DCCP Video Streaming over Multiple Connections in the Wireless Internet

Salah M. Saleh Al-Majeed

Martin Fleury

University of Essex
Dept. of Computer Science and Electronic Engineering
Wivenhoe Park
Colchester CO4 3SQ
United Kingdom
{ssaleha, fleum}@essex.ac.uk

Abstract: Multiplexing a single video stream over several Datagram Congestion Control (DCCP) connections is a way of coping with wireless channel losses *and* traffic congestion in the wireless Internet, without the need for complex cross-layer intervention. Multi-connections introduce the need for data re-ordering at the receiver. This paper considers the potential delay at an uplink destination on the Internet and extends this to a scheme for user-to-user mobile device via the wired Internet. It also considers the impact on video quality of packet drops due both to channel loss and router buffer overflow, when the DCCP congestion controller is applied. Results for IEEE 802.16e (mobile WiMAX) broadband wireless stages show a worthwhile gain in video quality from using three or more connections over a single connection but with start-up delay due to the need to avoid possible buffer underflow. More connections will be necessary to preserve sufficient wireless link utilization if the path is lengthened to reach an end device that is also mobile.

Keywords: DCCP; multi-connection; video streaming; wireless Internet.

1 Introduction

Datagram Congestion Control (DCCP) (Kohler et al., 2006) is along with Stream Control Transmission Protocol (SCTP) (Stewart, 2007) a Standards-based way of providing congestion control for video streaming without the disadvantages of TCP, which are described in the following. DCCP supports delay-sensitive streaming over UDP without TCP's delay-inducing reliability. It has two varieties: TCP-like congestion control and TCP-Friendly Rate Control (TFRC) (Handley et al., 2003), one variety of which is selected during initial handshaking. TFRC is employed in this study as it reduces the saw-tooth like rate changes associated with TCP congestion control mechanisms. 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. (SYN flooding occurs when an attacker sends a sequence of the synchronization messages, which form part of TCP's connection procedure. These connection initializations are a form of denial-of-service attack.) However, an interesting feature of SCTP is support for multi-streaming with optional out-of-order delivery to avoid TCP's potential head-of-line blockages. In this paper, we extend DCCP to include multi-connection streaming in which a single video stream is multiplexed across the connections. However, we apply the multi-connections to improve the wireless channel utilization in a similar way to MULTTFRC (Chen and Zakhor, 2005), as single connection streaming may fail to deliver the video at the frame rate of an end-user's display.

In multi-connection DCCP video streaming, the DCCP congestion controller's main role when congestion occurs 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, just as TCP can, TFRC can misinterpret as congestion packet losses due to wireless interference and noise. Though cross-layer approaches to avoid misinterpretation are possible, these are complex to implement and inflexible. 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 rate across the remaining connections. We assume a broadband wireless access link with a core wired-network. Specifically, IEEE 802.16e (mobile WiMAX) (IEEE, 2005) is modeled as the broadband wireless link that backhauls the video stream to the wired Internet. In a network consisting of a wireless and wired portion, a video stream is subject to packet loss due to wireless channel conditions *and* to traffic congestion on the wired network, as well as due to congestion on the access network.

In video streaming across an IP network unreliable UDP transport serves to reduce delay at the expense of some packet loss, while application-layer TCP emulation (Widmer et al., 2001), such as TFRC, acts as a form of cooperative congestion control (assuming most other traffic is carried through TCP transport). However, TCP emulation by the application is *not* the same as TCP. TCP itself is unsuitable for delay-variation intolerant video streaming, because it introduces unbounded delay in support of a reliable service. Instead, TFRC emulation mimics the average behavior of TCP, but is not 'reliable' and does not result in the 'saw-tooth'-like rate fluctuations that arise from TCP's aggressive congestion control algorithms, which can cause disconcerting quality fluctuations at an end-user's display if the compression ratio is varied according to the available bandwidth (through bitrate transcoding of stored video or altering the quantization parameter of live video).

An interesting research question is what would occur if multiple DCCP connections were opened in the uplink (UL) from a WiMAX subscriber station (SS) to base station (BS) in the presence of cross traffic from other mobile SSs. Thus, congestion also occurs on the uplink as well as fluctuating wireless channel conditions. In this situation, congestion will occur (Niyato and Hossain, 2005) at the WiMAX real-time polling service (rtPS) queue and packet loss will occur over the wireless channel (Issa et al., 2010). We also consider the effect of packet loss as the multiple connections pass over the core IP-network when other traffic sources contend for access to buffers at intermediate routers. Propagation over the wired network is realistically assumed to be error-free, as it may well consist of optical fiber links. In this scenario, uplink streaming is a first step to video clip exchange between users or it could be part of an interactive Internet Protocol TV (IPTV) service (Saleemi and Ilius, 2010).

This paper's main contribution is the finding that as the number of connections increases, reduced packet loss leads to improved video quality, because of the reduced sending time in sending the same video data. In contrast, in (Chen and Zakhor, 2005) improved video quality comes by increasing the quantity of video data that can be sent over the multiple connections. Of course, increased video data implies a lower compression rate and, hence, higher quality. Unfortunately, if the number of connections varies, as it does in (Chen and Zakhor, 2004; Chen and Zakhor, 2005; Chen and Zakhor, 2006) then sending rate oscillations occur, necessitating changes to the compression ratio. However, we show that the quality increases anyway without the need to change the compression ratio, by keeping the number of connections constant. This is because with multiple DCCP connections, DCCP is better able to control its sending rate.

In fact, in the original work on TFRC (Handley et al., 2003) the design assumed a high number of streams and has special measures if the number of streams is not high. Possibly, the difference in findings occurs because in work on MULTTFRC (Chen and Zakhor, 2004; Chen and Zakhor, 2005; Chen and Zakhor, 2006) apparently no account of the impact of cross-traffic occurs except to test the fairness of the scheme to coexistent TCP traffic.

Earlier work did not give much consideration to the effect of congestion in the TFRC feedback path. Nor was the issue of how a single video stream is multiplexed onto multiple connections addressed. In this paper, the unit of multiplexing was taken to be a Group-of-Pictures (GOP) (Sadka, 2006) with an Intra-refresh rate of 15 frames. Just as in Peer-to-Peer video streaming, when video is delivered as chunks from a number of sources, there is a need to employ a reordering buffer to correct out-of-order delivery. The resulting start-up delay in the scenario tested was about 6 s, but this is compensated for by the increased wireless link utilization that occurs with multi-connection schemes.

The remainder of the paper is organized as follows: Section 2 reviews related work, before Section 3 considers the modeled scenario. Section 4 presents an evaluation, with Section 5 concluding.

2 Related research

The research in (Chen and Zakhor, 2004) proposed MULTTFRC originally as a form of downlink control. Any single TFRC connection responds to packet loss by reducing its output rate by increasing the inter-packet gap. MULTTFRC represents a lightweight way to retain TFRC for the Internet path but avoid complex means of suppressing wireless channel packet loss feedback to TFRC. Misinterpretation of channel packet losses as congestion is the source of under-utilization of the wireless channel if a single TFRC connection were to be used. Alternatively, in the SNOOP approach (Balakrishnan et al., 2007), wireless link packet loss feedback to the congestion controller is suppressed by a SNOOP unit, which requires intervention at the data-link layer and cross-layer interaction. By suppressing channel loss reporting, SNOOP allows a congestion controller to respond only to packet loss by congestion. Interestingly in the work of Chen and Zakhor, 2004, strengthened Forward Error Correction was applied to counter increased packet loss from MULTTFRC. However, this has not been our experience with static scheduling of connections. It should be noted that MULTTFRC uses a dynamic scheduling of connection numbers.

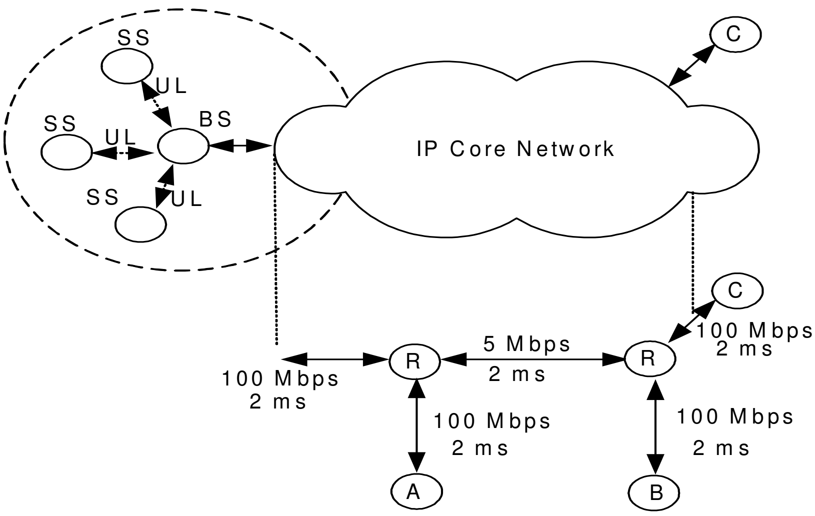
In the most definitive account of MULTTFRC so far, the work in Chen and Zakhor, 2006, there is no account of how a single video stream is multiplexed onto the multiple connections using dynamic scheduling or what the resulting video quality is in quantitative terms. Only generic packet loss and delay statistics are reported, even though the type of error pattern is known to change the video quality (PSNR) by several dBs. Other work on MULTTFRC and its variants such as Yogesh et al., 2006 appears largely confined to analysis of a generic link without other traffic.

In the research in Tappayuthpijam et al., 2009, single connection TFRC was applied over the Long-Term Evolution (LTE) 4G wireless technology. The rationale (Tappayuthpijam et al., 2009) is that, in this type of mobile network, retransmissions at the data-link layer effectively remove packet loss at the expense of end-to-end delay and throughput. TFRC was combined with the Scalable Video Coding (SVC) extension to the H.264/AVC (Advanced Video Codec), in such a way that layering was adapted to the bitrate. The results are promising in terms of reduced packet losses, reduction of streaming interruptions, delay, and buffering compared to not using TFRC. However, the approach in Tappayuthpijam et al., 2009 potentially reintroduces the risk of unbounded delay and head-of-line blocking which make TCP unsuitable for real-time services. There is also the overhead of maintaining retransmission state at the transmitter and the delay arising if retransmissions are still occurring when a handoff occurs.

Other examples of single connection TFRC for video streaming over wireless include Fu et al., 2006 and Görkemli et al., 2008. In Fu et al., 2006, SVC is again used in conjunction with DCCP in a scenario in which there are wireless access links to the core network at the sender and receiver sites. Therefore, this work investigates uplink as well as downlink streaming. Similarly in Görkemli et al., 2008, lower layer information is employed to modify TFRC's estimate of packet loss rates, in this case, the physical (PHY) layer ARQ information. The work in Görkemli et al., 2008 appears to have been anticipated in Yang et al., 2007, though TFRC was not used in that work. Both Fu et al., 2006 and Görkemli et al., 2008 are cross-layer modifications and, hence, suffer from the need to accord TFRC special treatment compared to other traffic.

In Juan et al., 2007, multiple connections over an IEEE 802.16e link were used to send different layers of a scalable video stream. However, the scheme was only tested for two layers and quality testing does not appear to have accounted for lost packets. This is an important issue in H.264/SVC, because of the complex inter-dependencies between the layers. If key packets are lost through channel error or dropped at buffers then many other encoded video-bearing packets that are dependent on them have to be discarded as well.

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



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 BS has allocated bandwidth to each SS, each SS must manage its queue according to the data arrival rate from user applications. In WiMAX Point-to-Multipoint

(PMP) mode (Andrews, 2007; Nuaymi, 2007), there is no SS-to-SS communication unless it is via the BS. WiMAX networks support multiple service classes to accommodate heterogeneous traffic with varying requirements. WiMAX's rtPS is most suitable for real-time service flows that generate variable-sized packets on a periodic basis such as video (Hosein, 2008), 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, as this type of queue is widely implemented. Queue sizes were all set to fifty packets. For the projected frame rate (30 frame/s) and packet sizes (refer to Section 3.2) this value amounts to at least one second. This is enough to absorb delay jitter generated at the BS and at the same time allows enough time for interactive TV (Issa et al., 2010). For the same reason a similar value was selected before removing packets into the decoder playout buffer in the WiMAX system of Hillestad, 2007. This value was not made larger because mobile devices, being battery-powered, should optimize energy consumption of which buffer memory is a static consumer (Segars, 2001) as well as a dynamic consumer through frequent accesses.

The WiMAX system operating in PMP mode was simulated by well-known ns-2 simulator (v. 2.29) augmented by a WiMAX module (Tsai, 2006). Mean data points are the average of at least ten runs. The simulator is allowed to reach steady-state over 20 s with other traffic passing over the network.

The PHY settings selected for WiMAX simulation are given in Table 1, with additionally the MAC settings defaulted from Tsai et al., 2006. The DL/UL ratio is not intended to be realistic but to aid in testing multiple-connection TFRC, as in practice the DL would be allocated the majority of the bandwidth. The antenna is modeled for comparison purposes only as a half-wavelength dipole. The Gilbert-Elliott 'bursty' channel model is further explained in Section 3.4. The frame length is significant, as a longer frame reduces delay at the BS by permitting more data to be removed from any queues at each polling time. The value of 20 ms is at the high end of the available durations in the IEEE Standard (IEEE, 2005) in order to reduce this source of queuing delay for real-time video streaming.

Table 1 Simulated WiMAX settings, OFDMA = Orthogonal Frequency Division Multiple Access, QAM = Quadrature Amplitude Modulation, TDD = Time Division Duplex

Parameter	*Value*
PHY	OFDMA
Frequency band	5 GHz
Duplexing mode	TDD
Frame length	20 ms
Max. packet length	1024 B
Raw data rate	10.67 Mbps
IFFT size	1024
Modulation	16-QAM 1/2
Guard band ratio	1/8
DL/UL ratio	1:3
Path loss model	Two-ray ground
Channel model	Gilbert-Elliott
MS transmit power	250 mW
BS transmit power	20 W

Approx. range to MS	0.7 km
Antenna type	Omni-directional
Antenna gains	0 dBD
MS antenna height	1.5 m
BS antenna height	32 m

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/AVC codec (Wiegand et al., 2003) and split between the multiple TFRC connections. The other SSs are simply introduced as sources of competing traffic across the wireless link and do not indicate the likely size of a WiMAX network, which obviously could be larger. For example, in Hillestad et al., 2007 it was shown that fixed WiMAX could support 20 SS when streaming video at about 500 kbps per stream. Tens of subscribers are supportable according to Cicconetti et al., 2006, and fixed WiMAX is said to support (Hoyman, 2005) a throughput of 10-20 Mbps for a realistic spatial distribution of SSs. In So et al., 2010, the analysis of capacity is extended to mobile WiMAX suggests 14 users of mobile TV in a ‘lossy’ channel with single antenna and simple scheduler, with 16 users after enhancements, according to modulation scheme and DL/UL ratio. A trace file was input to ns-2

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	

and packet losses recorded in the output. The output serves to calculate the PSNR. Video quality comparisons were made under the EvalVid environment (Klaue et al., 2003). As a test, we used the *Paris* clip H.264/AVC VBR-encoded at 30 frame/s at Common Intermediate Format (CIF) (352×288 pixel/frame) with initial quantization parameter set to 26 (from a range 0 to 51). The slice size was fixed at 900 B. The advantage of fixing the slice size at the codec is that VBR packets are not subsequently at risk of fragmentation within the network, provided the size is below the maximum transport unit size.

Paris consists of two figures seated around a table in a TV studio setting, with high spatial coding complexity. H.264/AVC’s Baseline profile was selected, as this is more easily supported by mobile devices because of its reduced computational overhead. The Intra-refresh rate was every 15 frames with IPBB…I structure, i.e. the GOP size was 15. 1065 frames were transmitted, which at 30 frame/s amounts to 35.5 s of display time. Simple Previous Frame Replacement (PFR) was set for error concealment at the decoder in order to compare results from those of others.

Table 2 records the simulated traffic characteristics for the three SSs' communication with the BS. Network Adaptation Layer Units (NALUs) from the H.264/AVC codec were encapsulated with Real Time Protocol (RTP) headers. After the addition of IP headers, these in turn formed a single WiMAX MAC Packet Data Unit (MPDU), which are variable-sized WiMAX packets. For simplicity, a WiMAX MPDU is now referred to as a packet.

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, 2004. 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 (1) models TCP New Reno to find the sending rate:

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

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, 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 (Stewart et al., 2007), which is why it is unwise to use a simplified form of (1). Clearly packet loss and round-trip time cause the throughput to decrease in (1), whereas other terms are dependent on these two variables in the denominator.

In our variant to standard TFRC, the packet size, s, in the TFRC equation was dynamically altered according to the EvalVid-created trace file sizes. This variant makes for more responsive control rather than the mean packet length employed in the original 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. Setting a mean packet length is inappropriate for interactive multimedia applications. The underlying TFRC transport protocol was set to UDP, as is normal.

Coexisting rtPS queue Constant Bit Rate (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 simply selected to fully occupy the DL link capacity.

3.3 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) in order to avoid confusing propagation delay with re-ordering delay. 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 fifty 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 core network edge before entry into a corporate or campus network.

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.

3.4 Management of connections

To systematically test the effect of multiple DCCP connections the number of DCCP connections was incrementally stepped up in successive experiments. In our experiments, a single queue was segmented into GOPs (15 frames). (Recall from Section 1 that a GOP consists of one intra-coded I-frame followed by a set of inter-coded frames that use the I-frame as a predictive anchor.) Each connection was statically allocated its GOPs, which are taken in interleaved manner from the video sequence. As previously mentioned, this assumes that a re-ordering buffer is available at the receiver.

3.5 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 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. However, the bad state packet loss probability, P_B, was also varied as [0.01, 0.02 , …, 0.1]. In this way, we were able to judge the effect of worsening burst error channel conditions.

4 Evaluation

4.1 Wireless link response

Initial investigations considered the WiMAX link alone in Figure 1. Table 3 shows the average over ten runs of the data-rate over time when transmitting the Paris clip over multiple connections, for two different WiMAX frame sizes: the default of 20 ms from Table 1 and 5 ms (frame duration code 2 in the Standard (IEEE, 2005)). For a frame size of 5 ms and one connection the sending period is well in excess of the display time (35.5 s) of the video clip. Therefore, given a choice the longer frame size is preferable and in general the system is sensitive to choice of frame size. Clearly, DCCP 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 DCCP sharply reduces its overall sending rate in response to packet loss.

Figure 2 plots the video stream packet drop rate relative to channel packet error rate. Included in the percentages in Figure 2 are any additional packet losses arising from buffer overflow at an SS caused by the SS packet scheduler being otherwise occupied servicing different queues or other SSs. In this Figure, the shorter frame size is employed, which gives more favorable results, at least in terms of packet loss, than the longer frame size. As will be observed, no strong effects result from increasing the number of connections. Moreover, for all but the highest error rates the packet loss rate is below 10%.

4.2 Network response

An interesting comparison is with the throughput when the core network is included, Table 4. In Table 4, as more connections are added the throughput rises and the sending period reduces to well below the display time (35.5 s) of the video clip.

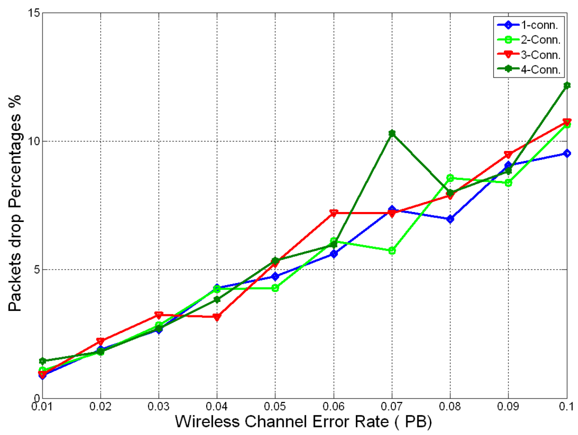
Table 3 Mean sending periods and throughputs from the video streaming SS to the WiMAX BS

No. of connection	*SS to BS (s) frame size 5 ms*	*Thruput (kbps)*	*SS to BS (s) frame size 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 Sending periods and throughputs from the video streaming SS to the core network destination (node C in Figure 1)

No. of connections	*SS to node c frame size 20 ms*	*Throughput (kbps)*
1-conn	35.2	444
2-conn	22.4	690
3-conn.	21.6	716
4-conn.	15.6	991

Figure 2 Mean packet drop rate for an increasing number of connections, according to channel error rate



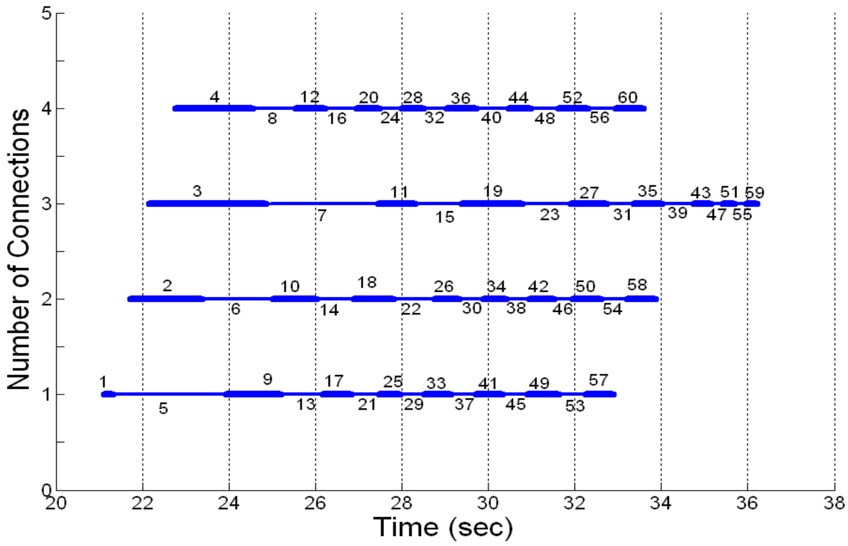
There is a similar pattern to the throughputs in Table 3 but the datarates are reduced to when streaming only over the WiMAX link. We interpret this effect as not due to DCCP's response to packet loss but due to its response to the increased round trip time caused by queuing delay in the buffer prior to the bottleneck link in Figure 1. This is confirmed by the increase in per slice/packet end-to-end delay as more connections are added, Table 5. In effect, the packets from other connections intervene in the router buffers causing an increase in latency.

More significantly for reconstruction of the video stream is the GOP ordering, which for four connections is shown in Figure 3. To avoid a sudden injection of traffic into the network, connection starting times were offset by 0.5 s. Notice that the first GOP only contains parameters that are fixed throughout the sequence (Wiegand et al., 2003), a feature of the H.264/AVC codec. Therefore, this GOP is transported more quickly. GOPs 2, 4, and 5 in Figure 3 have elongated sending times as discussed in the next paragraph. GOPs 3 and 7 of connection 3 have especially elongated sending times, which may also be due evidence of access starvation caused by the other connections. This possibility is discussed below. From the time 34 s onwards, connection 3 rapidly catches up with its GOP sending schedule after the other connections have completed their schedules.

Table 5 Mean per slice/packet end-to-end delay

No. of connections	*Mean end-to-end delay (s)*
1-conn.	*0.035*
2-conn.	*0.036*
3-conn.	*0.039*
4-conn.	*0.062*

Figure 3 Example arrival sequence at the receiver (node C in Figure 1) showing the start and end times. Connection start times are staggered by 0.5 s.



Key: Each number refers to the transmission interval of a GOP ordered in the sequence within the *Paris* video clip. GOP 1 contains the parameter-set for the sequence. GOPs 2, 3, 4, and 5 sending intervals show elongation at the start of the transmission. GOP 7 in connection 3 shows elongation of the sending interval due to presumed buffer starvation. GOPs 39, 43 … 59 in connection 3 show faster throughput after the release of resources by the other connections.

A noticeable feature of Figure 3 is the lengthier start-up periods in sending initial GOPs on each of the connections. This does mean that about 6 s of packets (amounting to 90 frames) should be stored in the reordering buffer, to avoid the possibility of subsequent underflow in the decoder's playout buffer. 6 s is longer than an ideal start-up time of around 2 s but not too large to be objectionable to the user. Interestingly, when compared to the behavior reported in Chen and Zakhor, 2004 and repeated in Chen and Zakhor, 2006, for MULTTFRC there are periods of at least three seconds when the throughput is approximately over half the peak rate. The aggregate throughput also may oscillate. In fact, on finding this problem, we compared with Chen and Zakhor, 2004 and discovered that allowance was made for a 10 s start-up buffer before beginning decoding also to avoid buffer underflow. However, Chen and Zakhor, 2004 used data from MPEG-4 at a lower 10 frame/s to test buffer occupancy.

The cause of the initial lengthier start-up periods may be a combination of factors. However, the early response of DCCP appears to be implicated. The initial rate of DCCP is set to one packet/s and no default settings for round-trip time or packet loss rate are used in the throughput equation (1). Normally, if no acknowledgement arrives within two round trip times then DCCP reduces its sending rate by half and goes into a slow start, similarly to TCP. However, the initial default value of the no-feedback timer is set to 2 s, which implies that DCCP's initial rate may be prolonged if acknowledgments are lost or delayed. If acknowledgement drops or delays still occur then it is possible that the rate will be halved again before slow-start. However, the timeout interval will be shorter, as it is now given by:

$$timeout = \max(4r, (2s)/DCCP), \tag{2}$$

where r is the estimated round-trip time, as before s is the packet size, and *DCCP* is the sending rate given by (1) for TFRC. Though in Chen and Zakhor, 2006 it was acknowledged that drastic reductions in sending rate could occur due to the onset of slow-start, this was attributed to heavy packet loss and not to the loss or delay of acknowledgments, without data packet loss necessarily occurring.

Corresponding to Figure 3, Figure 4 plots individual throughputs and the aggregate throughput. As might be expected from Figure 3, throughput gradually climbs until a plateau is reached. There is evidently some unfairness between the DCCP flows as connection 3 needs to prolong its delivery because of lower throughput at an earlier stage. However, there are less oscillations in rate than reported for MULTTFRC (Chen and Zakhor, 2004; Chen and Zakhor, 2006), which is explained by the static scheduling scheme employed by us. Differences in the rates of connections 1, 2, and 4 are explained by responses to packet loss which affects each connection in a random manner during a simulation. Whenever, a loss occurs from equation (1) DCCP reduces its throughput. Evidently from the behavior of connection 3 there is also an impact from the arrival times at buffers, noticing that in Figure 3 there is a staggered start time for the connections. From Figure 3 also, connection 3 prolongs the delivery of its GOPs while the other connections are still present. We note that TFRC has been reported to produce access starvation (Choi and Handley, 2007) when there are other TCP connections sharing the same link and we speculate that it is possible for TFRC to also cause starvation when other TFRC connections share the same path. However, further investigation is reserved for future work.

Packet loss over time displays an oscillatory pattern for the example in Figure 5, which is why it is unwise to rely on mean packet loss statistics alone. Based on the packet loss patterns the average objective video quality (PSNR) was found when increasing the number of connections. The results are recorded in Table 6. The frame sizes are adjusted in Table 6 to account for the buffer underflow that would occur were the shorter frame size to be used throughout the network path. However, counter-intuitively, employing a shorter frame size for a few connections over the WiMAX link alone results in lower video quality than when sending over the complete path. This is best explained by buffer overflow at the streaming SS, caused by the short WiMAX frame size, rather than packet losses on the WiMAX wireless channel.

When the number of connections increases, DCCP is better able to regulate its rate and the video quality increases over the single wireless link. Notice, however, that using smaller frame size even when the video quality is high can lead to excessive delay at the SS buffers if traffic is heavy.

Figure 4 Example run showing throughput over time for individual connections and the aggregate throughput

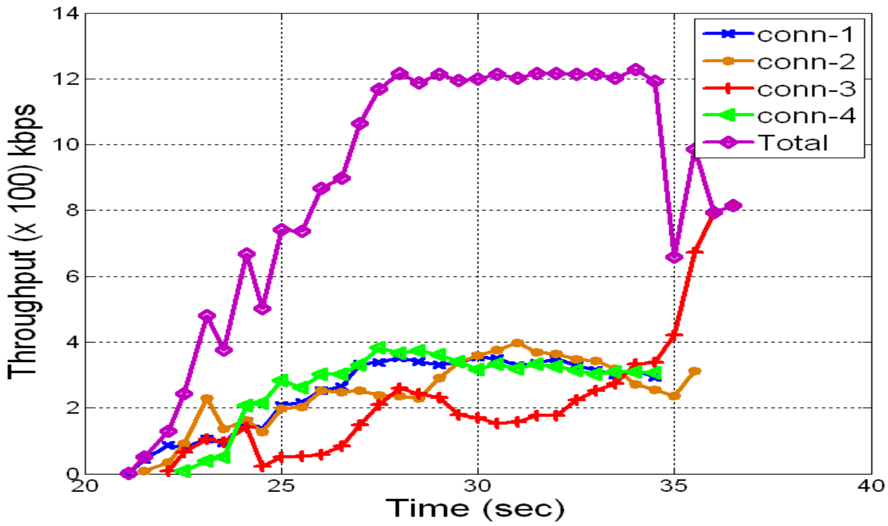


Table 6 Video quality (PSNR) according to number of connections

No. of connections	*PSNR (dB) recorded at WiMAX base station (frame size 5 ms)*	*PSNR (dB) recorded at node C in Fig. 1 (frame size 20 ms)*
1-conn.	26.72	31.84
2-conn.	31.32	32.34
3-conn.	35.92	33.15
4-conn.	35.32	33.34

4.3 Mobile user-to-user response

In further experiments, the scenario of Figure 1 was extended to include a further mobile stage. Thus, in Figure 6, video streaming takes place from a SS to the isolated SS in the right-hand WiMAX access network, that is user-to-user device streaming. For simplicity of analysis, extra congesting traffic in the rightmost WiMAX network is omitted. The UL/DL sub-frame ratio is now better set to be split equally. However, our experience is that in the given scenario this change has little effect. From experience gained with the sensitivity to the transmission frame size, larger sizes were tested. To reduce the need for the reordering buffer, the multiplexing granularity was changed to be on the packet level rather than the GOP level. In general, the problem with just one connection was accentuated by in effect increasing the impact of the wireless channel. The result is that more connections are needed to cope with the need to effectively utilize the wireless channel.

Figure 5 Aggregate packet loss numbers for all connections for a sample run over time

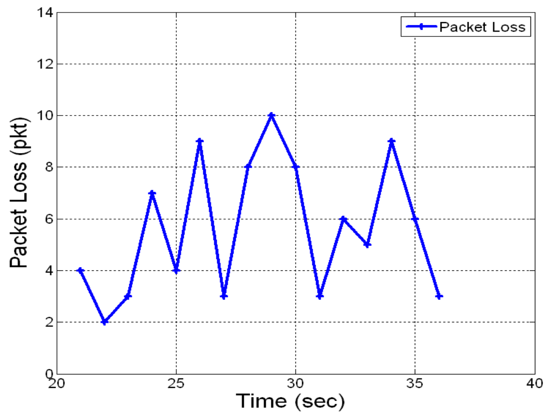
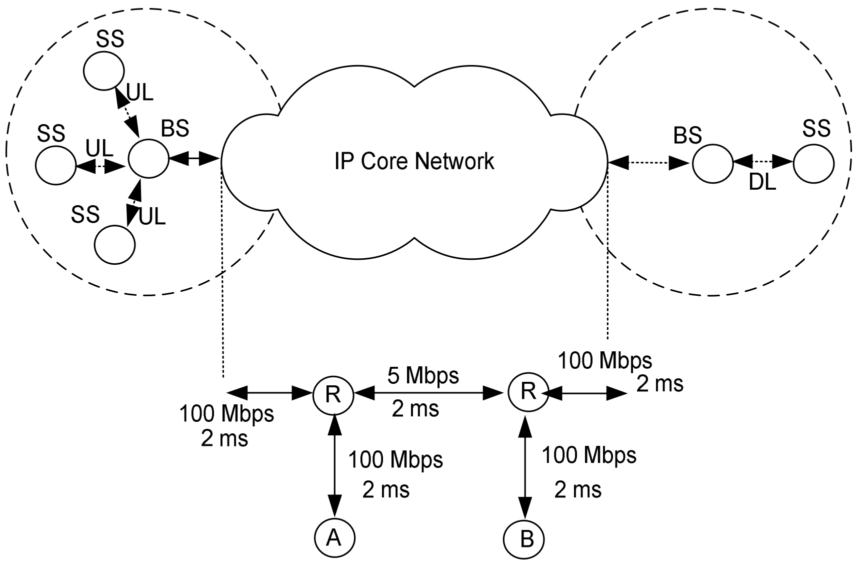


Figure 6 User-to-user device network with inset showing routing across the intervening network, A and B being sources and sinks, and R = router.



In Table 7, the period over which the test clip is sent is far longer than the time to display the clip at a rate of 30 Hz, i.e. about 36 s. To achieve sufficient wireless channel utilization a combination of at least six connections and a frame size of 12.5 ms is required. This is observable from Table 7 when the sending period is 34 ms, close to the 35.5 ms duration of the video sequence. For other combinations of frame size and connection number, the sending period exceeds the duration of the video sequence. In fact, with the addition of the extra stage, a frame size of 20 ms no longer results in the better performance, as the sending period is too long. Therefore, increasing the frame size improves performance for uplink streaming but it can cause increased delay at the base station while the downlink service is completed. Therefore, a different or adaptive frame size is required depending on the direction of services.

However, end-to-end packet delay remains moderate, Table 8, though because of the intervention of the DCCP congestion controller and the potential for self-congestion, end-to-end delay is a weak indicator of performance. Jitter with a single connection is high, leading to the need for a larger jitter buffer if not a reordering buffer.

As the total number of packets was 4739, from Table 9 no more than 7% of the slices were on average lost, whereas Quality of Experience (QoE) subjective testing (Agboma et al., 2008) suggests that broadly a round figure of 10% losses is needed on mobile devices to cause

Table 7 Mean throughputs and sending periods for user-to-user device streaming

	Frame size					
	8 ms		*12.5 ms*		*20 ms*	
	Throughput (kbps)	SS to SS (s)	Throughput (kbps)	SS to SS (s)	Throughput (kbps)	SS to SS (s)
1 conn.	187	148	187	152	100	179
4 conn.	658	41	745	37	466	60
6 conn.	630	40	797	34	591	46
8 conn.	653	38	910	29	706	39

Table 8 Mean packet end-to-end delay and jitter for user-to-user device streaming

	Frame size					
	8 ms		*12.5 ms*		*20 ms*	
	Delay (s)	Jitter (s)	Delay (s)	Jitter (s)	Delay (s)	Jitter (s)
1 conn.	0.041	0.034	0.063	0.033	0.139	0.063
4 conn.	0.048	0.009	0.057	0.008	0.081	0.012
6 conn.	0.047	0.007	0.058	0.006	0.077	0.008
8 conn.	0.046	0.006	0.061	0.005	0.076	0.007

Table 9 Mean packet loss numbers

	Frame size		
	8 ms	12.5 ms	20 ms
1 conn.	331	202	119
4 conn.	348	217	253
6 conn.	453	272	294
8 conn.	469	320	304

Table 10 Mean per frame PSNR and variation (stdv = standard deviation)

	Frame size					
	8 ms		12.5 ms		20 ms	
	PSNR (dB)	Stdv. (dB)	PSNR (dB)	Stdv. (dB)	PSNR (dB)	Stdv. (dB)
1 conn.	28.85	5.11	29.21	3.16	33.51	4.69
4 conn.	26.93	3.03	29.88	4.32	28.99	3.55
6 conn.	26.11	3.17	28.47	3.41	28.54	4.06
8 conn.	25.53	2.95	27.22	3.24	28.50	3.61

significant deterioration in delivered video quality. Turning to objective video quality, Table 10, though PSNR with one connection is generally better, poor channel utilization renders this quality unusable for streaming, as it also does in this scenario for a frame size of 20 ma.

4.4 Discussion

The main point of TFRC (Handley et al., 2003), i.e. effectively single-connection DCCP, is to maintain compatibility with TCP within the conventional wired Internet but to also avoid the deficiencies of TCP on a wireless network, as outlined in Section 1. TCP's congestion control is a co-operative process which seeks to avoid the acquisition of too much bandwidth by any one transport protocol. The concept TCP-friendliness (Floyd and Fall, 1999) on the conventional Internet was introduced to avoid the risk of congestion collapse arising from the use of uncooperative protocols. In Chen and Zakhor, 2006, the issue of whether multiple TFRC connections were fair to TCP was examined in detail. It was found that a single TCP connection reduced its throughput in proportion to the increase in roundtrip caused by the extra TFRC connections. However, the same behavior would occur if extra TCP connections had been introduced.

As an alternative approach to the multiple TFRC connections, there have been many attempts to improve the behavior of TCP by intervening at the interface between the wired and wireless Internet. For example, in the Snoop approach (Balakrishnan et al., 1997) a module resides on a base station. The Snoop module checks all traffic for TCP flows and carries out local retransmissions when a packet is lost on the wireless link. It is necessary for the Snoop module to suppress ACKs while retransmission is going on. Though the Snoop approach can be extended to TFRC it implies the introduction of additional delay to a real-time service by the introduction of retransmissions. It also increases the complexity of the implementation, especially if uplink rather than downlink streaming takes place. Another approach, for example in Cen et al., 2003, is to gather end-to-end statistics in order to distinguish congestion loss from wireless channel loss. Unfortunately, these techniques do not appear to be sufficiently accurate. A promising alternative is to modify TFRC to respond to explicit loss notification (Balakrishnan and Katz, 1998) of channel losses as opposed to congestion losses.

5 Conclusion

This paper has conducted an investigation of multiple DCCP connections for uplink video streaming over a concatenated network, consisting of a WiMAX access network and a fixed network with a bottleneck at the network edge. The study has shown that with static scheduling of the video stream over the connections, increased throughput results. Reducing the video send time reduces the risk from wireless channel error. However, it also implies that reordering at the receiver is required. The resulting start-up delay was about 6 s for the reasonably complex test video. Using a smaller WiMAX frame size can lead to further improvements across the wireless link itself but there is a risk of excessive queuing at the subscriber station devices leading to unacceptable delay when the wired network stage is taken into account. The role of the feedback channel is important, as loss or delay of acknowledgment packets seems to be implicated in the DCCP congestion controllers' slow start, one of the potential causes of buffer underflow. When the scope of the investigation was extended to stream across a further WiMAX link, there was a need for a larger WiMAX frame size to avoid a lengthy sending period but equally too large a frame size on the downlink size can also lead to lengthening streaming periods. Therefore, WiMAX frame size can be critical and its value for video streaming purposes should be a compromise between uplink and downlink behavior. It was also found that to achieve adequate wireless channel utilization over a network path that included two WiMAX links then more connections were needed.

Further investigation will consider the role of acknowledgments and whether a reduction in the acknowledgment rate may improve performance further. It may also be possible to 'warm-up' the DCCP connection handlers by sending non-video data to start with, which could be discarded thereafter. This can reduce the size of the reordering buffer, if such a reduction were required. The main practical issue to be faced is how to arrive in advance at the settings required, both for the WiMAX configuration and the video streaming configuration.

References

Agboma, F., Smy, M., and Liotta, A. (2008) 'QoE analysis of a peer-to-peer television system', *IADIS Int'l Conf. on Telecommunications, Networks and Systems*, pp.111-116.

Andrews, J. G, Ghosh, A., and Muhamed, R. (2007) *Fundamentals of WiMAX*, Prentice Hall, Upper Saddle River, NJ.

Balakrishnan, H., and Katz, R. (1998) 'Explicit loss notification and wireless Web performance', *IEEE GLOBCOM Internet Mini-Conf.*

Balakrishnan, H., Padmanabhan, V., Seshan, S., and Katz, R. (2007) 'A Comparison of Mechanisms for Improving TCP Performance over Wireless Links', *IEEE/ACM Trans. on Networking*, Vol. 5, No. 6, pp.756–769.

Cen, S., Cosman, P., and Voelker, G. (2003) 'End-to-end differentiation of congestion and wireless losses', Vol. 11, No. 5, pp.703-717.

Chen, M., and Zakhor, A. (2004) 'Rate control for streaming video over wireless', *IEEE INFOCOM*, pp.1181-1190.

Chen, M., Zakhor, A. (Aug. 2005) 'Rate control for streaming video over wireless,' *IEEE Wireless Comms.,* Vol. 12, No. 4, August, pp.32–14.

Chen, M. and Zakhor, A. (2006) 'Multiple TFRC connection based rate control for wireless networks', *IEEE Trans. Multimedia*, Vol. 8, No. 5, October, pp.1045–1062.

Choi, S.-H., and Handley, M. (2007) 'Fairer TCP-friendly congestion control protocol for multimedia streaming applications', *Int'l Conf. on Emerging Networking Experiments and Technologies,* article no. 54.

Cicconetti, C., Lenzini, L., Mingozzi, E. and Eklund, C. (2006) 'Quality of service support in IEEE 802.16 networks', *IEEE Network*, Vol. 20, No. 2, pp.50-55.

Fu, Y., Hu, R., Tian, G., and Wang, T. (2006) 'TCP-Friendly Rate Control for streaming service over 3G network', *Int. Conf. on Wireless Comms., Networking and Mobile Computing*, pp.1–4 .

Floyd, S., and Fall, A. (1999) 'Promoting the use of end-to-end congestion control in the Internet', *IEEE/ACM Trans. on Networking*, vol. 7, no. 4, pp.43-56.

Görkemli, B., Sunay, M.O., and Tekalp, A.M. (Oct. 2008) 'Video streaming over wireless DCCP', *IEEE Int'l Conf. on Image Processing*, pp.2028–2031.

Handley, M., Pahdye, J., Floyd, S., and Widmer, J. (2003) 'TCP-Friendly Rate Control (TFRC): Protocol Specification.', RFC 3448.

Haßlinger,G., and Hohlfeld, O. (2008) 'The Gilbert-Elliott model for packet loss in real time services on the Internet', *14th GI/ITG Conf. on Measurement, Modelling, and Evaluation of Computer and Commun. Systs.,* pp.269–28.

Hillestad, O.I., Perkis, A., Genc, V., Murphy, S, and Murphy, J. (2007) 'Delivery of on-demand video services in rural areas via IEEE 802.16 broadband wireless access networks', *Int'l Workshop on Packet Video*, pp.26-35.

Hosein, P. (2008) 'Broadcasting VBR traffic in a WiMAX network', *IEEE Vehicular Technology Conf.*, 5 pages.

Hoyman, C. (2005) 'Analysis and performance evaluation of the OFDM-based metropolitan area network IEEE 802.16', *Computer Networks*, Vol. 49, No. 3, pp.341-363.

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'.

Issa, O., Li, Wei, and Liu, H. (2010) 'Performance evaluation of TV over broadband wireless access networks', *IEEE Trans. Broadcasting,* Vol. 56, No. 2, pp.201-210.

Juan, H.-H., Huang, H.-C., Huang, C.-Y., and Chiang, T. (2007) 'Scalable video streaming over mobile WiMAX', *IEEE Int. Symp. Circ. Syst.*, pp.3463-3466.

Klaue, J., Rathke, B., and Wolisz, A. (2003) 'EvalVid—A framework for video transmission and quality evaluation', *Int'l Conf. on Modeling Techniques and Tools for Computer Performance*, pp.255–272.

Kohler, E., Handley, M. and Floyd, S. (2006). 'Datagram Congestion Control Protocol', *Internet Engineering Task Force*, March, RFC 4340.

Niyato, D., and Hossain, E. (2005) 'Queue-aware uplink bandwidth allocation for polling services in 802.16 broadband wireless networks', *IEEE GLOBECOM*, pp.3702-3706.

Nuaymi, L. (2007) *WiMAX: Technology for Broadband Wireless Access*, Wiley & Sons, New York, NY.

Sadka, A. (2006) *Compressed Video Communications*, Wiley & Sons, Chichester, UK.

Saleemi, M.M. and Lilius, J. (2010) 'Interactive applications of mobile TV', *IEEE Int'l Conf. on Multimedia Computing and Inform. Technol.,* pp.37-40.

Segars, S. (2001) 'Low power design techniques for microprocessors', *Int'l Solid-State Circuits Conf. Tutorial*.

So-In, C., Jain, R., Tamini, A.-K. 'Capacity evaluation of IEEE 802.16e WiMAX', *J. of Computer Systems, Networks, and Communs.*, Vol. 2010 [online], 12 pages.

Stewart, R., ed. (Sept. 2007). 'Stream Control Transmission Protocol', *Internet Engineering Task Force*, RFC 4960.

Yang, F., Zhang, Q., Zhu, W., and Zhang, Y.-Q. (2007) 'End-to-end TCP-friendly streaming protocol and bit allocation for scalable video over wireless Internet', *IEEE J. on Sel. Areas in Commun.*, Vol. 22, No. 4, pp.777-790.

Tappayuthpijam, K., Liebl, G., Stockhammer, T., and Steinbach, T. (2009) 'Adaptive video dtreaming over a mobile network with TCP-Friendly Rate Control', *Int. Conf. on Wireless Comms. and Mobile Computing*, pp.1325–1329.

Tsai, F.C.-D. et al. (2006) 'The design and implementation of WiMAX module for NS-2 simulator', *Workshop on NS2: The IP Network Simulator*, Article no. 5.

Widmer, J., Denda, R., Mauve, M. (May/June 2001). A Survey on TCP-friendly Congestion Control. *IEEE Network*, Vol. 15, No. 3, pp.28–37.

Wiegand, T., Sullivan, G. J., Bjontegaard, G., and Luthra, A. (2003) 'Overview of the H.264/AVC video coding standard', *IEEE Trans. Circuits Syst. Video Technol.,* Vol. 13, No. 7, pp.560–576.

Yogesh, Y., Bose, S., and Kannan, A. (2006) 'Effective TCP Friendly Rate Control scheme for video streaming over wireless networks', *Asian J. of Info. Tech.,* Vol. 5, No. 4, pp.442–447.