CONGESTION CONTROL ACROSS A VIDEO-DOMINATED INTERNET TIGHT LINK

E. A. Jammeh, M. Fleury and M. Ghanbari
University of Essex, ESE Dept.
Wivenhoe Park, Colchester CO4 3SQ, U.K.

Keywords: Congestion control, video streaming, fuzzy logic.

Abstract: Existing congestion controllers have been designed with TCP traffic in mind. Changing traffic patterns on the Internet imply that on some tight links all UDP video streams will occur. Three different congestion controllers (RAP, TFRC, and fuzzy-logic based), already successful in avoiding instability in current TCP-dominated internets, were tested across a tight link in which video traffic dominated. Congestion control is either achieved by modulating the sending rate in response to feedback of packet loss rates and/or round-trip delays (RAP/TFRC) or a congestion level based on packet dispersion across a network path (fuzzy controller). The controllers were found to differ in the smoothness of resulting video clip streams, with the fuzzy and TFRC controllers, in that order, producing the smoothest received video. Tests also demonstrated that, when controlled flows of different types compete across a tight link, it is possible for the sending rate of TFRC to exceed the available bandwidth, resulting in excess packet loss and implying quite poor video quality at the receiver. The results show that fuzzy-logic control is more flexible when video streams dominate.

1 INTRODUCTION

The Internet's growth has encouraged multimedia applications, such as real-time delivery of sports and news clips, and the exchange of pre-encoded personal video clips (peer-to-peer streaming). If some Internet links are dominated by video clip streams and/or DVD-video streaming, how will congestion controllers cope with this situation? Currently, TCP-based applications dominate the Internet. TCP's fusion of error and congestion control coupled with its fluctuating sending rate make it suitable for data traffic transport *and* unsuitable for unicast multimedia streams. TCP-friendly Rate Control (TFRC), now an RFC (Handley et al., 2003a), is representative of equation-based control, mimicking TCP's Additive Increase Multiplicative Decrease (AIMD) congestion probing algorithm. However, in (Cai et al., 2005), it is remarked that "It is unclear whether the model itself still applies when the mix of TCP and non-TCP traffic changes dramatically over time as new applications keep emerging over the Internet".

The Rate Adaptation Protocol (RAP) (Rejaie et al., 1999) is a well-known alternative to equation-based modeling, varying the inter-packet gap (IPG) between UDP packets at the sender, to allow its average sending rate to approach TCP's for a given available bandwidth. Equally, fuzzy logic-based congestion control has been shown (Jammeh et al., 2005) to closely track capacity over a tight link, especially when there are a limited number of flows, as feedback if over the same link may lead to a non-linear control system. Fuzzy-logic control avoids the need to directly utilize packet loss (Jammeh et al., 2006) as an input to the congestion controller, as packet dispersion on the Internet path forms the feedback. Therefore, when rate-controlled transcoding (Assunção and Ghanbari, 1998) is employed, the video rate can be modulated, resulting in low loss rates, which are clearly beneficial to video quality.

All three methods (TFRC, RAP, fuzzy) may avoid network instability in current traffic scenarios but how might they react in a situation in which all traffic over a particular congested link is controlled by means of the *same* scheme, which we call intra-protocol fairness or all traffic consists of congestion-controlled UDP streams, which we call inter-protocol fairness? This seems to be a question that has not received sufficient attention in the research literature.

A 2003 survey (Fraleigh et al., 2003), reporting on

A. Jammeh E., Fleury M. and Ghanbari M. (2006).
CONGESTION CONTROL ACROSS A VIDEO-DOMINATED INTERNET TIGHT LINK.
In *Proceedings of the International Conference on Signal Processing and Multimedia Applications*, pages 89-94
DOI: 10.5220/0001567700890094

traffic samples taken in 2001-2, indicated that media streaming and distributed file sharing traffic had risen to 60% of the total in the 'Sprint' network core, with only 30% contributed by Web traffic. The report found considerably variety in the traffic mix, depending on link type and link capacity. Peer-to-peer file sharing traffic had a strong impact but up to 26% of any sampled link's traffic was formed by streaming applications. This mix might lead to still higher link concentrations, once passing from the network core to the network edge.

Network congestion impacts the delivery of encoded video due to an inconsistent and constrained available network bandwidth, increase in one-way delay and jitter, with an increased packet loss rate. Over-provisioning is unlikely to remove this problem (Gevros et al., 2001), because of bandwidth mismatches at the Internet's periphery. Therefore, the key problem is reacting to congestion caused by other traffic as the stream passes across such a bottleneck.

It is also worth noting that RealVideo, which may account for over 40% of commercially streamed multimedia content, has been shown with a network testbed (Chun et al., 2002) to be particularly unfriendly to equivalent TCP flows, with 20% of UDP flow acquiring in the long run as much as twice the rate of a TCP flow during network bandwidth constriction (for a constriction of 300 kb/s and below, fed from a 700 kb/s DSL link to the Internet). This remains a problem that congestion controllers restrained by the TCP throughput equation are badly placed to meet.

The remainder of this paper is organized as follows. Section 2 introduces the three congestion controllers that are compared. Section 3 describes the experimental methodology. Section 4's results consider two key problems: 1) delivery of a smoothly varying video stream; and 2) avoiding packet loss from too aggressive bandwidth acquisition takes place. Finally, Section 5 draws some conclusions.

2 CONGESTION CONTROLLERS

2.1 Tfrc Control

In TFRC, (Handley et al., 2003b), the sending rate is made a function of the measured packet loss rate during a single round-trip time (RTT) duration measured at the receiver. The sender then calculates the sending rate according to the TCP throughput equation reported in (Padhye et al., 1998). Therefore, TFRC is restricted to situations in which the TCP throughput equation is applicable. TFRC was designed to produce smooth multimedia flows but, because it assumes constant-sized large (MTU) packet sizes, it introduces a bias against the variable-sized packets of variable bit-rate (VBR) video. TFRC's throughput model is sensitive to the loss probability and RTT, which are difficult to predict or measure. Ironically, TFRC may perform better when there are packet losses, but this is not conducive to the quality of delivered video. Another potential weakness is the response to short-term TCP flows, typically HTTP traffic, which never develop long-term TCP flow behavior.

2.2 Rap Control

RAP (Rejaie et al., 1999) uses acknowledgment packets (ACKs) to detect packet loss and infer RTTs. Every smoothed RTT, RAP implements an AIMD-like algorithm with the same thresholds and increments as TCP. Because this would otherwise result in TCP's 'sawtooth'-like rate curve, with obvious disruption to video streams, RAP introduces fine-grained smoothing (turned-on in Section 4's tests), which takes into account short- and long-term RTT trends (by forming the ratio of the two). As time-outs for single loss detection are not applied, RAP's output is greater than equivalent TCP traffic during heavy congestion. RAP's form of rate control also implies fixed-size packets, again creating problems for VBR video.

2.3 Fuzzy Control

Fuzzy congestion control is a sender-based system for unicast flows across the Internet, principally applied to rate-controlled transcoded video. Transcoded video allows the rate to dynamically adapt to changing network conditions as well as match the capabilities of a variety of receiver devices.

The main constituents of the fuzzy logic congestion controller are a fuzzy inference system, a rule base and a defuzzifier. The fuzzy inference system uses a Mamdani model (Mamdani and Assilian, 1975) with two inputs: 1) congestion level 2) instantaneous rate of congestion level change. The video receiver device returns a feedback message indicating the dispersion of packet inter-arrival times from which the sender then calculates time-smoothed changes to the congestion level. A fuzzy inference system maps the given inputs to an output using fuzzy-logic membership functions to 'fuzzify' the input. The mapping is performed by means of 'if-then' rules and fuzzy operators.

Subsequently, centroid of area defuzzification is applied to arrive at a crisp output level. The sender then modulates the sending rate according to this output level, in practice by applying a control signal to a transcoder's quantization level. The result is a stream of variable-length packets with a fixed IPG. The dispersion of the IPGs is recorded ar the receiver.

This is a flexible method of control, suitable for variable-bit rate (VBR) video, that can control output (Jammeh et al., 2006) in the presence of short- and long-term HTTP flows.

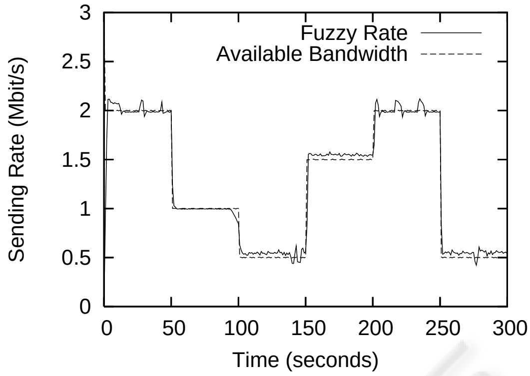
3 TEST METHODOLOGY

To ensure fairness to both RAP and TFRC publicly-available ns-2 models (in the form of object `tcl` scripts to drive the simulator) was availed of respectively from `http://netweb.usc.edu/reza/RAP/NewRAP/` and `http://www.icir.org/tfrc/`, with the fuzzy models available from this paper's first author.

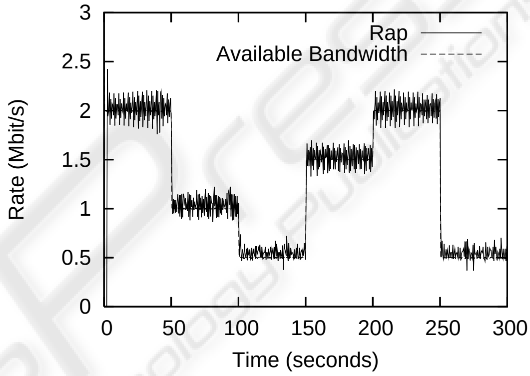
In both RAP and TFRC, the constant packet length was set in the supplied scripts to 1000 B. The maximum packet size in the fuzzy tests was also set to 1000 B but the fuzzy controller alters the packet length to achieve a desired bitrate. The fuzzy system IPG (the time between the end of transmission of one packet and the start of another) was set to 2.2 ms, which is equivalent to an MPEG-2 video stream at 25 frame/s for European SIF-sized pictures with 18 slice/frame and one slice in each packet. However, the source was Constant Bit-Rate(CBR) and not a VBR video emulation or trace, so as to avoid invalidating the comparison with TFRC and RAP. IPG dispersion information is fed back after every frame or frame transmission interval (every 40 ms in the simulation) from the receiver. Again a flexibility of fuzzy-control is that feedback does not need to be so often as in AIMD-based control. For fuzzy-controlled VBR traffic refer to (Jammeh et al., 2006).

The RAP model assumes that rate transitions are quantized (as in layered coding), though the source utilized in the simulation does not emulate or make use of video traces. Unlike, RAP, the fuzzy controller in practice (not simulation) is integrated with a transcoder and, thus, could readily achieve fine-grained adaptation. TFRC's simulation model source is not video-specific but simply provides packets up to the available rate, though a smoothing option was turned on. Decision frequency is every RTT. In all simulations, packets were always available at source.

In order to calibrate the models, the ability to track a varying available bandwidth was tested. Fig. 1 shows that both fuzzy and RAP control are able to follow scheduled changes by the simulator. The transitions made by the controllers are abrupt, though not as abrupt as the limited resolution of the graphs may suggest. The main difference is in the smoothness of the response, as RAP's response was notably less smooth, whereas TFRC's response (not shown) is similar to the fuzzy controller. From (Girod, 1992) and else-



(a) Fuzzy control response



(b) RAP control response

Figure 1: Tracking available bandwidth stepped over time.

where, abrupt changes in received video quality are known to having a disturbing effect on the viewer. The main determinant of RAP's behavior, as decision frequency is similar to TFRC, appears to be the size of (stepped) rate transitions, which coarsens the transition levels available despite 'fine-grain smoothing'. Fig. 1 as with others in this study shows sending rate and not router response. This implies that when the sending rate is above the available bandwidth there will be packet loss, though smoothing in intermediate router buffers may reduce the effect. However, in the case of Fig. 1 a fixed reduction in rate, a safety margin, should result in the fuzzy controlled stream avoiding loss.

In live tests of 4000 video clips (Chun et al., 2002), over 50% of clips were less than 200 s with a median of 180 s, which is similar to the experiments in Section 4. The same experiment found that the median packet length with UDP streaming was 640 B, which is similar to the fixed packet length of RAP and TFRC in the tests. By means of transcoding or layered coding, a VCR quality or even MPEG-2 broadcast video rate can be gracefully degraded to around 1 Mb/s, while in the U.K. peak ADSL bandwidth at household access links remains at 2.9 Mb/s, which explains the choice of rates and bottlenecks in the following Section.

4 RESULTS

In the intra-protocol fairness experiments, a number of CBR streams were simultaneously transmitted over a shared tight link, representing the minimum available bandwidth on an end-to-end path. In the ns-2 simulator, this arrangement was modeled by a 'dumbbell' topology. Five simultaneous flows were presented; other tests for up to ten simultaneous flows revealed similar results. No TCP traffic was present but a set of identical TCP-friendly controllers competed for available bandwidth. The simulator was unable to set the feedback at exactly the same time for all the controllers and, therefore, as for Internet sources, the output was not identical even though the congestion controllers were.

Fig. 2 shows the intra-protocol allocation of bandwidth for the three congestion controllers for a sample 2.0 Mb/s bottleneck over a 120 s period (time for a video clip to be exchanged). The average bandwidth of any one flow for each sampled bandwidth remains close to the required value, as the fuzzy example, Fig. 3, illustrates. Very similar plots occur for RAP and TFRC congestion controllers and at different bottleneck settings. Therefore, clearly all three congestion controllers appear 'friendly' to themselves.

Though all flows, for each type of controller, cluster around the correct level, Fig. 2, given the available bandwidth, RAP is visibly more 'bursty' in its response. The RAP plot shows a large initial sending rate, which was discarded in subsequent analysis to avoid biasing the comparisons by this start-up feature. The issue is less easy to resolve between TFRC and the fuzzy controller and to do so the Coefficient of Variation (CoV) was taken for the aggregated set of five flows. The CoV is calculated for any one bottleneck bandwidth by taking the mean and standard deviation (s.d.) of each flow over the 120 s period and then forming the mean of the means and the mean of the s.d.s. The aggregate s.d.s are then normalized by the aggregate means to form the CoV for each data point.

Fig. 4 plots the CoV response across a range of bottleneck bandwidths in which TFRC exhibits more excursions than fuzzy control. To confirm that fuzzy is less bursty than TFRC, Table 1 tabulates the s.d.s for a sample of two flows (the other flows were similar) taken at random from the five for each of the congestion controllers. In each case, the fuzzy controller s.d.s are consistently less than those of TFRC (and RAP), showing that in these circumstances the fuzzy controller's output is smoother.

Table 1: S.D.s for a sample two flows over 120 s tests.

Congestion controller	Bottleneck b/w (Mb/s)	Flow1 s.d. (Mb/s)	Flow2 s.d (Mb/s)
TFRC	2.0	0.448486	0.63709
RAP	2.0	1.35377	0.97761
Fuzzy	2.0	**0.27979**	**0.28815**
TFRC	5.0	1.483344	1.47669
RAP	5.0	2.80038	2.13319
Fuzzy	5.0	**1.38128**	**1.53012**

Turning to inter-protocol allocation of bandwidth, Fig. 5 shows two flows with different congestion controllers illustrating that both behave reasonably to each other, with similar bandwidth allocations and with dominance oscillating between each controller.

However, this impression is dispelled when multiple flows compete, Fig. 6, when the average bandwidth of the TFRC flows are always above that of the fuzzy-controlled flows and above an ideal or equally-shared bottleneck bandwidth. Clearly, if replicated in a network this behavior would result in increased packet loss for the TFRC flows. We postulate that the cause of this behavior is over-sensitivity on the part of TFRC to variation in packet loss and RTT. One implication is that TFRC may closely model TCP's average throughput but doing so may result in untoward loss of quality for received video.

5 CONCLUSIONS

A number of conformant congestion controllers (ones that are TCP-friendly) have been developed. Tests now show that, three of these at least (RAP, TFRC and fuzzy) are largely equivalent in their response to traffic flows from the same type of congestion controller (either all RAP or TFRC or fuzzy). However, RAP controller's stepped response to feedback does not result in smooth flows. Smoother flows allow graceful degradation of transmitted video in the face of congestion and smaller receiver buffers. The end result may translate into a better subjective experience by the user viewing the received video and reduce the problem of buffering memory on mobile devices. There is also a reduced risk of packet loss at router queues. Because it does not employ AIMD's bandwidth probing and through design of its membership models, fuzzy control is favored over TFRC in that respect. TFRC assumed more than an equal share of available bandwidth in inter-protocol tests (when traffic from other congestion controllers was present), which in practice would lead to greater packet loss. Feedback of packet inter-arrival time dispersion, rather than packet loss or two-way delay, may help the fuzzy controller to keep its sending rate to the available bandwidth, leading to higher quality video at the receiver.

REFERENCES

Assunção, P. and Ghanbari, M. (1998). Bit rate control of Internetworking video transcoders. In *Fifth International Conference on Electronics,Circuits, and Systems, ICECS'98*, volume 2, pages 247–250.

Cai, L., Shen, X., Pan, J., and Mark, J. W. (2005). Performance analysis of TCP-friendly AIMD algorithms for multimedia applications. *IEEE Transactions on Multimedia*, 7(2):339–335.

Chun, J., Zhu, Y., and Claypool, M. (2002). FairPlayer or FoulPlayer? —head to head performance of RealPlayer streaming video over UDP versus TCP. Technical report, Worcester Polytechnic Institution, MA.

Fraleigh, C., Moon, S., Lyles, B., Cotton, C., Khan, M., Moll, D., Rockell, R., Seely, T., and Diot, C. (2003). Packet-level traffic measurements from the Sprint IP backbone. *IEEE Network*, 17(6):6–17.

Gevros, P., Crowcroft, J., and Bhatti, S. (2001). Congestion control mechanisms and the best effort service model. *IEEE Network*, 15(3):16–26.

Girod, B. (1992). Psychovisual aspects of image communication. *Signal Processing*, 28(3):239–251.

Handley, M., Floyd, S., Padyhe, J., and Widmer, J. (2003a). TCP friendly rate control (TFRC): Protocol specification. IETF RFC 3448.

Handley, M., Floyd, S., Padyhe, J., and Widmer, J. (2003b). TCP friendly rate control (TFRC): Protocol specification. RFC 3448.

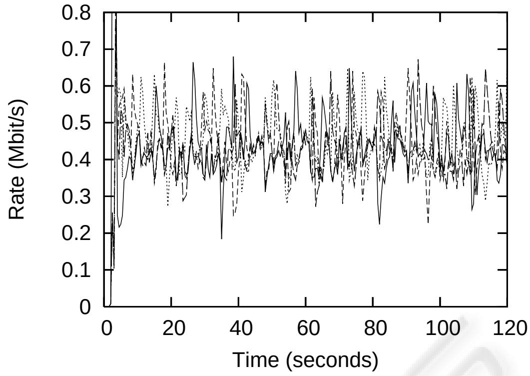
Jammeh, E., Fleury, M., and Ghanbari, M. (2005). A TCP-friendly fuzzy congestion controller for transcoded video over the Internet. In *International Symposium on Telecommunications, Shiraz, Iran, Sept.*

Jammeh, E. A., Fleury, M., and Ghanbari, M. (2006). Non-packet-loss-based rate adaptive video over the Internet. *Electronic Letters*, 42(8):492–494.

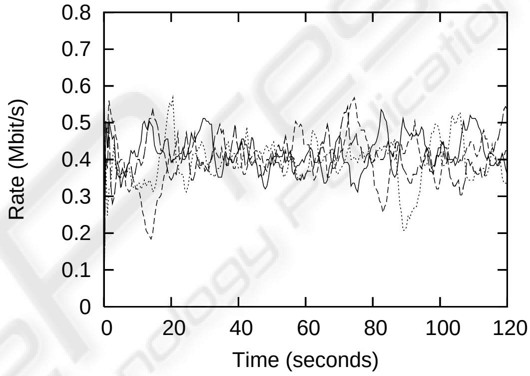
Mamdani, E. H. and Assilian, H. (1975). An experiment in linguistic synthesis with a fuzzy logic control. *International Journal of Man-Machine Studies*, 7(1):1–13.

Padhye, J., Firoiu, V., Towsley, D., and Krusoe, J. (1998). Modeling TCP throughput: A simple model and its empirical validation. In *ACM SIGCOMM '98*, pages 303–314.

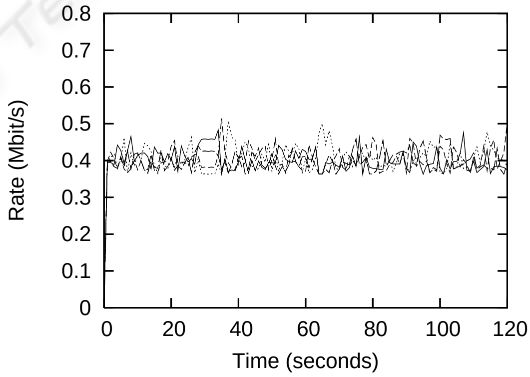
Rejaie, R., Handley, M., and Estrin, D. (1999). RAP: An end-to-end rate-based congestion control mechanism for realtime streams in the Internet. In *IEEE INFOCOM'99*, volume 3, pages 1337–1345.



(a) RAP



(b) TFRC



(c) Fuzzy

Figure 2: Five simultaneous flows each with identical congestion controllers for a 2 Mb/s bottleneck.

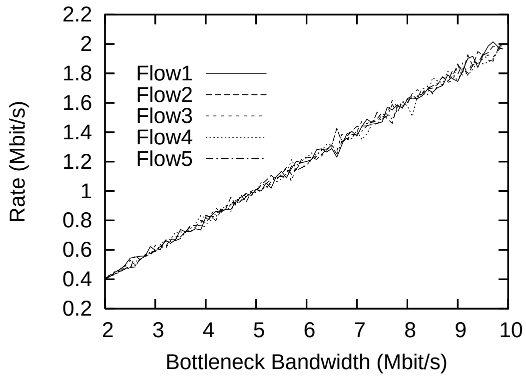


Figure 3: Average bitrate for each of five flows over a range of bottlenecks.

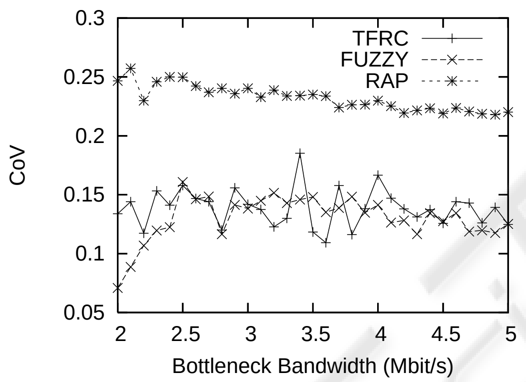


Figure 4: Mean CoVs of five flows for a range of bandwidth bottlenecks.

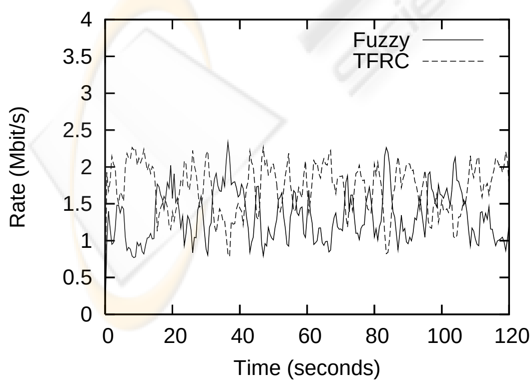
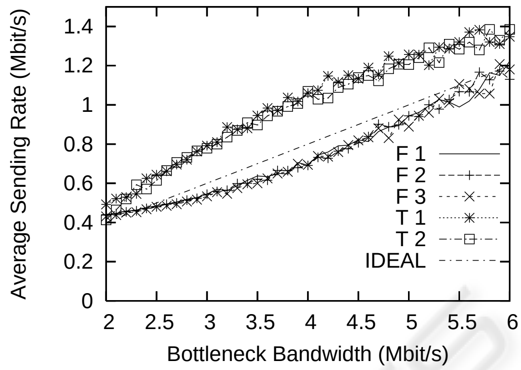
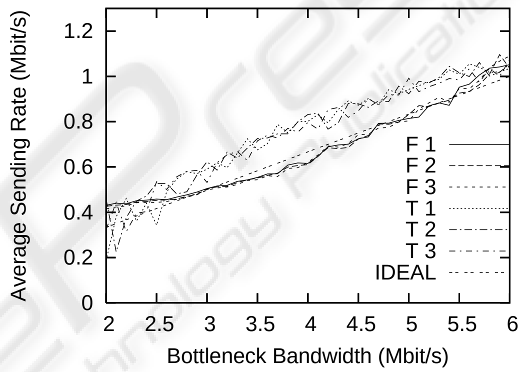


Figure 5: Competing flows with Fuzzy/TFRC over a 3.0 Mb/s bottleneck.



(a) 3 Fuzzy 2 TFRC



(b) 3 Fuzzy 3 TFRC

Figure 6: Competing flows with different controllers across a range of bottlenecks.