (0,1] is by the known target encoding rate of the video. The intention of the procedure in (10) is to modify the control signal *Ctrl* according to the magnitude of the current sending rate. A higher current sending rate has a reduced effect on *Ctrl*.

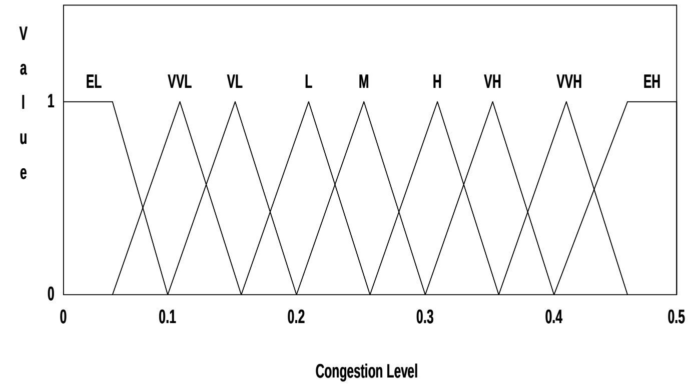

Figure 3: The membership functions for the measured congestion level, CL.

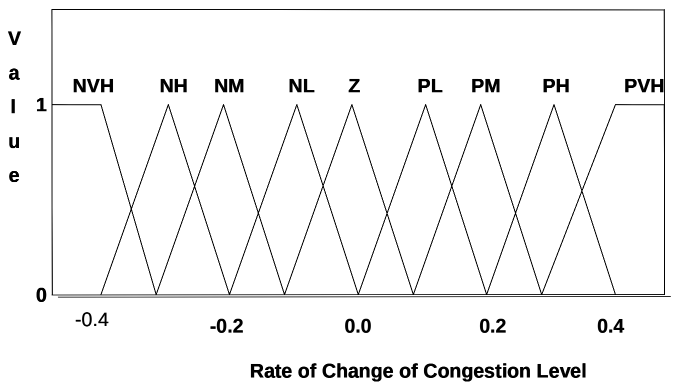

Figure 4: The membership functions for the rate of change, ∂CL, of the measured congestion level.

CL		∂CL	
EL	Extremely low	NVH	Negative very high
VVL	Very very low	NH	Negative high
VL	Very low	NM	Negative medium
L	Low	NL	Negative low
M	Medium	Z	Zero
H	High	PL	Positive low
VH	Very high	PM	Positive medium
VVH	Very very high	PH	Positive high
EH	Extremely high	PVH	Positive very high

Table 1: Linguistic variables in Figures 3 and 4.

3 Results

Simulations modelled a dumbbell network topology with a variable capacity tight link that was used to introduce network congestion. The delay over the tight link was set to 5 ms and the FIFO buffer size was set as twice the delay-bandwidth product, as in a normal testing configuration. There was a further 1 ms delay at the side links at either end of the tight link. The tests assume a 25 frame/s video source encoded at 1 Mbit/s and streamed as Constant Bitrate (CBR) from sender to receiver through the dumbbell topology with the sender employing the delay-based FLoCAD algorithm to adapt to the available network. As is normal for MPEG-2 encoded video, each row of macroblocks forms a slice for error resilience purposes and each slice forms a packet, there being 18 slices per frame. Delay data were fed back from the receiver to the fuzzy controller after every frame transmission interval (e.g. 40 ms). The algorithm was simulated with the well-known ns-2 network simulator.

Firstly, the bottleneck bandwidth was fixed at particular rate without any background traffic and several such tests were conducted. Figure 5 show the result of a 1Mbit/s CBR source being adapted to a bottleneck capacity of 500 kbit/s. The result show the algorithm was able to adapt to the link capacity smoothly, which is desirable for rate adaptive video communication to avoid discontinuities at the user's display.

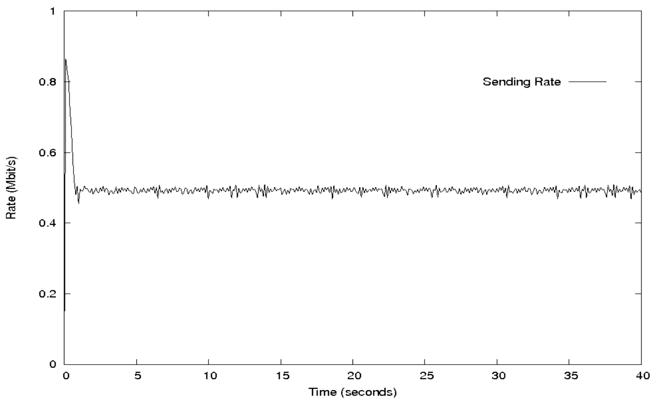

Figure 5: The result of a 1Mbit/s CBR source adapting to a 500 kbit/s bottleneck bandwidth.

Figure 6 shows the average sending rate of the 1 Mbit/s CBR source for different bottleneck bandwidths. The algorithm is shown to accurately track the network capacity. The divergence from the available bandwidth at the start and end of the ranges is due to the inability of the transcoder as implemented to reduce its sending rate below about 0.13 Mbit/s and above 0.85 Mbit/s. Packet loss was zero even for a bottleneck bandwidth of 200 kbit/s with a source rate of 1 Mbit/s.

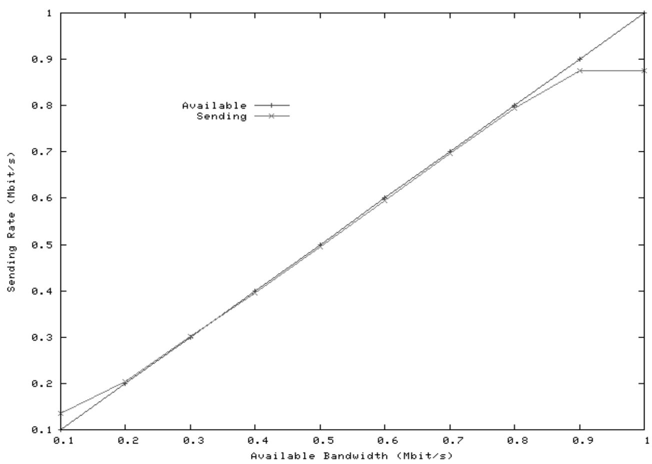

Figure 6: The average sending rate of a 1 Mbit/s source adapting to different bottleneck bandwidths.

Secondly, the performance of the scheme was tested by having the bottleneck available bandwidth change during a streaming session. This is achieved by injecting background traffic to coexist with the streamed traffic for varying available bandwidths. Figure 7 shows the measured network congestion level and the resultant sending rate. The congestion level clearly reflects the available network bandwidth and the result clearly shows the algorithm responding to varying available network bandwidth accurately and in a timely manner.

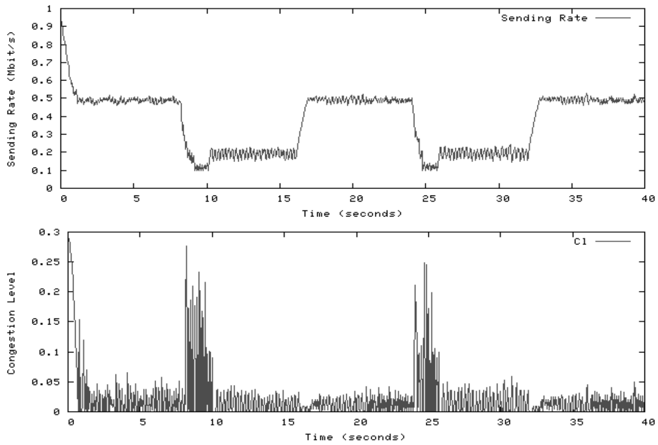

Figure 7: Graph showing the congestion level and sending rate of a 1Mbit/s CBR source adapting to a varying available bandwidth.

4 Conclusion

This paper demonstrated that a fuzzy logic congestion controller using delay information was able to accurately determine imminent congestion and in a timely manner adapt the source sending rate to the available sending rate. The results obtained from these tests confirm the suitability of using delay information in a fuzzy logic controller to achieve very good rate control for networked multimedia applications. However, further work needs to be undertaken to enhance and test FLoCAD to determine its performance in high bandwidth-delay product networks. This work, which is ongoing, involves extensive testing of network parameters. No doubt also the fuzzy models can be refined. However, this is an advantage of the approach compared to model-based methods of congestion control, as the model's governing equation is not easily altered.

Acknowledgements

This work was supported by the EPSRC, UK under grant no. EP/C538692/1.

References

[1] M. Allman, V. Paxon, W. Stevens. "TCP Congestion Control", *RFC 2581, 1999.*

[2] P. A. A. Assunção and M. Ghanbari, "Buffer analysis and control in CBR video transcoding", *IEEE Trans. on Circuits and Systems for Video Technology*, **vol. 10, no. 1**, pp. 83–92, (2000).

[3] S. Biaz and N. H. Vaidya, "Is the Round-trip Time Correlated with the Number of Packets in Flight?", *3rd ACM SIGCOMM Conference on Internet Measurement*, pp. 272–278, (2003).

[4] L. Brakmo, S. W. O'Malley, and L. L. Peterson. "TCP Vegas: New Techniques for Congestion Detection and Avoidance", *Proceedings of SIGCOMM 1994*, **vol. 8**, pp. 1465-1480, (1995).

[5] W. Dabbous, "Analysis of Delay Based Congestion Avoidance Algorithm," *High-Performance Networking IV*, pp. 283–298, (1992).

[6] S. Floyd and V. Jacobsen, "Traffic phase effects in packet-switched gateways," *Computer Communications Review*, **vol. 21, no. 2**, pp. 26–42, (1991).

[7] S. Ghosh, Q. Razouki, H. J. Schumacher, and A. Celmins, "A Survey of Recent Advances in Fuzzy Logic in Telecommunications Networks and New Challenges," *IEEE Trans. on Fuzzy Systems*, **vol. 6, no. 3**, pp. 443–447, (1998).

[8] R. Jain, "A Delay-based Approach for Congestion Avoidance in Interconnected Heterogeneous Computer Networks," *Computer Communications Review*, **vol. 19, no. 5**, pp. 56–71, (1989).

[9] C. Jin, D. X. Wei, and S. H. Low, "The Case for Delay-based Congestion Control," *IEEE Annual Workshop on Computer Communications*, pp. 99–104, (2003).

[10] R. Q. Hu and D. W. Petr, "A Predictive Self-tuning Fuzzy-logic Feedback Rate Controller," *IEEE/ACM Trans. on Networking*, **vol. 8, no. 6**, pp. 697–709, (2000).

[11] S. Liu, T. Başar, and R. Srikant, "TCP Illinois: A Loss and Delay-based Congestion Control Algorithm for High-speed Networks", *1st Int. Conf. on Performance Evaluation Methodologies and Tools*, Article no. 55, (2006).

[12] J. Padyhe, V. Firoiu, D. Towsley, and J. Krusoe, "Modeling TCP Throughput: A Simple Model and its Empirical Validation", *ACM SIGCOMM '98*, pp. 303–314, (1998).

[12] A. Pitsillides and A. Sekercioglu, "Congestion Control", in *Computational Intelligence in Telecommunications Networks*, CRC Press, Boca Raton, FL, pp. 109–158, (2000).

[13] C. E. Rohrs, R. A. Berry, and S. J. O'Halek, "A Control Engineer's Look at ATM Congestion Avoidance," in *IEEE GLOBECOM'95*, pp. 1089–1094, (1995).

[14] L. Rossides, C. Chrysostemou, A. Pitsillides, and A. Sekercioglu, "Overview of Fuzzy-RED in Diff-Serv networks," in *Soft-Ware 2002*, pp. 2–14, LNCS # 2311, (2002).

[15] K. Tan, J. Song, Q. Zhang, M. Sridharan. "A Compound TCP Approach for High Speed and Long Distance Networks", *IEEE INFOCOM,* pp. 1-12, (2006).

[16] X. Wang, D. Ye, and Q. Wu, "Using fuzzy logic controller to implement scalable quality adaptation for stored video in DiffServ networks," in *12th Int. PacketVideo workshop (PV2002)*, (2002).

[17] Z. Wang and J. Crowcroft, "Eliminating periodic packet losses in the 4.3-Tahoe BSD TCP congestion control algorithm," *Computer Communications Review*, **vol. 22, no. 2**, pp. 9–16, (1992).

[18] D. X. Wei, C. Jin, S. H. Low, S. Hedge. "FAST TCP: Motivation, Architecture, Algorithm and Performance", *IEEE/ACM Transactions on Networking*, **vol. 4:(6)**, pp. 1246-1259, (2006).

[19] L. Xu, K. Harfoush, and I. Rhee, "Binary Increase Congestion Control for Fast, Long Distance Networks", *IEEE INFOCOM*, pp. 2514–2524, (2004).