Dublin Institute of Technology
ARROW@DIT

Conference papers

Communications Network Research Institute

2006-01-01

Study of the Behaviour of Video Streaming over IEEE 802.11b WLAN Networks

Nicola Cranley
Dublin Institute of Technology, nicola.cranley@dit.ie

Mark Davis
Dublin Institute of Technology, mark.davis@dit.ie

Follow this and additional works at: http://arrow.dit.ie/commcon

Part of the Electrical and Electronics Commons

Recommended Citation

Cranley, N., Davis, M.: Study of the Behaviour of Video Streaming over IEEE 802.11b WLAN Networks. Wireless and Mobile Computing, Networking and Communications, 2006. (WiMob'2006). IEEE International Conference. Page(s): 349 - 355. DOI: 10.1109/WIMOB.2006.1696356.

This Conference Paper is brought to you for free and open access by the Communications Network Research Institute at ARROW@DIT. It has been accepted for inclusion in Conference papers by an authorized administrator of ARROW@DIT. For more information, please contact yvonne.desmond@dit.ie, arrow.admin@dit.ie.

Paper Title:

"Study of the Behaviour of Video Streaming over IEEE 802.11b WLAN Networks"

Authors:

Nicola Cranley, Mark Davis

Contact:

Communications Network Research Institute,
School of Electronic and Communications Engineering,
Dublin Institute of Technology,
FOCAS Institute,
Dublin 8, Ireland

Email:

Nikki.Cranley@CNRI.DIT.ie, Mark.Davis@DIT.ie

Abstract:

The performance of video streaming over WLAN networks is not only influenced by the state of the network but also by the encoding configuration parameters of the video stream, such as the video content being streamed, how the video is encoded and how it is transmitted. In this paper, we analyse the unique delay characteristic of video streaming applications in a WLAN environment. We show that the "burstiness" of video is due to the frame-based nature of encoded video. We show how each video frame is transmitted as a burst of packets that is queued at the Access Point causing the delay to exhibit a sawtooth-like characteristic over time that is related to the frame rate and frame structure of the encoded video. To our knowledge, this sawtooth-like characteristic of video streaming over WLAN has not been previously reported on. In this paper, not only do we consider the end-to-end delay, but more importantly we consider the total delay required to transmit the entire video frame. We present experimental results for VBR and CBR video streams and calculate the upper bounds on video encoding parameters for streaming real-time interactive video over a WLAN.

Keywords:

Video Streaming, IEEE 802.11 WLAN

Study of the Behaviour of Video Streaming over IEEE 802.11b WLAN Networks

Nicola Cranley, Mark Davis
Communications Network Research Institute,
School of Electronic and Communications Engineering,
Dublin Institute of Technology,
FOCAS Institute,
Dublin 8, Ireland

Nikki.Cranley@CNRI.DIT.ie, Mark.Davis@DIT.ie

***Abstract* - The performance of video streaming over WLAN networks is not only influenced by the state of the network but also by the encoding configuration parameters of the video stream, such as the video content being streamed, how the video is encoded and how it is transmitted. In this paper, we analyse the unique delay characteristic of video streaming applications in a WLAN environment. We show that the "burstiness" of video is due to the frame-based nature of encoded video. We show how each video frame is transmitted as a burst of packets that is queued at the Access Point causing the delay to exhibit a sawtooth-like characteristic over time that is related to the frame rate and frame structure of the encoded video. To our knowledge, this sawtooth-like characteristic of video streaming over WLAN has not been previously reported on. In this paper, not only do we consider the end-to-end delay, but more importantly we consider the total delay required to transmit the entire video frame. We present experimental results for VBR and CBR video streams and calculate the upper bounds on video encoding parameters for streaming real-time interactive video over a WLAN.**

***Index Terms* - Video Streaming, IEEE 802.11 WLAN.**

I. Introduction

Streaming multimedia over wireless networks is becoming an increasingly important service [1]. This trend includes the deployment of WLANs that enable users to access various services including those that distribute rich media content anywhere, anytime and from any device. There are many performance-related issues associated with the delivery of time-sensitive multimedia content using current IEEE 802.11 standards. Among the most significant are low delivery rates (e.g. theoretically up to 11Mbps for IEEE 802.11b, but in practice only a maximum throughput of approximately 6Mbps can be achieved), high error rates due to media characteristics, contention between stations for access to the medium, back-off mechanisms, collisions, signal attenuation with distance, signal interference, etc. Under these conditions it is difficult to provide any Quality of Service (QoS) guarantees.

There are a large and diverse number of variables that must be taken into consideration for unicast video streaming, each of which has an impact on the performance and behaviour of the video stream in a WLAN environment. Such variables include the content and complexity of the content, the compression scheme, the encoding configuration, the delivery method of the video, the streaming server used, and also the adaptation algorithm employed by the server.

End-to-end delay is of critical importance in real-time streaming applications. Since if a packet is delayed past its playout time, the packet is effectively lost. In a WLAN environment, lost or corrupted packets are re-transmitted until either the retransmitted packet is successfully ACKed by the receiving STA or until the retransmission counter has timed out. If a packet has expired, it cannot be used by the client application since its contents will be worthless when it arrives. For video streaming applications, not only is the end-to-end delay important, but also the delay incurred transmitting the entire video frame from the sender to the client since the video frame cannot be decoded until all or most of the packets belonging to the video frame are received by the client in time for playout.

Video streaming is often described as "bursty" and this can be attributed to the frame-based nature of video. Video frames are transmitted with a particular frame rate. For example, video with a frame rate of 25fps will result frame being transmitted every 40ms. In general, video frames are large, often exceeding the MTU of the network and results in a several packets being transmitted in a burst for each video frame. The frequency of these bursts corresponds to the frame rate of the video. The video frame cannot be decoded or played out at the client until all or most of the video packets for the particular video frame are received in time. Although error resilient encoded video and systems that include error concealment techniques allow a certain degree of loss tolerance [2], the ability of these schemes to conceal bursty and high loss rates is limited.

In this paper we investigate the behaviour of both VBR and CBR video streaming applications in a WLAN environment and show that video traffic has a sawtooth delay characteristic [3]. Consider, a burst of packets corresponding to a video frame arrive at the AP. The arrival rate of the burst of packets is high and typically these packets are queued consecutively in the AP transmission buffer. For each packet in the queue, the AP must gain access to the medium by deferring to a busy medium and decrementing its MAC back-off counter between packet transmissions. This process occurs for each packet in the queue at the AP causing the delay to vary with a sawtooth-like characteristic. It was found that the duration and height of the sawtooth delay characteristic varies depends on the number of packets in the burst and the packet size. This is expected since when there are more packets in the burst, it

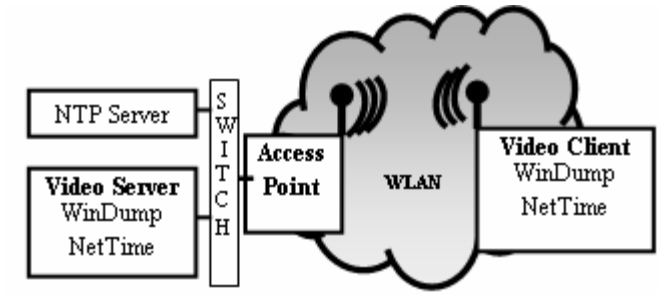

Fig. 1: Experimental Test Bed

takes the AP longer to transmit all packets relating to this video frame.

In this paper, we show that the delay experienced by the video stream is related to the bitrate of the video, the frame rate of the video, the number of packets required to send the video frames, and the packet size. The end-to-end delay and delay to send the entire video frame are considered to be important measures since these delays ultimately manifest themselves as lost data relating to the video stream which in turn affects the end-user perceived quality. Having investigated the behaviour of video over WLAN, we have determined the maximum encoding configuration in terms of video frame rate and frame size that should be used when streaming video over WLAN. This paper is structured as follows. Section 2 provides a description of the experimental test bed. Section 3 presents the results for VBR and CBR video streaming tests. We compare the measurements made with both CBR and VBR video streams and compare these to the ideal minimum delay. We then determine the maximum encoding configuration values for video streaming applications. Finally, we present some conclusions and directions for future work.

II. Experimental Test Bed

To evaluate unicast video streaming, a video server was set up on the wired network and streams video content to wireless clients via the Cisco Aironet 1200 AP (Figure 1) under lightly loaded conditions where there are no other wireless stations contending for access to the medium. Under these conditions, it is possible to isolate and study the behaviour of the video streaming session. In this paper, we have investigated using both VBR and CBR video traffic. For the VBR video analysis, we use the Darwin Streaming Server (DSS) [2] and for the CBR analysis we use a program, RTPTools [3] to mimic the sending behaviour of streamed CBR video. In addition, both the client and server were configured with the packet monitoring tool WinDump [4] and the clocks of both the client and server are synchronised before each test using NetTime [5]. However, in spite of clock synchronisation, there was a noticeable clock skew observed in the delay measurements and this was removed using Paxson's algorithm as described in [6]. The delay is measured here as the difference between the time at which the packet was received at link-layer of the client and the time it was transmitted at the link-layer of the sender.

III. Results

In our analysis, several key measurements have been defined. The Inter-Packet Sending Time (IPST) and Inter-Packet Receiving Time (IPRT) represent the time difference between the current packet and the previous packet at the server and client respectively. The Inter-Packet Delay (IPD) is the difference in measured delay between consecutive packets within a burst at the receiver and gives an indication of the service rate of the AP. Furthermore, the IPD provides a means of analysing the backoff counter values. The IPD remains relatively constant since it takes approximately the same amount of time to send a single packet of a fixed size at the head of the AP queue. The IPD can be measured by monitoring the difference in delay between consecutively queued packets of the same size belonging to the same packet burst. In the next section, we shall demonstrate the delay effects for streaming a single unicast video stream from the wired network via the AP to a wireless client when there is no background traffic.

A. *Analysis of VBR Video*

In the experiments reported here, the streaming server, Darwin Streaming Server (DSS), was used. The DSS streaming server system is a client-server architecture where both client and server consist of the RTP/UDP/IP stack with RTCP/UDP/IP to relay feedback messages between the client and server. The client can be any QuickTime Player or any player that is capable of playing out ISMA compliant MPEG-4. The client connects to the server via RTSP to establish a unicast video streaming session.

The video content was encoded using the commercially available X4Live MPEG-4 encoder from Dicas. This video content, JR, is a 5 minute extract from the film 'Jurassic Park' with a CIF display size and encoded as MPEG-4 SP with a target bit rate of 1Mbps using 2-pass encoding. The encoded video clip is subsequently hinted using MP4Creator from the MPEG4IP [7]. Hint tracks allow a server to stream media files without requiring the server to understand media types, codecs, or packing. Each track in a media file is sent as a separate stream, and the instructions for packetising each stream is contained in a corresponding hint track [8]. Each sample in a hint track tells the server how to optimally packetise a specific amount of media data. The hint track MTU setting means that the packet size will not exceed in the MTU size. Hint track settings are required for streaming MP4 files. However, given that in general most video-frames are quite large and so at most one video frame can be packetised into a single 1024B packet, hint tracks are especially important for audio streaming since multiple audio samples can be packetised into one packet. The mean packet sizes for video with hint track settings of 1024B and 512B are 912B and 468B respectively.

Video streaming applications are extremely variable and this variability in turn affects the end-to-end network delay and bandwidth usage in the WLAN. The bandwidth requirement of the video stream is related to the encoded bitrate of the video and the hint track setting used. Figure 2(a) shows how the encoded video bitrate for the video clip JR

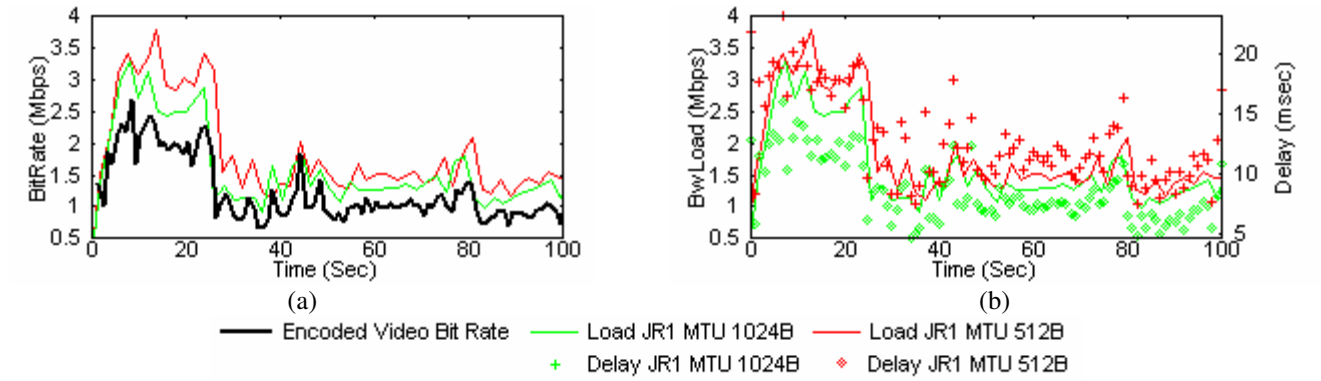

Fig. 2: Relationship between (a) Encoded bit rate and bandwidth load in WLAN (b) Mean delay, bandwidth load and packet size

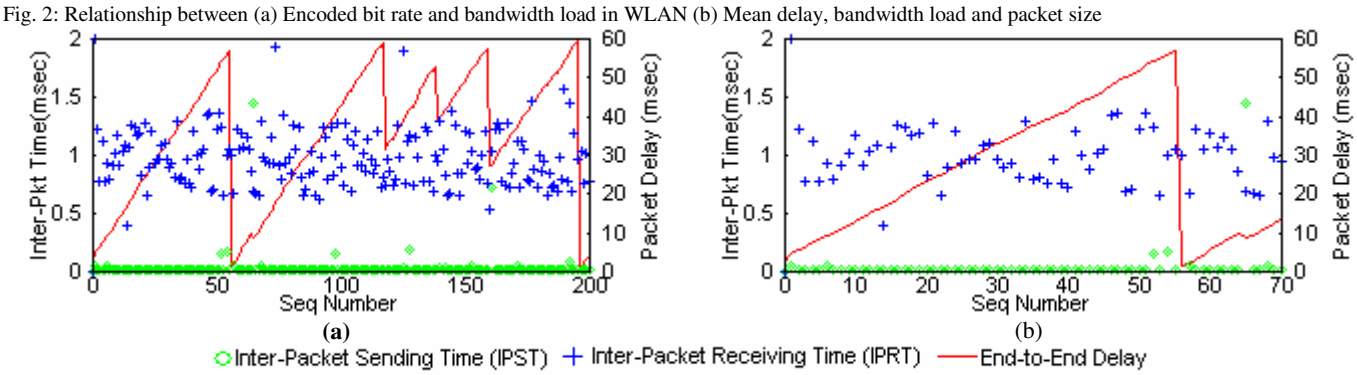

Fig. 3(a) Per-Packet Delay and Inter-Packet Time (b) Close-up view of the first video frame

varies over a period of 100sec. The video clip JR was encoded with a target bitrate of 1Mbps. There is extra overhead incurred transmitting the video stream over the network. This overhead is due to packet header overhead and that for each packet sent, a MAC Acknowledgement packet is sent. It can be seen that when using a hint track with an MTU setting of 1024B, the bandwidth requirement is increased by 16% and when using a hint track with an MTU setting of 512B, the bandwidth requirement is increased by 33% [9, 10]. Furthermore, it can be seen in Figure 2(b) how the mean packet delay is related to the bandwidth requirement and the hint track setting where the mean delay is calculated over all packets during an interval of 1sec. This is to be expected since when the MTU setting is small, more packets need to be sent, thus there are more packets in the queue at the AP waiting to be sent to the client. For each packet in the queue, the AP must gain access to the medium by deferring to a busy medium and decrementing its MAC back-off counter between packet transmissions. In this way, given that more packets need to be sent, the number of packets in the queue at the AP is larger. As a result, the time it takes to send one complete video frame is larger and also the mean packet delay is larger. Similarly, when the bitrate of the video is low, fewer packets need to be sent resulting in a lower mean network delay and lower video frame delay.

By looking more closely at the per packet delay, a sawtooth-like characteristic can be seen clearly. Figure 3(a) shows the delay, the IPST, and IPRT (i.e. the time difference between the current packet and the previous packet) against the sequence number of the packet. It can be seen that the delay varies quite rapidly in a sawtooth manner. It can also be seen that the IPST at the server is very low for packets that belong to the same packet burst or video frame. However, there is a larger IPST at the server between the last packet of the previous burst and the start of the next burst. In contrast, the IPRT at the client is much more varied since there are numerous additional sources of delay along the transmission path such as the network delay on the wired link from the server to the AP, the number of packets in the queue ahead of packet at the AP, the MAC back-off counter value at the AP required to gain access to the medium, and finally the transmission delay which varies with the packet size.

By focusing on a burst of packets as in Figure 3(b), it can be seen that the end-to-end packet delay increases at a steady rate with each packet within the burst and then drops dramatically for the packet with sequence number 32. However, this packet has a large IPST which indicates that this packet is the start of a burst and therefore the start of the next video frame. From this we can say that the arrival rate of the burst of packets corresponding to a particular frame is greater than the service rate at the AP, so the delay accumulates for each packet queued since each packet in the burst must wait until the packets ahead of it in the queue are transmitted. The duration of these accumulations relates to the number of packets required to send the particular video frame. Thus, the larger the size of the frame, the larger the number of packets in the burst, the more the delay will accumulate and the longer it will take to send the complete video frame. Similarly, if a smaller hint track MTU is specified, then there will consequently be more packets in a burst and the delay will climb even higher.

This IPST is for packet sequence number 32 represents a sufficiently large gap in time for the packets from the previous burst to be served from the AP. However, the inter-packet sending time for packet sequence number 64 is much smaller and is not enough for the AP to fully recover from the packet

TABLE 1: VBR DELAY ANALYSIS

Clip	MTU 1024B				MTU 512B			
	Inter-Pkt Delay	Mean Packet Delay (ms)	Mean Frame Delay (ms)	Number Pkts/Burst	Inter-Pkt Delay	Mean Packet Delay (ms)	Mean Frame Delay (ms)	Number Pkts/Burst
JR	1.27	7.7	13.4	10.0	0.96	11.8	19.8	17.7
EL	1.27	8.1	13.7	10.0	0.96	12.3	20.1	17.9
DS	1.27	7.7	13.5	9.8	0.96	11.4	19.4	17.3
DH	1.27	7.4	13.2	9.6	0.96	11.3	19.8	18.4
FM	1.26	6.8	9.7	6.8	0.96	11.2	19.9	18.6

build-up of the previous burst. The IPST can be controlled by encoding the video with a lower frame rate which would decrease the burst rate and increase the gaps between the bursts. This reduces the delay accumulation and allows sufficient time for the AP to send all packets relating to the previous burst before the next burst arrives. Thus, in the short term the arrival rate is greater than the service rate, causing the delay to accumulate within a packet burst. However, in the long term, the service rate is greater than the arrival rate since the mean delay remains relatively low.

The Inter-Packet Delay (IPD) is the difference in delay between consecutive packets within a burst and represents a direct measure of the service rate of the AP. Figure 4 shows the PDF of the IPD for the video JR streamed with a hint track setting of 512B and 1024B where the y-axis is a log-scale of the percentage frequency of the IPD value. It can be seen that the peak occurs between 0.6ms and 1.3ms for video with a hint track MTU of 512B. During this peak, there are 32 spikes which correspond to the randomly chosen Backoff Counter values of the 802.11 MAC mechanisms contention windows. The tail of this distribution is related to retransmissions.

Table 1 presents a summary of the results relating to the measured delays for streaming the video clip JR with a hint track MTU setting of 512B and 1024B. The results show that it takes on average 0.21ms longer to send a video packet with an IP packet size of 1024B than a packet of size 512B. As expected, it was found that by using a smaller hint track setting, there were more packets in a burst or video frame resulting in on average 17.7 packets when using a hint track setting of 512B and 10 packets for a hint track setting of 1024B. As a consequence, the mean delay required to send the complete video frame was 19.77ms and 13.42ms respectively and the mean packet delay in a burst was 7.69ms and 11.82ms. Although, the mean frame delay is in the range 13-20ms, very large video frames have a significantly larger frame delay. This is evident in Figure 3(a) where it took approximately 60ms to send 5 video frames. The experiments were repeated for a number of different video clips with the same encoding configuration and similar results were found for the inter-packet delay. However, the mean packet delay, mean frame delay, and number of packets per burst were found to be different since they relate to the bit rate of the encoded video content.

B. *Analysis of CBR Video*

Given that there is a large number of encoding parameters that can be varied whilst preparing the video content for streaming over the network, in this section we will focus only

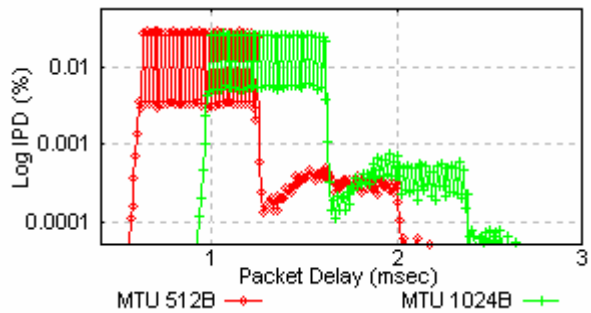

Fig.4: PDF of the IPD for video at 512B and 1024B

on how varying the inter-frame gap (i.e. the frame rate of the video) and the burst size (i.e. the size of the video frame) affects the mean packet delay, the IPD, and the delay required to send a complete video frame. In order to control these parameters, a sending script was generated that emulates the sending behaviour of the video streaming server as observed in VBR video but enforces the desired burst gaps and burst sizes. This sending script was then used by the RTPSender program to generate the exact video traffic stream. Figure 5 shows how the frame rate (or the time between each burst) was increased every 300sec. In addition, the burst sizes (i.e. the size of the video frame) was increased every 100sec. By varying both of these parameters, the bitrate behaves in an Additive Increase, Proportional Decrease (AIPD) manner. In these experiments several different hint track MTU sizes were investigated. However, in order to maintain the bit rate for each test case, the number of packets in a burst was varied. For example, when using a hint track MTU setting of 512B the burst size was varied in a stepwise manner in the set of {6, 12, 18} packets/burst and when using a hint track setting of 1024B the bursts size was in the set of {3, 6, 9} packets/burst.

Figure 6 shows how the delay varies over the entire test period for each packet sent. It can be seen that as the burst size is increased over a period of 300sec, the delay is steadily increased. However, as the frame rate is increased, there is very little increase in the delay. Figure 7 shows how the delay varies for 6 video frames towards the end of the test using different packet MTU sizes. At this time the video has a bitrate of approximately 2.2Mbps, a frame rate of 30fps, and a burst size that is related to the hint track MTU setting. It can be clearly seen that when using a hint track MTU setting of 512B, the delay is much greater to send one complete frame despite the fact that the inter-packet delay is less. However, when using a larger hint track MTU setting, it takes more time to send each individual packet but since there are fewer packets in the video frame, overall it takes less time to send

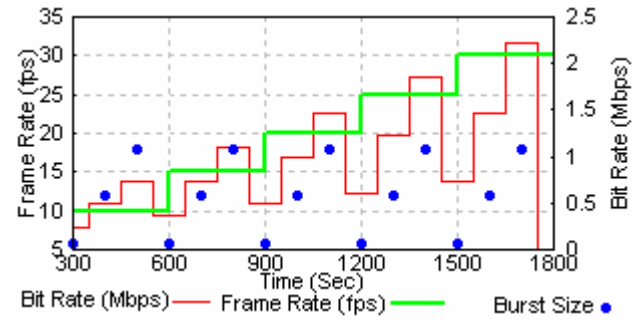

Fig. 5: Offered CBR traffic

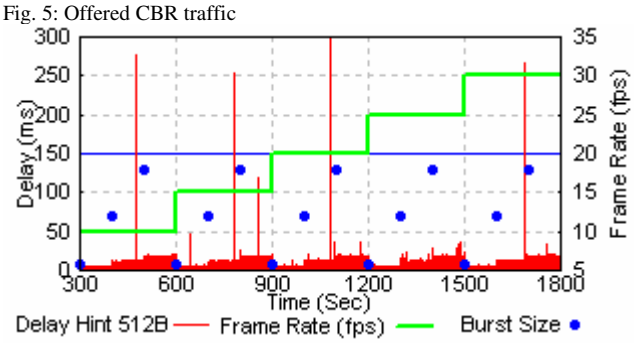

Fig. 6: Delay Variations over time for video at 512B

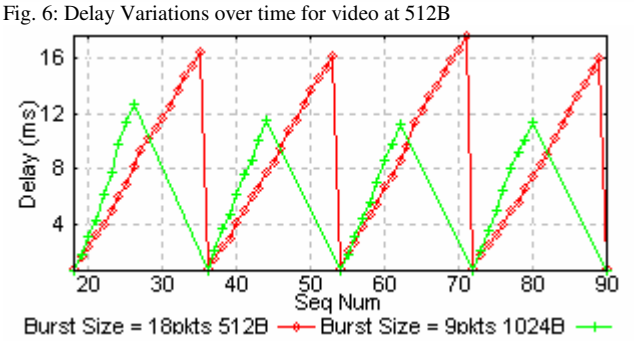

Fig. 7: Delay accumulation for 4 video frames of equal size using a packet size of 512B and 1024B

the complete video frame. The IPD was recorded over the entire test period and was calculated to be 0.96ms for 512B sized packets and 1.34ms for 1024B sized packets.

These measurements are very close to the expected IPD since this delay includes the time the AP spends accessing the medium, including DIFS, Backoff (i.e. *TimeSlot*BC*, where *BC* is in the range 0 to 31), data transmission, SIFS, and the time to receive the MAC Acknowledgement [11] which results in a service time in the range of (0.76ms, 1.38ms) with a mean of 1.07ms for a single 512B packet and in the range of (1.1ms, 1.75ms) for a 1024B packet. In addition, these values are very close to the IPD measured for the VBR video traffic, where the IPD with a hint track of 512B was 0.96ms and 1.27ms with a hint track of 1024B. There is a slight difference with the IPD measured for CBR at 1024B packets which was measured as 1.34ms. The reason for this difference is that for VBR traffic, not all packets were exactly 1024B as when the video frame is packetised, this results in several maximum MTU-sized packets and a fragment packet that contains the remainder of the video frame data which is less than the MTU size since it is unlikely that the size of the video frame would be exactly *n*MTU*, where *n* is the number of packets. We have found the mean packet size for VBR video with a hint track setting of 1024B and 512B is 912B and 468B respectively. This explains why the IPD for the VBR streams when using larger packet sizes is slightly less than that observed using CBR traffic since when using a smaller MTU setting, the proportion of packets that are less than the MTU is less.

To summarise the results, we found that regardless of the burst size and video frame rate, the IPD remains relatively constant for a particular hint track MTU setting. The mean frame delay increases dramatically with burst size since there are more packets to be sent. However, by using a smaller MTU setting, it takes much longer to send a complete video frame despite the fact that it takes less time to send a smaller packet. Furthermore, we found that the mean packet delay is greater since there are on average more packets to be served in the queue ahead of it.

C. Maximum Encoding Configuration

Using these measurements, the maximum bounds for video streaming applications can be inferred including the maximum possible throughput and the maximum packets per video frame. We have found that the inter-packet delay or rather the time it takes the AP to send a packet is 0.96ms for 512B sized packets. Thus, the maximum throughput at the AP is 1041 packets per second which is equivalent to a bit rate of 4.26Mbps. Similarly, when using 1024B packets, the mean delay is 1.34ms which results in a maximum of 746 packets per second to be served per second and is equivalent to a bit rate of 6.11Mbps. For real-time applications, there are strict delay bounds imposed on packet delivery. For real-time interactive traffic, the delay constraint is 150ms. Using this delay bound, we can infer the maximum number of packets that can be sent before packets are lost due to exceeding this delay bound. Using 512B sized packets, the maximum number of packets in a burst before the delay exceeds 150ms is $\left(150/0.96\right)$ which equals 156 packets. Similarly a maximum of 111 packets can be sent in a burst when using a packet size of 1024B. Finally, we have observed that if video frames arrive too quickly (i.e. the inter-burst gap is too small), then the delay for the frames increases since not all the packets for the previous frame have been cleared from the queue at the AP before the next frame arrives. For example, if the video has a frame rate of 10fps, then a new video frame is created and transmitted every 100ms. This requires that all packets pertaining to this video frame should be sent within 100ms before the next frame is sent. When using 512B packets, then given that there is 100ms before the next frame to arrives, given that it takes 0.96ms to send each packet, then no more than 104packets (i.e. $100/0.96$) should be in a video frame, which implies that the maximum size of the video frame should not exceed 426kb or 53.33kB. This calculation has been performed for each frame rate and presented in Table 2. (Predicated on a best case scenario where there is no competition for resources from other stations).

Figure 8(a) shows a PDF of the I and P-Frame sizes for the video clip JR encoded with a target VBR bitrate of 2Mbps and a frame rate of 25fps. It can be seen that there are a number of I-Frames that exceed the maximum encoding configuration as shown in Table 2. When the frame size exceeds this

TABLE 2: MAXIMUM VALUES FOR VIDEO STREAMING APPLICATIONS

Frame Rate (fps)	Frame Interval (ms)	Hint Track Setting 512B		Hint Track Setting 1024B	
		Maximum Pkts/Frame	Maximum Avg. Frame Size (kB)	Maximum Pkts/Frame	Maximum Avg. Frame Size (kB)
10	100	104	53	74	76
15	66.66	69	36	49	51
20	50	52	27	37	38
25	40	41	21	29	31
30	33.33	34	18	24	25

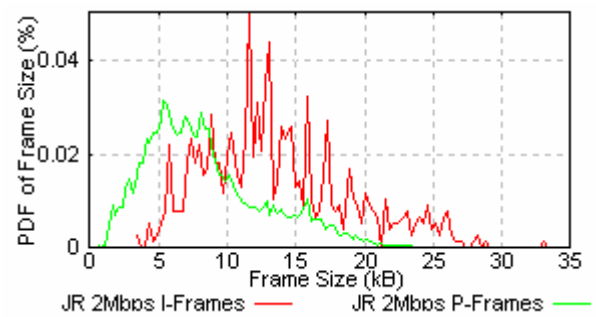

Fig. 8(a): PDF of I and P Frame Sizes

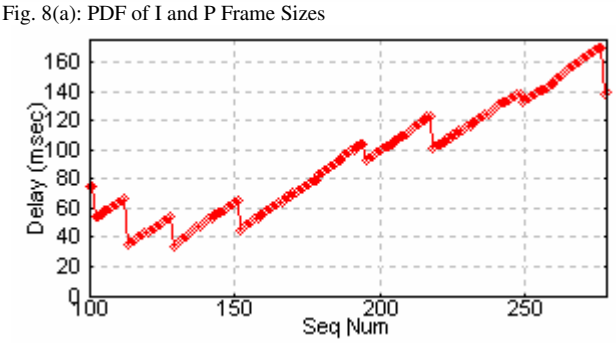

Fig. 8(b): PDF of I and P Frame Sizes for Video Clip JR encoded at 25fps with a target VBR bitrate of 2Mbps

maximum value, the AP does not have sufficient time to transmit all the packets belonging to a video in its transmission queue before the next frame arrives. Thus, the first packet of the next video frame must wait until all packets in the queue ahead of it have been transmitted. This causes the delay to accumulate. For example, Figure 8(b) shows several video frames streamed with a hint track MTU of 1024B. The video frame with the sequence numbers 151-193 contains over 40 packets. This exceeds the maximum encoding configuration for video with 25fps and streamed with 1024B packets. As a result, the AP does not have sufficient time to transmit all packets relating to this video frame before the next frame arrives. The first packet of the newly arrived video frame must wait 50ms whilst the remaining packet belonging to the previous frame are transmitted. This causes the delay to gradually accumulate. It is clear, that by ensuring that the video frames are below the maximum video encoding frame size, the possibility of this delay buildup can be reduced.

IV. CONCLUSIONS

In this paper, we have experimentally investigated the behaviour of video streaming applications. We began by showing the correlation between the bitrate fluctuations of the encoded video stream and the bandwidth usage of the WLAN which depends on the packetisation scheme used for the video. By using small packets not only is there an increased header overhead due to the fact that more packets are required to send the same amount of data, but also more MAC layer ACKs need to be sent. In addition, by using small packets the AP must access the medium more often.

We have demonstrated the primary challenge posed by streaming video over WLAN networks by analysing both VBR and CBR video. Video is a frame-based media, whereby frames are generated at a particular rate. In general, several packets are required to transmit a video frame to the client. On the network, this appears as periodic bursts of packets. Each burst of packets is queued at the AP. The AP must gain access to the medium to send each packet. Since each packet must wait for the packets in the queue ahead of it to be transmitted, the end-to-end delay steadily increases until all packets in the burst have been transmitted. The rate at which the delay increases depends on the size of the packet to be transmitted since the AP can send a smaller packet faster than a large one. However, by using a smaller MTU for the video stream the packet bursts are much larger. The gap between the packet bursts (i.e. consecutive video frames) allows the remaining packets in the queue to be transmitted before the next burst of packets arrives. This results in the end-to-end delay for consecutive packets relating to the same video frame to rise and fall in a saw-tooth manner. If however the AP cannot clear the packets from the previous burst before the next burst arrives, then the delay is increased by an offset that is proportional to the number of packets from the previous burst still awaiting transmission. To our knowledge, this sawtooth characteristic has not been reported on for video streaming over WLAN. Using this understanding of the behaviour of video streaming over WLAN, we determined the maximum encoding configuration values for real-time interactive video streaming applications, including the maximum throughput and the maximum video frame size for a given frame rate. By conforming to the maximum encoding configuration, the possibility of a delay buildup can be reduced.

The results reported here represent an ideal situation where there are no other stations contending for access to the medium nor is there any other traffic interleaved with the video packets queued at the AP. Further analysis is being conducted in order to establish how contention with other wireless stations with varying traffic loads and packet characteristics affects the ability of the AP to empty the queue of video packets.

ACKNOWLEDGEMENT

The support of the Science Foundation Ireland, grant 03/IN3/1396, under the National Development Plan is gratefully acknowledged.

REFERENCES

[1] J. Wexler, S. Taylor, "2004 Wireless LAN State of the Market Report", *Webtorials*, Feb. 2004, [Online]. Available: http://www.webtorials.com/ main/resource/papers/taylor/paper4/2004-WLAN.pdf

[2] Y. Wang, S. Wengers, J. Wen, A.K. Katsaggelos, "Error resilient video coding techniques", IEEE Signal Processing Mag., vol. 17, no. 4, pp. 61-82, July 2000

[3] N. Cranley, M. Davis, "Delay Analysis of Video Streaming over IEEE 802.11b WLAN Networks", submitted to IEEE Electronic Letters, 5th November 2005

[2] Darwin Streaming Server, http://developer.apple.com/darwin/projects/streaming/

[3] RTPTools, http://www.cs.columbia.edu/IRT/software/rtptools/

[4] WinDump, http://windump.polito.it/

[5] NetTime, http://nettime.sourceforge.net/

[6] S. B. Moon, P. Skelly, D. Towsley, "Estimation and Removal of Clock Skew from Network Delay Measurements", in Proc. of IEEE InfoComm'99, March 1999

[7] MPEG4IP, http://mpeg4ip.sourceforge.net/index.php

[8] "Hint Track Format", http://developer.apple.com/documentation/QuickTime/REF/Streaming.29.htm#pgfId=19901

[9] N. Cranley, M. Davis, "Performance Evaluation of Resource Usage for Unicast Video Streaming over IEEE 802.11 WLAN Networks", 5th Workshop on Applications and Services in Wireless Networks ASWN 2005, Paris, France, July 2005

[10] N. Cranley, M. Davis, "Performance Analysis of Network-level QoS with Encoding Configurations for Unicast Video Streaming over IEEE 802.11 WLAN Networks", WirelessCom 2005, Maui, Hawaii, June 2005

[11] J. Jun, P. Peddabachagari, M. Sichitiu, "Theoretical Maximum Throughput of IEEE 802.11 and its Applications", in Proceedings of the Second IEEE International Symposium on Network Computing and Applications, Washington, DC, USA, 2003