IADIS International Journal on Computer Science and Information Systems
Vol. 5, No.2, pp. 98-116
ISSN: 1646-3692

EFFECTIVE MULTI-CONNECTION VIDEO STREAMING OVER WIMAX

Salah Saleh Al-Majeed. *University of Essex, Colchester CO4 3SQ, UK.*
ssaleha@essex.ac.uk

Martin Fleury. *University of Essex, Colchester CO4 3SQ, UK.*
fleum@essex.ac.uk

ABSTRACT

The idea of multi-connection congestion control was originally applied to aggregate flows passing from computer cluster to cluster communicating over the public Internet. This paper considers the extension of multi-connection streaming to wired/wireless networks and in doing so reviews theoretical results for multi-connection streaming, including virtual multi-connections within a single physical connection. Streaming a single video over multiple TCP-Friendly Rate Control connections is a promising way of separately coping with both wireless channel losses and traffic congestion, without the need for cross-layer intervention or retransmission delay at the data-link layer. At the same time, the wireless channel is properly utilized, as throughput improves with an increasing number of connections. Nevertheless, over IEEE 802.16e (mobile WiMAX) tuning is needed to select the number of connections and the Time Division Duplex (TDD) frame size. The paper assesses the impact on video quality of packet drops due both to channel loss over a WiMAX access link and router buffer overflow across an all-IP network, consisting of broadband wireless access and core network. The paper also considers end-to-end delay and start-up delay when employing several connections. Results show that provided the TDD frame size is selected appropriately then using multiple connections preserves video quality and improves wireless channel utilization, with a minimal impact on end-to-end delay. As a trade-off, there is an increase in start-up delay arising from the need to avoid possible buffer underflow.

KEYWORDS

Multi-connections, TFRC, video streaming, WiMAX

1. INTRODUCTION

The demand for IPTV services such as the BBC's iPlayer in the UK suggests that delivery of video streams will shift towards the mobile user from its current emphasis on Asymmetric Digital Subscriber Link access to the home. The iPlayer allows TV programs to be streamed on

demand, either live programs or time-shifted TV. Video delivery is currently based on Adobe Flash Player technology, which has various limitations such as TCP transport (with possible unbounded delays and fluctuating bitrates) and a video chunk unit of delivery, which can lead to breaks in the video stream, causing freeze-frame effects. Elsewhere there are video-on-demand (VoD) companies including MovieFlix and news services such as ABC, BBC, and NBC provide news video clips ready to be streamed. In particular, because of the duration of the films, VoD suffers from the need for lengthy start-up delays to avoid the problem of delivery interruptions resulting to buffer underflow, due to congestion. Such problems are likely to be compounded with broadband wireless access, as, in addition to restrictions upon available bandwidth, there is the risk of adverse channel conditions. One type of solution is offered by the video coding community, which has provided scalable video in the Scalable Video Coding (SVC) extension to the H.264 codec (Schwarz et al, 2007) and error resilience methods in the H.264/AVC (Advanced Video Coding) codec (Wenger, 2003). However, this type of solution typically requires a specialist variety of the decoder (for example, H.264/SVC) situated on the mobile device or anticipation of the possibility of wireless errors when stored video is originally encoded. Channel coding, unless it is adaptive, is a burden during the periods when wireless channel conditions are reasonable and additionally does introduce coding delay, which is a disincentive for the use of interactive video applications.

An interesting development is multiple description coding (MDC) (Wang et al, 2005), as this does not suffer from the weakness of scalable video, data dependencies between the layers. In MDC, the video descriptions are split between two or more connections which can exploit path diversity on wireless as well as wired networks (Apostopolous et al, 2007). MDC certainly does require a specialist decoder, which may not generally be available on the destination mobile device. It also requires dynamic routing that positively selects alternative routes for each description. However, it has made the idea of video streaming over multiple connections more acceptable, provided the solution is not codec-dependent.

TCP's congestion control in single-connection form leads to large fluctuations in the data rate (Tullimas, 2008). Consequently for video streaming, TCP-Friendly Rate Control (TFRC) (Floyd et al, 2000) congestion control has become an industry standard (Handley et al, 2003). TFRC is applied to UDP transport in a way that reduces data-rate fluctuations but maintains the average throughput of TCP, thus not acquiring excessive bandwidth compared to equivalent TCP sources. However, this behavior only applies to the wired Internet and not to wireless access, unless multi-connections are used, as now discussed.

In the multi-connection variant of TCP-friendly Rate Control (TFRC) (Chen and Zakhor, 2006; Handley et al, 2003) video streaming, a *single* video source is multiplexed onto several connections across the wireless link in order to increase the throughput, thereby improving wireless channel utilization. By multiplexing a video stream across multiple connections it is hoped that the impact of packet loss on one or more of these connections will be mitigated by the aggregate datarate across the remaining connections. TFRC's main role (Handley et al, 2003) when congestion occurs across the network path is to reduce the video streaming data rate across the wired portion of the concatenated network. It does this in response to packet drops at intermediate routers, which signal the presence of contending traffic. Unfortunately, TFRC can misinterpret as congestion packet losses due to wireless interference and noise, leading to a reduction in wireless channel utilization. Though cross-layer approaches to avoid misinterpretation are possible, these are complex to implement and inflexible. In fact, cross-layer approaches are most appropriate when a network has a fixed application, not one in which multimedia streaming is mixed in with other types of traffic.

In pioneering work on multi-connection TFRC, that is in MULTTFRC (Chen and Zakhor, 2006), improved video quality comes about by increasing the quantity of video data that can be sent over the multiple connections. Of course, increased video data implies a lower compression ratio and, hence, higher-quality delivered video, provided the rise in packet losses across the wireless channel does not degrade the quality. If burst errors occur then during the time that they occur all connections are affected, leading to a rise in packet losses, which was countered in Chen and Zakhor (2006) by means of application Forward Error Control (FEC). Unfortunately, if the number of connections varies, as it does in Chen and Zakhor (2006), then sending rate oscillations can occur. If the compression rate is varied at the source (either by changing the quantization parameter at the codec if live video or through a bit-rate transcoder if stored video) then oscillations in bitrate run the risk of disconcerting changes in displayed video quality.

However, we show that delivered video quality is maintained without the need to dynamically change the compression ratio by keeping the number of connections constant. This is because, with multiple TFRC connections, TFRC is better able to control its sending rate. In fact, TFRC (Handley et al, 2003) was originally designed with a high number of streams in mind, as may arise from a Video-on-Demand server, and special measures are recommended if the number of contending flows is *not* large enough. We consider an IEEE 802.16e (mobile WiMAX) (IEEE, 2005) uplink, which is the access network stage of an all-IP network (Lin and Pang, 2005). There is interest in uplink media services as a complement to IPTV video broadcasting. In this environment, the paper assesses the impact on video quality of packet drops due both to channel loss and router buffer overflow. It should be remarked that in Chen and Zakhor (2006) there was no investigation of actual video quality beyond the packet loss statistics.

The paper also considers end-to-end delay and start-up delay when employing several connections. Results show that provided the TDD frame size is selected appropriately then using multiple connections preserves video quality, as a result of the differential effect of packet loss patterns. Wireless channel utilization is considerably improved, with a minimal impact on end-to-end delay. However, for a WiMAX uplink this only becomes apparent if the Time Division Duplex (TDD) frame length is tuned to avoid queue servicing scheduling delays. The frame length is significant as a longer frame reduces delay at a WiMAX subscriber station, thus permitting more data to be removed from queues when the subscriber station's queues are polled. As a trade-off, there is an increase in start-up delay arising from the need to avoid possible buffer underflow. However, this delay is generally smaller than that arising from TCP-based streaming, when large buffers are normally employed (Hsiao, Kung, and Tan, 2003) to counter the possibility of repeated retransmissions. For example in Shen et al (2009) a buffer size of 10 s was required to absorb the effects of wireless burst errors in TCP-based streaming.

IEEE 802.16e provides broadband wireless access independent of a pre-existing cellular system, is not dependent on hardware authentication, can deliver data in a cost-effective way at 3–4 times the rate of 3G cellular systems, and is currently deployed, rather than in development. Its main technological weakness may be that it uses Orthogonal Frequency Division Multiple Access (OFDMA) for both the uplink and downlink transmission, rather than OFDMA for the downlink and Single Carrier-Frequency Division Multiple Access (SC-FDMA), which confers power saving advantages upon Long Term Evolution (LTE) (Ekstrom, 2006). WiMAX is suited to provide dedicated multimedia services, with existing services in Brazil, Mexico, and Korea (as WiBro is now harmonized with WiMAX) (Chen and de Marca,

2008), and systems throughout the world are being deployed to rural areas and in areas without a good pre-existing 3G infrastructure.

It was suggested in Tappayuthpijam et al (2009) that in LTE packet loss can be virtually eradicated by retransmission at the data-link layer. This then allowed TFRC to be used over a wireless link without the worry of erroneous response to packet loss. However, that approach has the potential to introduce unbounded delay across the wireless link, apart from the drop in throughput that results. In fact, the approach reintroduces the problems that led to the search for an alternative to TCP transport for multimedia streaming. There is also the overhead of maintaining state at the evolved node B (an LTE radio head) and the delay arising if retransmissions are still taking place when a handoff occurs. Therefore, we consider that further investigation of multiple TFRC connections is a way forward in these networks, especially if there is a wired network present beyond the wireless access link.

This paper now considers the theory behind multi-connection streaming. The following Section, describes the WiMAX scenario in which effective ways of utilizing multi-connection streaming are explored. Section 4 presents analysis and results from that scenario. Section 5 reviews alternative ways of providing end-to-end congestion control in a heterogeneous network. Finally, Section 6 draws the paper to an end by making some concluding remarks.

2. MULTI-CONNECTION THEORY

This Section commences with an overview of the research literature on multi-connection video streaming. Before multi-connection TFRC, aggregate flow management was proposed by Ott et al (2004), whether by TFRC or TCP, to improve network utilization. Aggregate flow management differs from multi-connection streaming only in that instead of one video streaming source there are multiple sources. In cluster-to-cluster applications such as distributed sensor networks, the aggregate cluster traffic shares an Internet path with other data flows. Multi-connections were subsequently exploited without the presence of clustersto improve wireless channel utilization for TFRC (Chen and Zakhor, 2006) in tandem networks containing a wired and wireless path component. In fact, the same authors (Chen and Zakhor, 2006a) suggested that multi-connection TCP could also improve wireless channel utilization in a tandem network. The idea of multi-connection TCP was explored in the wired Internet environment (Tallimas et al, 2008) to improve throughput while congestion control was operating. That work (Tallimas et al, 2008) build upon the research of Crowcroft and Oeschlin (1998) (mulTCP) in which the TCP receiver window size was varied to provide weighted fair sharing amongst coincident Web flows. In Tallimas et al (2008), because the data-rate is adjusted by changing the TCP receiver window, there is no need in a congested network to alter the number of connections to accommodate changing levels of congestion. The Stream Control Transmission Protocol (SCTP) (Stewart, 2007) also supports multi-connection streaming with optional out-of-order delivery to avoid TCP's potential head-of-line blockages. Though SCTP mitigates other TCP shortcomings, such as lack of message structuring and exposure to SYN flooding, it still essentially provides a TCP-like reliable service, potentially exposing each connection to long delays while packets are retransmitted. Wetzl and Stadler (2005) showed in user tests that both TCP's and other aggressive congestion control mechanisms not resulting in smooth throughput are not appreciated by users because of the variations in video quality at the receiver display.

In Damjanovic and Wetzl (2009), the TFRC connections were virtual connections, because a single physical TFRC connection was enabled to acquire sufficient bandwidth, *as if it was* a

multi-TFRC connection flow. In Damjanovic and Wetzl (2009) the TFRC equation (Floyd et al, 2000) (see Section 3.2) is altered to give the rate of *n* parallel TFRC flows. This approach is an echo of All-in-One TFRC (AOI-TFRC) (Chen and Zakhor, 2005) which also merged the rate of multiple TFRC connections into a single connection. There are two main advantages of connection merging: 1) a reduction in connection management and 2) the ability to include a fractional number of connections, allowing a closer approach to optimal channel utilization. The latter is a particular problem for multi-connection systems that dynamically change the number of connections, as occurs in MULTTFRC. This is because it is possible for the connection number decision algorithm to cause the number of connections to oscillate around the ideal fractional number of connections, either selecting one more or one less than this fractional number. If the connections are merged into a single flow then a fractional aggregate rate can be selected. However, a further issue arises because TFRC judges its sending rate by both packet-loss rate and round-trip time (RTT) (refer to Section 3.2 for a description of TFRC). Whereas it is possible to average packet RTTs, a lower packet loss rate for the increased sending rate of the aggregate flow leads TFRC to increase its sending rate. In particular, the problem arises because the sending rate is computed by dividing by the square-root of the packet loss rate rather than the loss rate itself. This can lead to TCP unfriendliness when the loss rate on the wireless network is low, leading to potential congestion collapse within the wired Internet. Therefore, in AIO-TFRC, the Bandwidth Filter Loss Detection (BFLD) technique was borrowed from Ott et al. (2004) to address this problem. In BFLD, a virtual single-rate TFRC flow is created by selectively marking packets within AIO-TFRC. This allows the packet loss rate to be found at the receiver based on the loss rate of the marked packets rather than a loss rate for all the packets in the aggregate flow. This by no means exhausts methods of finding the appropriate loss rate for an aggregate flow. For example, in PA-mulTCP (Kuo and Fu, 2008), a separate probe is injected into the network to judge the loss rate.

One further, apparently undocumented, disadvantage of aggregating multiple TFRC connections into a single TFRC flow is that it is no longer possible to relieve congestion on the wired portion of the network by dynamically rerouting connections. Unfortunately, simulated investigations (e.g. Chen and Zakhor, 2006) use a dumbbell network topology with no other wired network path available. As a consequence, the possibility of one or more connections taking different routes on the network is not tested. However, the principle problem that connection aggregation addresses is the risk of oscillation around a fractional rate. That problem no longer exists when a fixed number of connections is selected, allowing each TFRC rate to adjust to its local loss rate and round-trip time. There is a deficit in loss of optimality by selecting a fixed number of connections but this can be compensated by individual TFRC connections adjusting their rate, as discussed in this paper. Though it is possible that excessive resources will be consumed handling multiple connections, as a WiMAX subscriber station already manages multiple quality-of-service queues (Andrews et al, 2007), for this technology a lack of processing power at the terminal may not be an issue. For other mobile devices and wireless technologies, the need to keep state for each of the flows should be considered.

One difficulty with MULTFRC's dynamic connection management scheme is its rate of response to packet loss, which in original MULTTFRC consisted of an additive decrease in the number of connections. However, in conditions of a low packet loss rate due to wireless channel errors, MULTTFRC's sending rate can be too high, causing TCP unfairness to other traffic (principally TCP flows). Therefore, in response to an increase in the round-trip-time, in

E-MULTFRC (Chen and Zakhor, 2006a) and EAOI-TFRC (Chen and Zakhor, 2006b) the number of connections or virtual connections respectively is multiplicatively decreased.

Assuming the average RTT, rtt_avg, has been found over some suitable interval, then original MULTTFRC adjusts the number of connections as follows:

$$n = \begin{cases} n - \beta, & \text{if rtt\_avg} - \text{rtt\_min} > \gamma . \text{rtt\_min} \\ n + \frac{\alpha}{n}, & \text{otherwise} \end{cases} \quad (1)$$

where rtt_min is the minimum queuing delay experienced so far, whereas γ. rtt_min is a threshold that triggers a decrease in the number of connections. In MULTTFRC, the calculated number of connections is rounded to the nearest integer, whereas in AIO-TFRC, the calculated number is not rounded. If β = 1 then the number of connections incrementally decreases and automatically increases otherwise (α=1 in simulated implementation (Chen and Zakhor, 2006)). From equation (1), another problem becomes apparent: other than in a simulated environment it is not possible to know global rtt_min in advance, because there will always be cross-traffic causing queuing delay. If it is argued that the local rtt_min is appropriate at any one time, then that local rtt_min once used as a global rtt_min may become inappropriate at some later point in time.

For completeness, (2) is the adjusted version of (1) for E-MULTTFRC (Chen and Zakhor, 2006) i.e. EAOI-TFRC but without aggregation:

$$n = \begin{cases} \beta n + \alpha/n, & \text{if rtt\_avg} - \text{rtt\_min} > \gamma . \text{rtt\_min} \\ n + \frac{\alpha}{n}, & \text{otherwise} \end{cases} \quad (2)$$

when α=1-β, with α, β < 1. Though (2) fixes the problem of too slow a reduction in connection numbers. It does not seem to address the threshold setting problem.

Though the investigations in Section 3 are empirical, it is possible to change the TCP window size to achieve a desired throughput (Tullimas et al, 2008). During conditions of no congestion, the TCP throughput is (Tullimas et al., 2008):

$$T = W.M/R \quad (3)$$

where *W* is the maximum window size, *M* is the maximum transport unit, and *R* is the round-trip time. The throughput reduction or difference as a result of *n* packet losses with *N* connections is

$$D_N = \frac{nW^2M}{N^2} \quad (4)$$

Thus, the reduction is less for an increasing number of connections *N*.

From (3) and (4), the relationship between desired throughput, T_D, and window size is given by (Tullimas et al, 2008):

$$W = T_D.R/M \quad (5)$$

Equation (5) allows the initial window size to be set based on a desired throughput and a measured RTT. Then if the measured throughput, T_M, is less than the desired throughput, the window sizes of the connections in order of increasing window size are each incremented until the aggregate difference in throughput given by (6) is less than or equal to zero. A similar procedure is followed if the aggregate difference exceeds the desired throughput except that the window sizes are decremented in decreasing order of current size.

$$D_S = |T_D - T_M|R/M \quad (6)$$

It is also necessary to constrain the rate of increase and decrease in throughput by setting limits on the sum of the connections' window sizes.

Future work, outside the scope of this paper, is to adapt TFRC to an internal method of adjusting its rate in a similar manner to window-resizing regimes. TFRC already adjusts the inter-packet gap but in some circumstances the gap size can result in prohibitive delays. Therefore, a need arises to limit the inter-packet gap size, which implies that packet size should also be altered. However, this is not a simple matter when dealing with compressed video data because of the risk of separating internal header information in one packet from the compressed data in another. If the packet with header information is lost then the data in another surviving packet could not be reconstructed.

3. SCENARIO INVESTIGATED

The scenario tested in this paper is shown in Figure 1. The following describes the WiMAX part and this description is followed by a description of the inset, showing traffic sources and sinks within the core IP network.

3.1 WiMAX System

In Figure 1, once a Base Station (BS) has allocated bandwidth to each subscriber station (SS), each SS must manage its queue according to the data arrival rate from user applications. WiMAX networks support multiple service classes to accommodate heterogeneous traffic with varying requirements. WiMAX's rtPS is most suitable for real-time video services, particularly for Variable Bitrate Video (VBR), which is employed to maintain delivered video quality but may lead to 'bursty' arrival rates. Other congesting traffic is assumed to enter the non-real-time Polling Service (nrtPS) queue at the SS. In our experiments for both queues, a drop-tail queuing discipline was simulated. Queue sizes in tests were all set to fifty packets. This value was selected as it seems appropriate to mobile, real-time applications for which larger buffer sizes might lead both to increased delay and also greater active and passive energy consumption at the buffer's memory. Access to the SS service class queues was round-robin.

The physical layer (PHY) settings selected for WiMAX simulation are given in Table 1. The antenna is modeled for comparison purposes as a half-wavelength dipole. The antenna heights are typical ones taken from the Standard (IEEE, 2005). The Gilbert-Elliott 'bursty'

channel model is further explained in Section 2.5. The TDD frame length was varied in experiments, because, as mentioned in Section 1, it has an important effect on the service rate at an SS. Current implementations have apparently mostly opted for a fixed 5 ms TDD frame size, which corresponds to the WiMAX Forum recommendation. The uplink (UL)/downlink (DL) is adaptable at the BS and is set to favor the UL for the purposes of our tests. The parameter settings in Table 1 such as the modulation type and physical-layer coding rate are required to achieve a datarate of 10.67 Mbps. However, in the Standard (IEEE, 2005) these settings are mandatory for the downlink, while here they have been adopted for the uplink. The corresponding downlink rate is 2.69 Mbps.

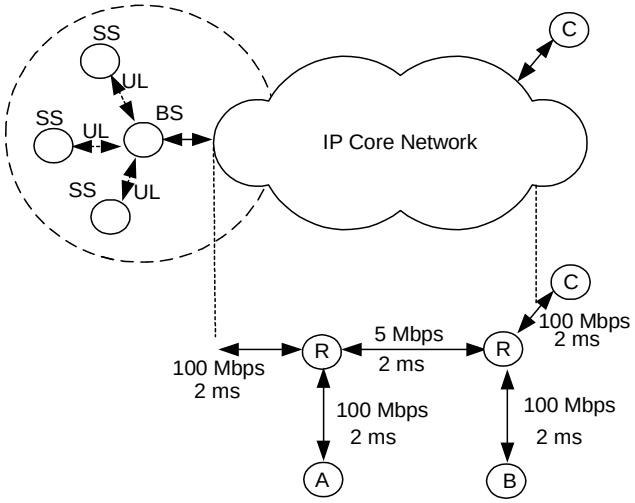

Figure 1. Network scenario with inset showing routing across the core network, A, B and C being sources and sinks, and R = router

3.2 WiMAX Traffic Characteristics

There were three SSs communicating to the BS, with one of the SS sending a VBR video sequence encoded with the H.264/Advanced Video Codec (AVC) (Wiegand et al, 2003) and split between the multiple TFRC connections. The other SSs are introduced as sources of contending traffic across the wireless link and do not indicate the likely size of a WiMAX network, which obviously could be larger. A trace file was input to the well-known network simulator ns-2 and packet losses recorded in the output. The output serves to calculate the PSNR. Video quality comparisons were made under the EvalVid (version 2) environment (Klaue et al, 2003). Data points are an average of fifteen runs. The output serves to calculate the Peak-Signal-to-Noise Ratio (PSNR). As a test, we used the 'Paris' clip H.264 Variable Bit Rate (VBR)-encoded at 30 Hz (frame/s) at Common Intermediate Format (CIF) (352×288 pixel/frame) with initial quantization parameter set to 26 (from a range 0 to 51). 'Paris' consists of two figures seated around a table in a TV studio setting, with high spatial coding complexity. The intra-refresh rate was every 15 frames with IPBB…I structure, i.e. the GOP size was 15. 1063 frames were transmitted. Previous Frame Replacement (PFR) was set for error concealment at the decoder for comparison with coding results, which assume PFR. The

slice size was fixed at the codec as 900 B. In selecting codec determination of slice size, packet segmentation is avoided, which improves video quality, as slices are not separated from their resynchronization headers.

Table 1. Simulated WiMAX settings

Parameter	*Value*
PHY	OFDMA - 1024
Frequency band	5 GHz
Duplexing mode	TDD
Frame length	5 to 20 ms
Max. packet length	1024 B
Raw data rate	10.67 Mbps
Modulation	16-QAM 1/2
Guard band ratio	1/16
DL/UL ratio	1:3
DL length	1.25 to 5 ms
UL length	3.75 to 15 ms
Channel model	Gilbert-Elliott
MS transmit power	245 mW
BS transmit power	20 W
Approx. range to SS	0.7 km
Antenna type	Omni-directional
Antenna gains	0 dBD
MS antenna height	1.5 m
BS antenna height	32 m
Receiving threshold	7.91e-15 W

OFDMA = Orthogonal Frequency Division Multiple Access, QAM = Quadrature Amplitude Modulation, TDD = Time Division Duplex

Table 2 records the simulated traffic characteristics for the three SSs communication with the BS. Network Adaptation Layer units (NALUs) from the H.264 codec were encapsulated with Real Time Protocol (RTP) headers. After the addition of IP headers, these in turn formed a single WiMAX MAC Protocol Data Unit (MPDU), which are variable-sized WiMAX packets. (This assumes that just one MAC Service Data Unit (MSDU) is assigned to each MPDU.) For simplicity, a WiMAX MPDU is now referred to as a packet. Coexisting rtPS queue CBR sources were all sent at 1500 kbps, i.e. at a similar rate to the video source. The inter-packet gap was 0.03 s for the CBR traffic. The FTP applications, which continuously supplied data according to available bandwidth, were set up out of convenience as a way of occupying the nrtPS queues; otherwise a Best-Effort (BE) queue might be more appropriate. Likewise, the DL traffic is selected to fully occupy the DL link capacity.

3.3 TFRC Traffic

For TFRC, the inter-packet sending time gap was varied according to the TFRC equation (Handley et al, 2003), not the simplified version reported in Chen and Zakhor (2006). As described in Handley et al (2003), TFRC is a receiver-based system in which the packet loss

rate is found at the receiver and fed-back to the sender in acknowledgment messages. The sender calculates the round-trip time from the acknowledgment messages and updates the packet sending rate. A throughput equation models the mean TCP New Reno to find the sending rate:

$$TFRC\ (t_{rtt}, t_{rto}, s, p) = \frac{s}{t_{rtt}\sqrt{\frac{2bp}{3}} + t_{rto}\ \min\left(1,3\sqrt{\frac{3bp}{8}}\right) p(1+32\ p^2)} \quad (7)$$

where t_{rtt} is the round-trip time, t_{rto} is TCP's retransmission timeout, s is the segment size (TCP's unit of output) (herein set to the packet size), p is the normalized packet loss rate, w_m is the maximum window size, and b is the number of packets acknowledged by each ACK. b is normally set to one and $t_{rto} = 4t_{rtt}$. It is important to notice that t_{rto} comes to dominate TFRC's behavior in high packet loss regimes, which is why it is unwise to use a simplified form of (7). General inspection of (7) indicates that if the round-trip time and/or the packet loss rate increase then the throughput reduces as terms containing these parameters exist in the denominator. As mentioned previously, the intention of (7) is to approximate the mean throughput of TCP (as derived in (Padyhe, J. et al, 1998)) so that TCP sources see a TFRC source as another TCP source and respond in a collective manner in respect to global congestion control (Floyd and Fall, 1999). There is a considerable literature on such TCP-friendly congestion controllers for media streaming (Widmer et al, 2001), which the interested reader is referred to. Though TFRC was designed to have the mean throughput of TCP, it is possible that in some situations this will not occur, as discussed in Rhee and Xu (2005).

In our variant to standard TFRC, the packet size, s, in the TFRC equation (7) was dynamically altered according to EvalVid-created trace file sizes. This variant makes for more responsive control rather than the mean packet length employed in the reference TFRC formulation (Handley et al, 2003). TFRC was originally intended for video-on-demand applications, when it is feasible to calculate the mean packet length from the stored video. Setting a mean packet length is inappropriate for interactive multimedia applications. The underlying TFRC transport protocol was set to UDP, as is normal for real-time applications. Though (7) appears to represent a considerably computational task that could impede real-time performance, it is possible to extract a term parameterized by p, the packet loss rate. Therefore, a look-up table indexed by p represents a practical way to speed up calculations.

Table 2. Simulated WiMAX traffic characteristics

SS-UL	*Service type*	*Traffic type*	*Protocol*	*Packet Size (B)*
1	rtPS	VBR (video)	Multiple TFRC	Variable
		CBR	UDP	1000
	nrtPS	FTP	TCP	
2	rtPS	CBR	UDP	1000
	nrtPS	FTP	TCP	
3	rtPS	CBR	UDP	1000
	nrtPS	FTP	TCP	
SS-DL				
1,2	rtPS	CBR	UDP	1000
3	nrtPS	FTP	TCP	

3.4 Channel Model

A Gilbert-Elliott two-state, discrete-time, ergodic Markov chain (Haßlinger and Hohlfeld, 2008) modeled the wireless-channel error characteristics at the ns-2 physical layer. The result of applying this model is that burst errors typical of known wireless channel conditions appear. The probability of remaining in the good state was set to 0.95 and of remaining in the bad state was 0.94, with both states modeled by a Uniform distribution. The packet loss probability in the good state was fixed at 0.01 and the bad state default was 0.05. The Gilbert-Elliott scheme though simple has been widely adopted, as it is thought to realistically model the burst errors that do occur and, more significantly, can be particularly damaging to compressed video streams (Liang et al, 2008), because of the predictive nature of source coding. Therefore, the impact of 'bursty' errors should be assessed (Liang et al, 2008) in video-streaming applications.

3.5 Core Network Traffic Characteristics

In Figure 1, all links except a bottleneck link within the core network are set to 100 Mbps to easily accommodate the traffic flows entering and leaving the network. The link delays are minimal (2 ms) to avoid confusing propagation delay with re-ordering delay in the results. A bottleneck link with capacity set to 5 Mbps is set up between the two routers. The buffer size in each router was set to 50 packets. This arrangement is not meant to physically correspond to a network layout but to represent the type of bottleneck that commonly lies at the network edge.

Node A sources to node B a CBR stream at 1.5 Mbps with packet size 1 kB and sinks a continuous TCP FTP flow sourced at node B. Node B also sources an FTP flow to the BS and a CBR stream at 1.5 Mbps with packet size 1 kB (see Table 2 downlink). Other SS sources apart from the video connections do not pass over the core network shown but are assumed to be routed elsewhere after passing the WiMAX BS. Node C in Figure 1 is the sink for the TFRC multiple connections.

3.6 Management of Connections

To systematically test the effect of multiple TFRC connections the number of TFRC connections was incrementally stepped up in successive experiments. In MULTTFRC (Chen and Zakhor, 2006), the number of connections is changed over time according to the average round-trip time of all the connections, but this hides the interpretability of results. It is also unclear from Chen and Zakhor (2006) how a single video stream would be apportioned between a varying number of connections. In our experiments, a single queue was segmented into GOPs (one GoP = 15 frames). Each connection was statically allocated its GOPs, which are taken in interleaved manner from the video sequence. This assumes that a re-ordering buffer is available at the receiver, the size of which is discussed in Section 4.

4. EVALUATION

Initial investigations considered the WiMAX link alone in Figure 1. Table 3 shows the average data-rate when transmitting the Paris clip over one or more connections, for two different WiMAX frame lengths: 5 ms and 20 ms. Allowable frame lengths are specified in the Standard (IEEE, 2005), ranging from 2.5 to 20 ms. Clearly, TFRC is able to multiplex more data onto a link as the number of connections increases, though observation of a time-wise plot of throughput shows that during transmission TFRC sharply reduces its overall sending rate in response to packet loss. Because the sending period for one connection with the shorter frame duration is more than the display period of the 'Paris' clip, the longer frame length is preferable if only one connection were to be used. However, with more than one connection, throughput and, hence, wireless channel utilization by the congestion-controlled video streams increases significantly. There is a marked difference if the larger frame length is used whether one or four connections. As smaller frame lengths than 20 ms are generally used for WiMAX, this is an important observation. In fact, the UL proportion of the frame length, that is 15 ms, is more than the total 5 ms frame length that appears to be usually implemented.

Table 3. Sending periods and throughputs when streaming from a mobile SS to the WiMAX BS

No. of connections	*SS to BS (s) (frame length 5 ms)*	*Throughput (kbps)*	*SS to BS (s) (frame length 20 ms)*	*Throughput (kbps)*
1-conn	71.4	217	33.5	467
2-conn	35.8	437	20.5	754
3-conn.	23.3	663	17.7	874
4-conn.	17.4	889	14.6	1059

Table 4. Streaming periods, throughputs and mean packet end-to-end delays from mobile SS to node C in Figure 1 (frame length 20 ms)

No. of connections	*Sending Period (s)*	*Throughput (kbps)*	*Mean end-to-end delay (s)*
1-conn	35.2	444	0.035
2-conn	22.4	690	0.036
3-conn.	21.6	716	0.039
4-conn.	15.6	991	0.062

In respect to the longer frame length of 20 ms, an interesting comparison is with the throughput when the core network is included. In Table 4, there is a similar pattern to the throughputs in Table 3 but the rates are reduced to when streaming only over the WiMAX link. We interpret this effect as not being due to TFRC's response to packet loss but being due to its response to the increased round trip time caused by queuing delay in the buffer prior to the bottleneck link in Figure 1. Notice that TFRC uses reliable TCP to return ACKs, which will tend to add to the round-trip time. Recall also that from equation (1) that round-trip-time is one of the parameters determining TFRC's sending rate. This interpretation is confirmed by the increase in per slice/packet end-to-end delay as more connections are added. In effect, the packets from other connections intervene in the router buffers causing an increase in latency. However, even though the delay is larger for four connections the mean is still less than 100 ms for this scenario.

More significant than end-to-end delay for reconstruction of the video stream is GOP arrival ordering, as this ordering has the potential to introduce interruptions to the display. GOP arrival ordering for four connections is shown in Figure 2. Firstly, a few points about this Figure are explained. Notice that the first H.264/AVC GOP contains parameters that are fixed throughout the sequence (Wiegand et al, 2003). Therefore, this GOP is transported more quickly. Secondly, to avoid a sudden injection of traffic into the network, connection starting times were offset by 0.5 s. In respect to the general findings, a noticeable feature of this Figure is the lengthier start-up periods in sending initial GOPs on each of the connections. We attribute this to the loss of packets at an early stage, which causes TFRC to sharply reduce its rate in a similar manner to TCP's slow-start mechanism. This does mean that about 6 s of frames (amounting to 90 frames) should be stored in the reordering buffer, to avoid the possibility of subsequent underflow in the decoder's playout buffer. As the destination is on the fixed network the reorder buffer is not expected to be a drain on energy resources, as it might be on an SS. Of course, data is not physically reordered in the buffer but accessed through pointers. 6 s is longer than a typical start-up time of around 2 s but is not too large to be objectionable to the user.

Returning to the effect of frame length, video quality (PSNR) and mean packet end-to-end delay were found for a range of WiMAX frame lengths. However, the standard deviation (stdv) over the runs is relatively large (but similar to those reported in Chen and Zachor (2006)). This is explained by the strong effect resulting from the position of error bursts. From Table 5, video quality is generally 'good', as there is an approximate equivalence of PSNR's over 31 dB and above to the ITU's subjective scale. Again the larger TDD frame size results in better and surprisingly in this instance improves in the mean with an increasing number of connections. However, we take this to signify that using four connections produces equivalent video quality at the destination to using one connection, provided the larger frame size is employed. A 5 ms frame size consistently reduces the quality by one or two dB, which on a logarithmic scale is significant. Examination of the total packet losses (congestion and channel loss), Figure 3, shows that losses also are generally higher for a 5 ms TDD frame length than a 20 ms frame length. However, between the connections, it is *not* the case that mean PSNR is a direct reflection of mean packet loss. As might be expected, employing four connections leads to an increase in congestion loss and also channel loss (because error bursts affect more than one connection). Examining the relative breakdown between frame types, shows that anchor frames (I-frames) and reference frames (I- and P-frames) are evenly affected whatever the number of connections. Therefore, we conclude that the differences in the mean PSNRs are explained by the relatively low number of packet losses when using congestion control and possibly the volatility in the pattern of packet losses when burst errors occur.

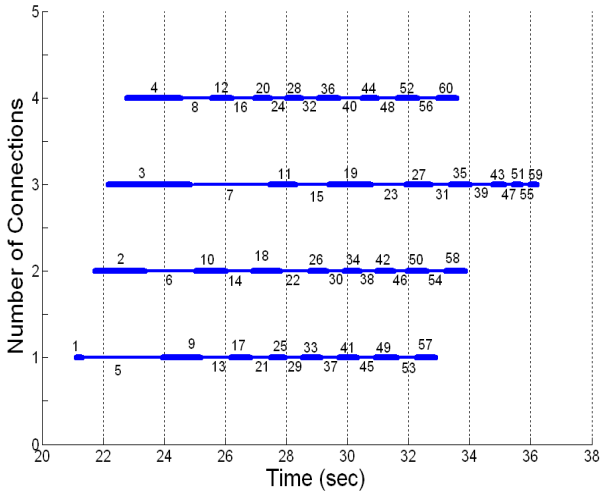

Figure 2. Example GoP arrival sequence at the receiver (node C in Figure 1) showing the start and end times of each GoP

Table 5. Mean PSNR for a range of frame lengths when streaming from a mobile SS to node C in Figure 1

Frame length:	*5ms*		*8ms*		*10ms*		*12.5ms*		*20ms*	
	PSNR (dB)									
Connections	*Mean*	*stdv*	*Mean*	*stdv*	*Mean*	*stdv*	*Mean*	*stdv*	*Mean*	*stdv*
1-conn	29.95	2.90	32.85	3.32	32.44	3.45	33.22	3.45	31.84	3.78
2-conn	28.88	3.07	31.28	3.81	32.17	3.38	33.07	3.47	32.34	3.49
3-conn	29.54	3.25	31.07	3.04	31.83	3.14	30.79	3.25	33.15	3.68
4-conn	28.12	3.11	31.92	3.51	31.20	3.45	33.31	3.80	33.34	3.74

From Figure 3, packet loss is particularly high for three connections. The reasons for this anomaly in this scenario are unclear. More generally, the advantages of using four connections in terms of improved wireless utilization and video quality equivalent to one connection are offset by the increased mean end-to-end packet delay, Table 6. However, as remarked earlier, the mean is still below 100 ms in this scenario.

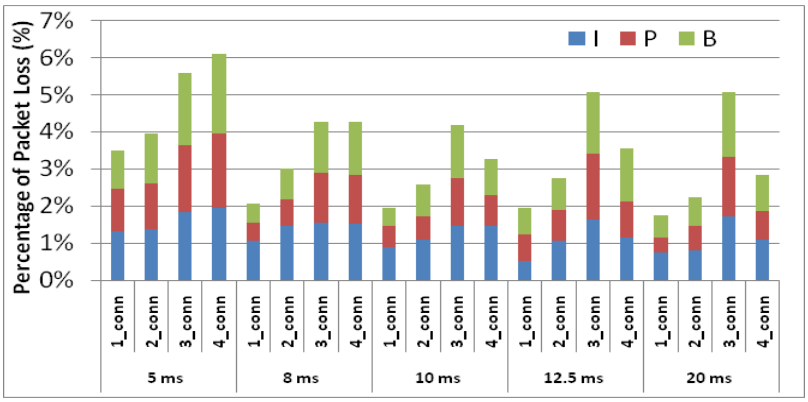

Figure 3. Mean percentage packet loss broken down according to frame type, by connection and TDD frame length, I = Intra-coded frame, P = Predictively-coded frame, B = Bi-predictively coded frame.

Table 6. Mean packet end-to-end delay according to frame length

Frame length:	*5 ms*	*8ms*	*10ms*	*12.5ms*	*20ms*
Connections	*Mean end-to-end packet delay (s)*				
1-conn	0.0195	0.017	0.017	0.020	0.035
2-conn	0.0446	0.029	0.029	0.027	0.036
3-conn	0.0725	0.024	0.028	0.023	0.039
4-conn	0.0982	0.069	0.067	0.073	0.062

5. ALTERNATIVE END-TO-END CONTROL

As an alternative approach to the use of multiple TFRC connections in this paper, there have been many attempts to improve the response of congestion control in the presence of a wireless channel without the need to intervene at the interface between the wired and wireless networks. The main aim is to avoid an inappropriate reduction in throughput due to wireless packet losses, as reduction in throughput due to random losses can cause precious bandwidth to be underutilized. This Section reviews several of these alternatives as they could be adapted to TFRC to decide which losses are taken into account in applying equation-based control.

End-to-end statistics can be gathered in order to distinguish congestion loss from wireless channel loss. In TCP Santa Cruz (Parsa and Garcia-Luna-Aceves, 1999), an increase in one-way delay is judged to be a sign of congestion loss if there is also packet loss. If there is no increase in delay at the same time as packet(s) are lost the cause is judged to be channel loss and no change is made to the throughput. By using one-way statistics, the intention is to avoid the impact of lost ACKs which can lead to data bursts when a congestion window's content is suddenly released. By combining packet loss with delay, TCP Santa Cruz also avoids the suppression of round-trip time measurements when retransmitting lost packets, which occurs because Karn's algorithm is employed.

Another method of distinguishing the type of loss is through an estimation of the variation of RTT (Barman and Matta, 2002). If the RTT varies significantly then congestion is declared but if there is limited variation then random channel errors are assumed. The intuition behind this decision is that congestion losses are assumed to be grouped, whereas wireless packet

losses are assumed to be sporadic and consequently will not cause the RTT to vary much. In work by Barman and Matta (2002), both a long-term and a short term estimator of RTT was kept, with the short term estimator used first unless the pattern of RTTs is erratic.

Cen et al (2003) develop an end-to-end loss differentiation algorithm, which is not only applicable to protocol congestion control but can also allow a video streaming application to decide upon an appropriate ratio of forward error correction and source coding bits, depending on whether losses or occurring because of the wireless channel conditions or not. In fact, this scheme is applied to TFRC, so that TFRC only changes its sending rate if congestion losses are suspected. Cen et al (2003) experimented with two schemes, namely: Biaz (Biaz and Viadya, 1999) and Spike (Tobe et al, 2000), to form a hybrid scheme (ZigZag) of their own. A feature of Biaz is that it allows for consecutive packets to be lost and not just sporadic losses. If an out-of-sequence packet arrives, based on an estimate of the minimum inter-arrival time, it is decided whether the packet arrived on time. If the packet arrived earlier than expected it is assumed to be because packets have been dropped at a buffer. If the packet arrived later than expected then it is also assumed to be due to congestion causing queuing at an intervening buffer. Spike uses the relative one-way delay that is the one-way delay without correction for clock skew. Spikes occur in the relative one-way delay during periods of congestion. Cen et al. found that the Biaz scheme in its original form tended to classify too many packets as congestion losses when the last link was a wireless link and was also a bottleneck link due to congestion. The Spike scheme was found to require tuning and was more appropriate for wireless backbones rather than access networks. The ZigZag scheme corrected some of the problems but it was reported in Cen et al (2003) that some scenarios still resulted loss misclassifications. Similar difficulties in classification seem to hold for other statistics based schemes.

Another class of end-to-end proposals employs cross-layer information to classify packet losses. In Görkemli et al (2008), physical-layer ARQ information is used to modify TFRC's estimate of the packet loss rate. That is an ARQ unit at the receiver requests retransmission of a packet a number of times until a timing threshold is exceeded, after which a wireless loss is declared. This information is then passed up the layers to the receiver application which sends feedback to the sender. However, losses from congestion are not relayed back to the TFRC sender. In general, the use of cross-layer information in Görkemli et al (2008) and others (Fu et al, 2002; Yang et al, 2007) suffer from the need to accord TFRC special treatment compared to other traffic.

6. CONCLUSION

Multi-connection congestion control adapts existing congestion controllers to all-IP networks that include a broadband wireless access link. In effect, they allow the congestion controller to accommodate wireless channel losses but still respond to congestion with the network edge and possibly the core. This in turn leads to improved network utilization, whereas previous observers have noticed a marked drop in utilization if congestion controllers are employed. However, for any wireless technology there still remain issues about how many connections should be used if the disadvantages of multi-connections are to be avoided. This study has found that though there is a small percentage increase in packet loss with four connections over just one, video quality remains equivalent because of the differential effect of packet loss

patterns when burst errors are present. There was also a small (in practical terms) increase in packet end-to-end delay. An important observation is that a longer WiMAX TDD frame size is favorable to video transport, though this may not be apparent unless tests are conducted across the whole of a network path and not just the wireless link. An advantage of the multi-connection method of congestion control is the reduction in state when it comes to handoff in a cellular WiMAX, which is important for a delay-intolerant application. Further investigation will examine this issue. Another advantage of the multi-connection method is that a portion of the additional throughput that results is available for error protection, either application layer FEC, or more promising, in terms of compatibility with existing physical-layer FEC, the use of source-coded error resilience.

REFERENCES

Andrews, J.G., et al, 2007. *Fundamentals of WiMAX*. Prentice Hall.

Apostopolous, J., et al, 2007. Path Diversity for Media Streaming. In: M. Van der Schaar and P.A. Chou, eds. *Multimedia over IP and Wireless Networks*, Academic Press, pp. 559-590.

Barman, D. and Matta, I., 2002. Effectiveness of Loss Labeling in Improving TCP Performance in Wired/Wireless Networks. *Proc. of Int. Conf. on Network Protocols*, pp. 2-11.

Biaz, S. and Vaidya, N., 1999. Discriminating Congestion Losses from Wireless Losses using Inter-Arrival Times at the Receiver. *Proc. of IEEE Symp. on Application-Specific Systems and Software Engr. and Techn.*, pp. 10–17.

Cen, S., et al, 2003. End-to-End Differentiation of Congestion and Wireless Losses. In *IEEE/ACM Trans. Networking*, Vol. 11, No. 5, pp. 703-717.

Chen, K.-C. and de Marca, J. R. B. eds., 2008. *Mobile WiMAX*. Wiley & Sons, Chichester, UK.

Chen, M. and Zakhor, A., 2005. AIO-TFRC: A Light-Weighted Rate Control Scheme for Streaming over Wireless. *Proc. of IEEE WirelessCom*.

Chen, M. and Zakhor, A., 2006. Multiple TFRC Connection Based Rate Control for Wireless Networks. In *IEEE Trans. Multimedia*, Vol. 8, No. 5, pp. 1045-1062.

Chen, M. and Zakhor, A., 2006a. Flow Control over Wireless Network and Application Layer Implementation. *Proc. of INFOCOM*, pp. 1-12.

Chen, M. and Zakhor, A., 2006b. An Enhanced All-in-One TFRC Protocol for Streaming Video in Wireless Networks. *Proc. of IEEE Int. Conf. on Image Processing*, pp. 1-4.

Crowcroft, J. and Oeschlin, P., 1998. Differentiated End-to-End Internet Servies using Weighted Proportional Fair Sharing TCP. In *ACM SIGCOMM Comput. Commun. Rev.*, Vol. 28, No. 3, pp. 53-69.

Damjanovic, D. and Wetzl, M., 2009. MulTFRC: Providing Weighted Fairness for Multimedia Applications (and others too!). In *ACM SIGCOMM Comput. Commun. Rev.*, Vol. 39, No. 3, pp. 6-11.

Ekstrom, K. et al, 2006. Technical Solutions for the 3G Long-Term Evolution. In *IEEE Commun. Mag*., Vol. 44, No. 3, pp. 38-45.

Floyd, S. and Fall, K., (1999). Promoting the Use of End-to-End Congestion Control in the Internet. In *IEEE/ACM Trans. Networking*, Vol. 7, No. 4, pp. 458-472.

Floyd, S. et al, 2000. Equation-Based Congestion Control for Unicast Applications. In *ACM SIGCOMM Computer Communication Review*, Vol. 4, No. 4, pp. 1-14.

Fu, Y., et al, 2006. TCP-Friendly Rate Control for Streaming Service over 3G Network. *Proc. of Int. Conf. on Wireless Comms., Networking and Mobile Computing*, pp. 1–4.

Görkemli, B., et al, 2008. Video Streaming over Wireless DCCP. *Proc of IEEE Int. Conf. on Image Processing*, pp. 2028–2031.

Haßlinger, G. and Hohlfeld, O., 2008. The Gilbert-Elliott Model for Packet Loss in Real Time Services on the Internet. *Proc. of 14th GI/ITG Conf. on Measurement, Modelling, and Eval. of Comp. and Commun. Systs.*, pp. 269-283.

Handley, J. et al, 2003. TCP-Friendly Rate Control (TFRC): Protocol Specification. *IETF*, RFC 3448.

IEEE, 802.16e-2005, 2005. IEEE Standard for Local and Metropolitan Area Networks. Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems.

Hsiao, P.-H., Kung, H. T., and Tan, K.-S., 2003. Streaming Video over TCP with Receiver-based Delay Control. In *IEICE Trans. Commun.*, Vol. 86, No. 2, pp. 572-584.

Klaue, J. et al, 2003. EvalVid - A Framework for Video Transmission and Quality Evaluation. *Proc. Int. Conf. on Modeling Techniques and Tools for Computer Performance*, pp. 255-272.

Kuo, F.-C. and Fu, X., 2008. Probe-Aided MulTCP: An Aggregate Congestion Control Mechanism. In *ACM SIGCOMM Comput. Commun. Rev.*, Vol. 38, No. 1, pp. 17-28.

Liang, Y.J. et al, 2008. Analysis of Packet Loss for Compressed Video: Effect of Burst Losses and Correlation Between Error Frames. In *IEEE Trans. Circ. Syst. Video Technol.*, Vol. 18, No. 7, pp. 861-874.

Lin, Y.-B. and Pang, A.-C., 2005. *Wireless and Mobile all-IP networks*. Wiley and Sons, Indianapolis, IN.

Ott, D.E. et al, 2004. Aggregate Congestion Control for Distributed Multimedia Applications. *Proc. of IEEE INFOCOM*.

Padyhe, J. et al, 1998. Modeling TCP Throughput: A Simple Model and its Validation. *Proc. of ACM SIGCOMM*, pp. 303-314.

Parsa, C. and Garcia-Luna-Aceves, J., 1999. Improving TCP Congestion Control over Internets with Heterogeneous Transmission Media. *Proc. of Int. Conf. on Network Protocols*, pp. 213–221.

Rhee, I. and Xu, L., 2005. Limitations of Equation-Based Congestion Control. *Proc. of ACM SIGCOMM*, pp. 49-60.

Schwarz, H.et al, 2007, Overview of the Scalable Video Coding Extension of the H.264/AVC Standard. In *IEEE Trans. Circ. Syst. Video Technol.*, Vol. 17, No. 9, pp. 1103–1119.

Shen et al, 2009. Receiver Playout Buffer Requirement for TCP Video Streaming in the presence of Burst Packet Drops. *Proc. of London Commun. Symposium*.

Stewart, R., ed., 2007. Stream Control Transmission Protocol. Internet Engineering Task Force, RFC 4960.

Tobe, Y. et al, 2000. Achieving Moderate Fairness for UDP Flows by Path-Status Classification. *Proc. IEEE Conf. on Local Computer Networks*, pp. 252–61,.

Tullimas, S., et al, 2008. Multimedia Streaming Using Multiple TCP Connections. In *ACM Trans. Multimedia Computing, Commun. and Applications*, Vol. 4, No. 2, article 12.

Tappayuthpijam, K. et al, 2009. Adaptive Video Streaming over a Mobile Network with TCP-Friendly Rate Control. *Proc. of Int. Conf. on Wireless Communs. and Mobile Computing*, pp. 1325-1329.

Wang, Y., et al, 2005. Multiple Description Coding for Video Communications. In *Proceedings of the IEEE*, Vol. 93, No. 1, pp. 57-70.

Wenger, S., 2003. IP over H.264/AVC. In *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 13, No. 7, pp. 645-656.

Wetzl, M. and Stadler, W. 2005. User-Centric Evaluation of TCP-friendly Congestion Control for Real-Time Video Transmission. In *Elektrotechnik und Informationstechnik*, June, pp. 1-10.

Widmer, J. et al, 2001. A Survey on TCP-Friendly Congestion Control. In *IEEE Network Mag.*, Vol. 15, No. 3, pp. 28 - 37.

Wiegand, T., et al, 2003. Overview of the H.264/AVC Video Coding Standard. In *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 13, No. 7, pp. 560-576.

Yang, F., et al, 2007. End-to-end TCP-friendly Streaming Protocol and Bit Allocation for Scalable Video over Wireless Internet. In *IEEE J. on Sel. Areas in Commun.*, Vol. 22, No. 4, pp. 777-790.