

# Multimedia Application Performance on a WiMAX Network

Emir Halepovic, Majid Ghaderi, Carey Williamson  
Department of Computer Science, University of Calgary, Canada  
{emirh, mghaderi, carey}@cpsc.ucalgary.ca

## ABSTRACT

In this paper, we use experimental measurements to study the performance of multimedia applications over a commercial IEEE 802.16 WiMAX network. Voice-over-IP (VoIP) and video streaming (RealPlayer) applications are tested. The WiMAX-based network solidly supports VoIP, providing adequate quality for short to medium duration calls. The voice quality degradation compared to high-speed Ethernet is only moderate, despite higher packet loss and network delays. The effects of the uplink and the downlink on call quality are comparable, despite their different characteristics. On-demand video streaming performs well using UDP. Smooth playback of high-quality video/audio clips at aggregate rates exceeding 700 Kbps is achieved about 63% of the time, with low-quality playback periods observed only 7% of the time. Our results show that WiMAX networks can adequately support currently popular multimedia Internet applications.

**Keywords:** Measurement, Performance, 802.16, WiMAX, Multimedia, VoIP, Video streaming

## 1. INTRODUCTION

Internet access has undergone a fundamental change in recent years. Stationary wired access is becoming a thing of the past, with low-cost radio technologies and more powerful wireless devices driving a transition to fully mobile Internet access. For applications and services, users are expanding their demand from Web browsing and email to multimedia services, including Voice-over-IP (VoIP) and media streaming.

Broadband Wireless Access (BWA) promises Megabit (or faster) Internet access anywhere and any time. BWA relies on a combination of technologies, including cellular telephony, as well as recent wireless standards such as IEEE 802.11n, 802.16e, and 802.20. However, to satisfy the user demand that accompanies new technologies, improvements of the current radio access systems are needed, to enable the transition to all-IP networking for both voice and data services.

One exciting new technology that promises wide-area high-speed Internet access is WiMAX (Worldwide Interoperability for Microwave Access). Based on the IEEE 802.16 family of standards, WiMAX is currently being deployed primarily as an alternative to cable and DSL service. Two Canadian service providers are currently offering BWA services, based on fixed-location version of 802.16 standards.

In this paper, we report empirical measurement results from a commercial WiMAX-based network. We focus on the performance of two popular multimedia applications: Skype for VoIP [1] and RealPlayer for video streaming [2]. Contrary to recent reports in the literature [3], we find that VoIP performance, while not perfect, is adequate. We also observe that the WiMAX-based network consistently supports high-quality video streaming over UDP. Our experiments indicate that the network can sustain a smooth playback of a 700 Kbps video and audio stream 63% of the time, with low quality video observed only 7% of the time.

The remainder of this paper is organized as follows. Section 2 presents background on WiMAX and related work. Section 3 describes our experimental methodology. Section 4 presents the VoIP results, while Section 5 presents the results for video streaming. Section 6 concludes the paper.

## 2. BACKGROUND AND RELATED WORK

### 2.1 WiMAX

IEEE 802.16 is a family of standards for local and metropolitan area networks, whose fixed and mobile versions have recently been consolidated as 802.16e-2005 [4]. The standards define the Physical (PHY) and Medium Access Control (MAC) layers of the air interface.

The PHY layer uses adaptive modulation based on Orthogonal Frequency Division Multiplexing (OFDM). Adaptive modulation is used to achieve the highest possible data rate for a given link quality. Modulation can be adjusted at very short time intervals (e.g., 5 ms), to provide robust transmission links and high system capacity. The OFDM mechanism uses many individual carriers to transmit user data, and effectively adapts to multi-path fading in the wireless environment. Depending on the frequency range and modulation used, WiMAX can theoretically achieve a data rate of over 120 Mbps, and up to 50 km in range. However, feasible data rates for fixed WiMAX are around 18 Mbps using 5 MHz channel bandwidth in the 2-11 GHz frequency range [5].

The WiMAX MAC layer supports point-to-multipoint (PMP) and mesh topologies, both of which rely upon a shared access medium. In PMP topology, a WiMAX network is divided into cells and sectors consisting of one base station (BS) and many subscriber stations (SS), similar to a cellular telephone network. This architecture naturally lends itself to PMP operation in the downlink direction, from BS to SS, where time-division duplex (TDD) or frequency-division duplex (FDD) is used. In practice, TDD is typically used, where BS dynamically adjusts the duration of the downlink and uplink portions of the data frame, depending on the requirements. Uplink access is usually TDMA, with scheduling fully controlled by the BS.

The MAC layer is connection-oriented and unidirectional. All service flows are mapped to connections between BS and SS. For example, one TCP connection would be mapped to two MAC connections, one for each direction.

While often seen as an evolutionary extension of WiFi, WiMAX has several important differences. Wireless channel access is controlled by the BS in PMP mode, in contrast to WiFi where the access point contends with mobile nodes for channel access. WiMAX is intended for infrastructure deployment as a long-range access technology rather than for short-range home and office networking. Licensed spectrum is predominantly used for WiMAX, and Quality of Service (QoS) is explicitly supported, unlike in WiFi.

The 802.16 standard explicitly supports QoS differentiation. Subscriber stations are assigned access slots on a demand basis. An SS may request access continuously or upon user request, depending on the class of service. A QoS architecture needs to be in place to facilitate communication between multimedia, file transfer or interactive applications, which have different requirements in terms of bandwidth, delay, and jitter. Data packets are mapped onto MAC connections and service flows that are associated with a particular QoS level.

To support different priorities of service flows, the 802.16 standard specifies four traffic classes [4, 6]:

- Unsolicited Grant Service (UGS) supports real-time constant bit-rate (CBR) data flows. Fixed size data packets are accepted by the BS at periodic intervals, with firm guarantees for bandwidth and access delay. UGS is suitable for applications that require a constant bandwidth and limited delay variation, such as Skype VoIP without silence suppression.
- Real-Time Polling Service (rtPS) also supports real-time applications, but with variable bit-rate (VBR) and less stringent delay/jitter requirements (e.g., video conferencing, video streaming, VoIP with silence suppression). The BS provides transmission opportunities to each SS periodically via a basic polling mechanism.
- Non-Real-Time Polling Service (nrtPS) is intended for use by non-real-time applications requiring better than best effort service in terms of bandwidth, but that are not delay sensitive. Examples include file transfer or database applications.
- Best Effort (BE) service is for best effort applications with elastic traffic, such as email, Web browsing, and telnet. No guarantees in terms of bandwidth, delay, or request access are offered by the BS. This service has the lowest priority.

The standard does not specify how QoS policies are implemented. Rather, this is left to the vendors. The scheduling algorithm is also left unspecified for non-UGS flows.

## **2.2 Related work**

Relatively few experimental results are available for the performance of multimedia applications on WiMAX networks, because of the limited deployment of WiMAX and the proprietary nature of these deployments.

Many scheduling algorithms for WiMAX networks have been evaluated using analysis and simulation [5, 6]. Simulation has also been used to evaluate performance of TCP over WiMAX links [7, 8]. The reported simulation results indicate

that improved TCP performance would result from different modulation schemes between data and acknowledgment channels, proper link-layer retransmission settings, as well as a TCP-aware MAC scheduling policy.

Ciconetti et al. [6] present simulation results for a mix of best effort and multimedia traffic. Their results indicate that the average delay for the TCP best effort traffic grows more sharply on the uplink than on the downlink, because of the bandwidth-request mechanism and signaling overhead. For multimedia traffic in the rtPS traffic class, delay and delay variation are stable until the SS population saturates.

Pellegrini et al. [5] discuss WiMAX support for VoIP using results from an experimental testbed. Using a computed index of voice quality, the authors report that the SS capacity for high-quality voice calls ranges between 10 and 17, depending on the codec used. Important findings indicate that the downlink was the bottleneck, and that the voice quality was better on the uplink than on the downlink, contradicting the results in [6]. Although the scheduling policy of the BS was not disclosed to the authors, they used sample delay distributions to infer that the BS used a strict threshold on the bit rate accepted from the SS, penalizing all flows if the threshold was exceeded.

Our previous work included a comprehensive study of the TCP performance on a commercial WiMAX network [9]. We evaluated downlink and uplink performance of TCP flows at residential and campus locations and studied the effects of TCP variants, socket buffer size, TCP window auto-tuning, and traffic directionality. Our observations for TCP flows include high RTT caused by transcontinental wired hops, higher RTT on the uplink than on the downlink, high RTT variability caused by the wireless hop, and the dominance of congestion loss over wireless transmission losses. Some of these results confirm earlier findings by Perez et al [10], who studied TCP downlink performance on a WiMAX-based access network in Belgium. Their main results include a high average RTT near 200 ms, a high loss rate around 6% with frequent occurrence of bursty losses, and high correlation between packet losses and delay, indicating buffer overflows.

A study of Skype traffic in 3G UMTS network uses a testbed and a live network to investigate whether Skype can perform well in UMTS environment [11]. The findings show that Skype calls are possible but that voice quality is not as good as on the testbed. The older version of Skype used (1.20) generates packets at a constant rate of about 26 Kbps and does not adapt the sending rate based on packet loss.

While video streaming results from live WiMAX networks are lacking, other wireless technologies have been experimentally evaluated for video streaming support. The IEEE 802.11 wireless LAN technology has been well explored by measurement studies involving different types of traffic [12, 13]. It has been shown that the streaming quality is robust in a variety of channel conditions, and that both the link-layer and application-layer loss recovery mechanisms contribute to smooth playback and allow for good user experience [12]. Similar results come from the study of streaming performance in a 1xEV-DO cellular network [14].

### **3. EXPERIMENTAL METHODOLOGY**

#### **3.1 Network environment**

Our experiments are conducted using a commercially offered fixed WiMAX service deployed across Canada by two network providers. The service provider limits the maximum data rates to 1.5 Mbps on the downlink and 256 Kbps on the uplink. Nomadic movement between base stations is fully supported, while mobility during a session is not.

The indoor wireless modem used is Motorola Expedience RSU-2510F, operating inside the licensed 2496-2690 MHz band. Expedience technology uses TDD/OFDM combination with 4/16/64 QAM modulations and 3-6 MHz channels. The modem connects via Ethernet to the user computer. The MAC-layer protocol and scheduling policy are proprietary. Therefore, we treat the wireless modem as a black box.

The experimental testbed consists of two commodity laptops, with one connected to the wireless modem and another to the University of Calgary campus network using a 100 Mbps Ethernet LAN (Fig. 1). Both laptops are running Windows XP SP2. Further details on experimental setup are stated with the corresponding results in the following sections.

#### **3.2 Voice quality assessment**

The traditional method for voice quality assessment is the Mean Opinion Score (MOS), based on a subjective evaluation by a human listener [15]. The result is expressed as a number between 1 and 5, with 5 representing the highest perceived quality.

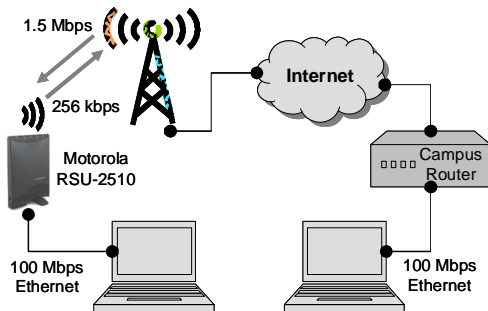


Fig. 1. Experimental setup.

Table 1. Mapping between R-factor, quality rating, and MOS.

R-factor	Quality rating	MOS
$90 < R \leq 100$	Best	4.34 – 4.5
$80 < R \leq 90$	High	4.03 – 4.34
$70 < R \leq 80$	Medium	3.60 – 4.03
$60 < R \leq 70$	Low	3.10 – 3.60
$50 < R \leq 60$	Poor	2.58 – 3.10

To avoid issues associated with subjective evaluation, researchers have automated the computation of MOS by directly comparing the original and degraded signal for a recorded audio stream. This method, recommended by the International Telecommunication Union (ITU) in a ITU-T P.862 document, is called Perceptual Evaluation of Speech Quality (PESQ) [11]. Input to the PESQ algorithm are the audio files containing the sent and received voice. The output is a value between -0.5 and 4.5, indicating the worst and best quality, respectively.

Other researchers have proposed an objective model that relates network performance metrics and voice quality, obviating the need for subjective assessment and voice recordings. The E-model takes into account several parameters known to affect the quality of voice conversation [5, 15]. Using computational method, E-model's outcome is the R-factor, a number between 0 and 100, which can be mapped to MOS, as shown in Table 1. Some typical R-factor values include 82 for traditional phone calls and 68 for VoIP [5]. E-model is based on the assumption that impairments to the voice signal are additive. The main advantage of E-model is its simple and objective computation; for a given codec and its impairment factor, only delay and loss are needed for quality estimation. The model can be further simplified so that online monitoring and quality estimation are feasible [16].

An alternative metric for voice quality assessment, called User Satisfaction Index (USI), has recently been proposed [1]. USI is strongly based on call duration of VoIP calls, under the assumption that call duration is indicative of call quality (i.e., the better the quality, the longer the conversation). USI is calculated based on the source bit rate, the standard deviation of the received bit rate, and the end-to-end round-trip time (RTT). While theoretically unbounded, USI in practice lies between 4 and 10. Interestingly, the impact of RTT is given very little weight in the formulation of USI.

### 3.3 Video streaming quality assessment

At this time, there is no standardized method for video quality assessment. For on-demand video streaming, bandwidth and jitter play important roles in the quality of video seen by the user. Video clip consists of video frames that should arrive at the client at a rate adequate for smooth playback, usually 25 or 30 frames per second (FPS). Absolute network delay is not very important, because users tolerate some startup delay. Bandwidth is the main factor influencing whether a video of the desired quality, i.e. bit rate, can be played by the client. The proportion of time a user can watch the video playback at its intended bit rate is an indicator of video quality used in this study.

Inter-frame jitter is easily perceptible by the user. As video frames are transmitted in packets over the network, lost or delayed packets may cause frame drops at the client. Packets may be recovered by the application if there is enough time before the frame is due for playback. Therefore, variation in packet arrival time (inter-packet jitter) is not as important as inter-frame jitter (referred to as simply jitter in the context of video streaming). Low jitter is a requirement for a smooth streaming experience. Wang et al. [17] propose a simple quality classification based on jitter, wherein *smooth* video has jitter less than 50 ms, *rough* video has jitter exceeding 300 ms, and *average* video has jitter between these two thresholds. We adopt this evaluation method in our study.

## 4. VoIP PERFORMANCE

We use Skype for our VoIP experiments [18]. Skype is arguably the most dominant VoIP application, characterized by its peer-to-peer nature and proprietary protocol and packet structure. We use the latest version of Skype (3.8) available at the time of experiments. Skype has significantly evolved over several years, especially in terms of voice codec used. The latest codec, Sinusoidal Voice Over Packet Coder (SVOPC), was introduced in version 3.2 [19]. SVOPC is intended to

offer good voice quality between two computers at around 20 Kbps, and especially under conditions of higher frame loss (5% to 30%), which makes it highly suitable for wireless channels.

We focus on two aspects of Skype performance over WiMAX. First, we evaluate the voice quality using PESQ MOS, and relate it to the network conditions in terms of delay, loss, and jitter, as reported by Skype during the call. We also calculate the USI and compare it to MOS results. Second, we show the impact of the WiMAX-based network by comparing the packet generation and inter-arrival times from collected traces, and extract the actual packet rate and bandwidth used by Skype 3.8. We do not analyze packet delay during Skype sessions for the following reasons: it has been shown that delay (or RTT) has minor effect on Skype call quality when within reasonable bounds [1]; it is difficult to match sent and received packets within a Skype call, due to encrypted payload.

Our experiments are set up such that the two laptops are running Skype and the trace collection tool. One unidirectional Skype call is active at a time to emulate a single user's experience. The sample audio clip is the standard test file for evaluating VoIP and contains male English-spoken text with 8 kHz sample rate and 16-bit encoding [20]. The sample clip is played through one Skype client and received by the other. Laptops are connected to the network and Internet as shown in Fig. 1. The third computer is added to record the voice signal from the Skype receiver via an audio cable, as shown in Fig. 2. The clip playback through Skype is facilitated by changing the audio input from microphone to audio mixer in Windows XP on the sender laptop. Recording cannot be performed on the receiver by microphone redirection because that would introduce a feedback loop into the Skype session, hence the need for the third computer. We experimented with some of the available Skype recording software, but were not satisfied with their ease of use and recording quality. We consistently used the same machine for sending, receiving and recording audio.

Skype measures and reports network conditions including packet loss, RTT, jitter, bandwidth used, and others, but we concentrate on these as the most relevant. All of reported metrics except jitter are defined as expected. While there is no publicly available definition of jitter as computed by Skype, we empirically found that jitter value is in the range of 20 - 40 on 100 Mbps Ethernet, and generally between 70 and 200 on WiMAX network. Lower values of jitter correspond to better voice quality. Average bandwidth used as reported by Skype is too coarse-grained for our purposes, so we derive average values from traffic traces.

#### **4.1 Subjective quality assessment**

Our subjective impressions from several Skype sessions are that the voice quality is moderate, but acceptable. There was no difficulty in understanding the other party when talking normally. However, most of the time when one person starts talking concurrently with the other one, neither could understand the other party. The voice heard over the uplink channel was noticeably delayed with respect to the original voice overheard across the hallway, up to a couple of seconds. The overall experience was tolerable, although sometimes unacceptable.

#### **4.2 Voice quality degradation due to equipment and codec**

To establish the baseline MOS without the network impact, we played the audio file on one of the laptops and recorded it on another computer using audio cable only. This way we measured the signal degradation due to equipment involved, in particular sound cards and playback and recording software. This baseline MOS is 3.20, with standard deviation of 0.10, based on nine measurements. While this MOS seems low, we note that better results are achievable after advanced adjustment of audio parameters. However, we believe that the average users are unlikely to even attempt that, so we resort to more practical and realistic approach. Using Skype's auto-tuning of sound settings, we simply obtain the stable limits of microphone and speaker levels, and fix them during experiments. It is still noteworthy that the MOS obtained from the live experiments should be interpreted in relative terms of a reduction factor, rather than as an absolute value.

Next, we seek to determine the impact of the SVOPC codec so that we can properly evaluate the impact of the WiMAX network. To accomplish this goal, we run five live experiments over 100 Mbps campus Ethernet. It turns out that these are near-perfect conditions as seen by Skype, according to its reported 0% loss, 0 ms RTT, and very low jitter of 26. Therefore, we expect that the only signal degradation would be due to codec. The average MOS based on five experiments is 3.18, with standard deviation of 0.11, indicating a negligible degradation.

#### **4.3 Performance over WiMAX network**

We conducted the total of 20 experiments on the downlink and 20 on the uplink involving Skype 3.2 and 3.8, and tested two different audio samples, 51 and 16 seconds in duration. While the results across Skype versions and audio samples

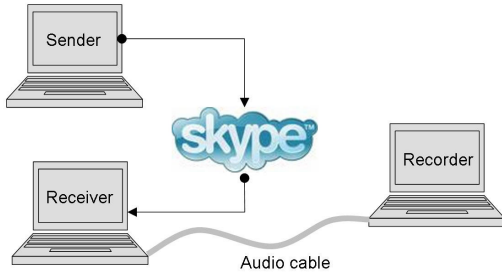


Fig. 2. Experimental setup for recording audio signal transmitted by Skype.

Table 2. Measurements reported by Skype and PESQ MOS shown as mean values with standard deviation in parentheses.

	Ethernet	WiMAX Downlink	WiMAX Uplink
<b>Jitter</b>	26 (8)	129 (30)	130 (51)
<b>Loss (%)</b>	0 (0)	0.82 (0.63)	3.68 (0.97)
<b>RTT (ms)</b>	0 (0)	192 (28)	137 (26)
<b>MOS</b>	3.18 (0.11)	2.49 (0.12)	2.47 (0.26)

are nearly the same, we present the results from the combination of Skype version 3.8 and shorter 16-second audio sample. The summary of the MOS results and network metrics obtained from Skype are shown in Table 2.

Our primary interest is in the MOS as compared to the baseline of 3.20, indicating the compound effects of the codec and the network. The average MOS for the downlink of 2.49 indicates a reduction by 0.71, which is moderate. This magnitude of reduction effectively degrades the voice quality by one or two quality ratings, as per Table 1. With the hardware and settings we used, the perfect voice signal, with the score of 4.5, sent into the network would degrade to 3.79. Therefore, users should expect at most “Medium” voice quality.

We can observe other interesting information from the measurement summary. MOS for the uplink is nearly the same as for the downlink, so the same conclusions stand. However, loss is much higher on the uplink without any apparent effect on the MOS. This confirms that loss rates incurred on this WiMAX network are not an impediment to Skype traffic, due to efficient handling of the Skype protocol and/or codec. Loss rate shows no correlation with MOS.

We find that the correlation coefficient between jitter and MOS is -0.91 for the downlink, and -0.65 for the uplink, indicating a strong negative correlation, as expected, and confirms findings in [1] that jitter has a significant effect on user satisfaction. However, RTT has no correlation with MOS.

The average PESQ MOS values of 2.49 and 2.47 are comparable to the ones obtained from a live UMTS network, 2.49 and 2.24, for the downlink and the uplink, respectively. However, degradation due to hardware in our measurements was very high, and in the UMTS study very low, so that the relative quality degradation due to network environment is much lower in our case.

In addition, we have anecdotal evidence that excessive RTT on the uplink, between 1000 ms and 2000 ms, has virtually no effect on MOS, confirming that RTT has very small effect on user satisfaction, as argued in [1]. These results were obtained during heavy congestion on campus network on June 16, 2008.

#### 4.4 Effects of WiMAX network on VoIP traffic

We next turn to the analysis of traffic traces collected at the source and destination of a Skype session. We summarize the measurements of one downlink and one uplink Skype call in Table 3. *Packet rate* is the average number of sent or received packets per second (PPS) of the call. *Data rate* is the average total amount of data sent (received) in each second of the call, including all protocol headers. This number represents the total bandwidth used by Skype for voice communication. *Mean PIAT* is the mean Packet Inter-Arrival Time, and represents the mean time between sent (received) packets. For brevity and convenience of notation, we refer to the sender-side observation as PIAT rather than “inter-packet generation time”. *StDev PIAT* is the standard deviation of PIAT.

Table 3. Skype traffic on the WiMAX network.

	Downlink		Uplink	
	Sender	Receiver	Sender	Receiver
<b>Packet rate (PPS)</b>	21.00	20.65	26.71	26.33
<b>Data rate (Kbps)</b>	43.10	42.16	47.74	46.39
<b>Mean PIAT (ms)</b>	47.9	48.3	37.6	38.3
<b>StDev PIAT (ms)</b>	15.7	34.5	13.5	25.7

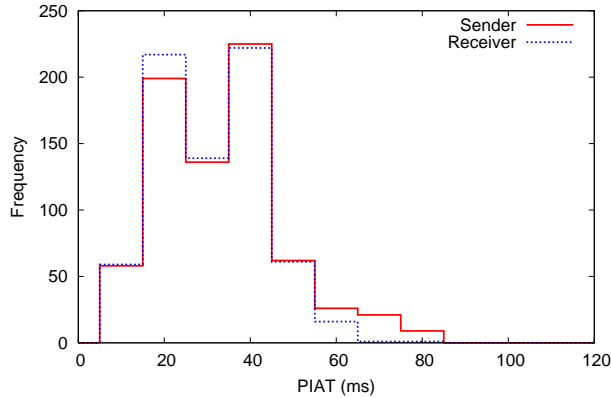


Fig. 3. Sender and receiver side distributions of sample PIAT of a Skype call on Ethernet are nearly identical.

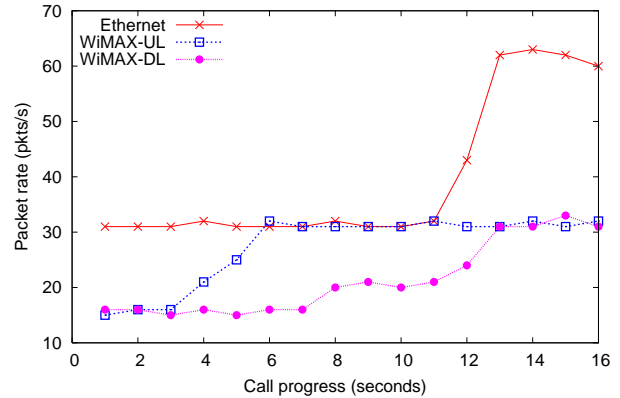


Fig. 4. Skype increases packet rate over 16 seconds of the test call and stabilizes at 31 PPS over WiMAX or 62 PPS over Ethernet.

Differences in measurements between the sender and receiver side reveal potential reasons for the degraded VoIP call quality. On both downlink and uplink, the average received packet and bit rates are very close to the ones sent, corresponding to the low packet loss rates reported by Skype. The most important difference is the variability in PIAT at the receiver, which is about twice as high as on the sender side. This much jitter in packet arrival times is the major factor that affects call quality.

Interestingly, packet sending rate is higher on the uplink, where link capacity and delay are lower. Bit rate on the uplink is not proportionally higher as the packet rate because packet sizes used are smaller on the uplink. This behavior is also observed in Skype version 3.2 that uses the same codec. We attribute this behavior to the adaptation mechanism under higher loss and shorter RTT conditions of the uplink, where smaller and more frequent packets would lead to fewer dropped audio frames.

We plot the histogram of PIAT at the sender and receiver to show the impact of the network environment. We start by showing the sample distribution of PIAT as captured on the Ethernet in Fig. 3. The primary observation is that the sender's and receiver's distributions are bi-modal and nearly identical. This agrees with the results of negligible signal degradation and near-perfect environment for Skype.

However, it is assumed that Skype generates voice data at a constant rate, i.e. the average rate of the codec, without silence suppression. That is not the case, according to our traces, where the following is observed (Fig. 4). Over Ethernet, during the first 11 seconds of the voice clip, the packet rate is 31 PPS. It then doubles to 62 PPS and stays at this level, even for longer clips. While the packet rate is 31 PPS, the data rate increases over time from 45 to 58 Kbps due to larger packet size. The higher packet rate of 62 PPS has a consistent 72 Kbps data rate. Therefore, there are two forms of adaptation to favorable network conditions, using packet rate and packet size. The observed packet rates and the adaptive behavior explain the two modes of the PIAT distribution.

Next, we consider the PIAT distribution on WiMAX network for the downlink, as shown in Fig. 5. We notice significant differences from the Ethernet scenario. Sender-side PIAT is spread out and majority of packets leave at 40 to 70 ms intervals. This is caused by a slightly different pattern of increasing packet rate, from 15 - 16 PPS during the first 7 seconds, followed by around 20 PPS for the next 5 seconds, and then peaking and stabilizing at 31 PPS (Fig. 4). These packet rates correspond to the shape of the PIAT distribution on the sender side. The effect of the WiMAX network is obvious in that the receiver-side distribution is very much spread out, nearly uniform over a wide interval between 10 and 70 ms. The tail is truncated and it actually extends to 190 ms. Packets arrive with higher burstiness, with 7.5% having PIAT over 100 ms, which is highly likely to be noticeable by the user.

Finally, the PIAT distribution on the uplink shows different characteristics than on the downlink (Fig. 6). The distribution has a dominant mode at 40 ms, corresponding to the prevalent sending rate of 31 PPS. The WiMAX network does not affect the shape of the distribution at the receiver side, which is only slightly more spread out, but has the same general shape. It is clear that the uplink has smaller effect on Skype traffic than the downlink. The WiMAX link has the highest delay variability on the path, as shown in our previous work [9]. The different impact of the downlink and uplink can be attributed to the compound effect of the variable delay of the wireless hop, the scheduling algorithm at the BS,

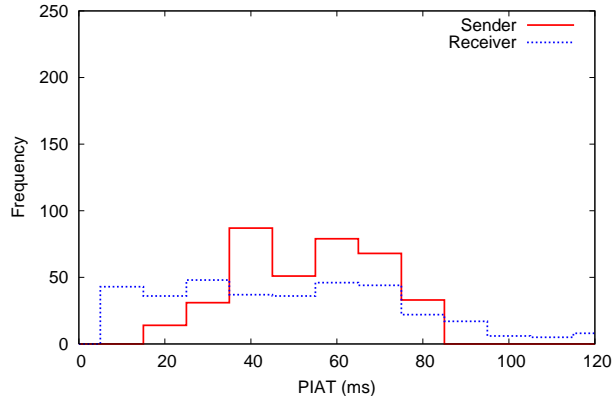


Fig. 5. WiMAX downlink significantly changes the PIAT distribution of a Skype call.

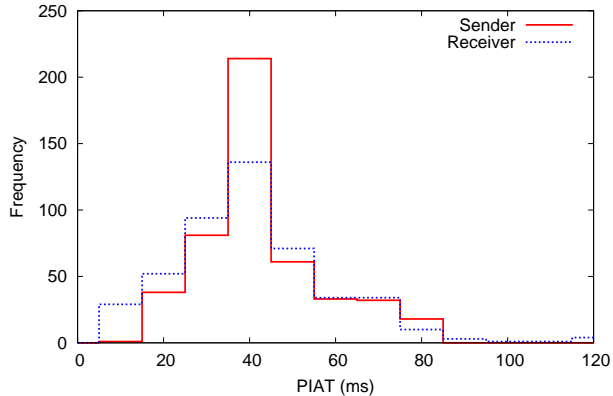


Fig. 6. WiMAX uplink has minor effect on the PIAT distribution of a Skype call.

and the bandwidth request mechanism of the SS. As shown in [6], SS can continuously request bandwidth on the uplink when data backlog exists and ensure that packets are accepted to the network within a bounded interval.

#### 4.5 Alternative voice quality assessment

In this section, we briefly try to compare the obtained PESQ MOS results with USI, the alternative user satisfaction assessment method. USI is defined as follows:  $USI = 2.15\log(\text{bit rate}) - 1.55\log(\text{jitter}) - 0.36RTT$  [1].

It is suggested to use the highest bit rate, smallest bit rate jitter and smallest RTT recorded over the pre-determined intervals within the session [1]. Since our sample audio clip lasts only 16 seconds, we use the highest bit rate at which Skype stabilizes, and the RTT reported at the end of the clip. However, USI model does not account for increasing bit rate over short time scales, which causes bit rate jitter to increase. We therefore use the bit rate jitter over the most stable interval at least 5 seconds long. We compute USI for one of the traces obtained from Ethernet, WiMAX downlink, and uplink scenarios (Table 4). Recall that the full MOS scale is from -0.5 to 4.5, and that USI typically takes values from 4 to 10. Firstly, USI for Ethernet is significantly higher than for WiMAX, which is reasonable to expect. Secondly, USI for WiMAX downlink and uplink is comparable, but MOS differs. Thirdly, considerably higher RTT on the WiMAX downlink does not affect its USI, which is slightly higher than on the uplink due to marginally higher bit rate and lower jitter. This demonstrates the difference between USI and MOS, where USI predicts higher user satisfaction when the source bit rate is higher and jitter lower, although the actual voice signal may be more degraded, as per MOS.

Since USI is based on call duration, it predicts the expected call duration for the scale of its values, such that the logarithmic duration is approximately proportional to USI. According to [1] and the calculated USI, calls over WiMAX are expected to last up to 10 minutes, and over Ethernet more than 100 minutes, essentially as long as desired.

## 5. VIDEO STREAMING PERFORMANCE

To test the video streaming performance, we use RealNetworks applications; RealProducer 11 to create the sample video clip, Helix Server 11 to provide streaming content, and RealPlayer 11 to play the streaming video [2]. RealNetworks software supports both live and on-demand streaming. In this study, we focus only on on-demand streaming.

We collect several performance metrics via a RealTracker tool, specifically tailored to extract the information from the RealPlayer running in the background [17]. RealTracker samples the bit rate, jitter, frame rate and other metrics over

Table 4. Calculated USI and its parameters for Ethernet and WiMAX.

	<b>Ethernet</b>	<b>WiMAX Downlink</b>	<b>WiMAX Uplink</b>
<b><i>Bit rate (Kbps)</i></b>	72.68	61.68	61.57
<b><i>Jitter (Kbps)</i></b>	1.9	2.88	2.97
<b><i>RTT (s)</i></b>	0	0.217	0.172
<b><i>USI</i></b>	9.77	7.14	7.11
<b><i>MOS</i></b>	3.13	2.58	2.20



Table 5. The highest coded bit rate played nearly 2/3 of the time during 32 video streaming experiments.

Coded bit rate	Time Played
180.9 Kbps	14%
353.5 Kbps	20.6%
603.5 Kbps	65.4%

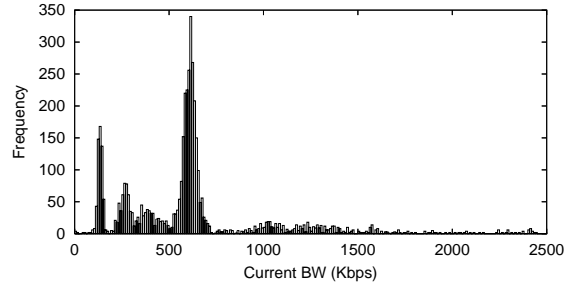


Fig. 7. Histogram of streaming BW over UDP.

half-second intervals of the media playback. In addition, we collect traces to characterize the video streaming traffic on the network and analyze the impact of the network conditions on the data stream.

Experiments are conducted between two laptops, one connected to the campus LAN and running the Helix Server, and another one connected to the Internet via WiMAX link, running the RealPlayer client. The client is configured to use either UDP or TCP. We only consider the WiMAX downlink direction for streaming with one clip streamed at a time, as the most common scenario. Our sample video clip is a 91-second medium-action outdoor recording with background music. Video clip has a resolution of 480x320 pixels with the frame rate of 29.9 frames per second. Media encoding uses RealNetworks' format called SureStream, which allows multiple bit rates to be encoded in a single media object. The purpose of SureStream is to allow the streaming server to switch between bit rates depending on network conditions. This improves the quality experienced by clients by avoiding re-buffering and interruptions during playback. Our video clip is encoded at three rates, 603.5 Kbps, 353.5 Kbps, and 180.9 Kbps. Audio stream bit rates are 96.5 Kbps paired with two higher video rates, and 44.1 Kbps paired with the lowest video rate.

These bit rates are chosen to resemble commonly used video quality on the Web. For example, popular video sharing Web site YouTube has a standard video rate of 314 Kbps with 320x240 resolution [21]. Reportedly, the new and higher quality format is upcoming and some YouTube videos have been re-encoded to 602 – 665 Kbps with 480x360 resolution [21]. The lowest bit rate represents the lower bound on acceptable broadband experience and it is the last resort should bandwidth not be available for the higher two bit rates.

### 5.1 Video streaming over UDP

For video streaming over UDP, our results include 32 runs of the test video clip. We present measurements of coded bit rate, streaming bandwidth (BW), and jitter taken at each second during all 32 test runs. *Coded bit rate* is the bit rate at which the video is encoded as the frames are sent by the server. It can take one of three values encoded in the SureStream format inside the media file, 603.5 Kbps, 353.5 Kbps, and 180.9 Kbps. *Streaming BW* is the actual data rate achieved at the client side, including the media streams and overhead, i.e. the total bandwidth used by the application.

Table 5 shows the portion of time that each coded bit rate was received and played by the client. The maximum bit rate was played 65.4% of the time and the next lower one 20.6% of the time. We can conclude that 86% of the time a user could expect to watch video streams of at least the YouTube's standard quality. Frame rate played was exceptionally high, with full frame rate of 29.9 FPS playing over 97% of the time.

Next we consider streaming BW to get an insight into the causes of such distribution between the received coded bit rates. A histogram of streaming BW samples is shown in Fig. 7. In this multi-modal distribution, it is clearly seen that the peaks and concentration of samples correspond to the three coded bit rates. However, there is an additional concentration around 270 Kbps, which can be attributed to network conditions and the transitions between two lower coded bit rates. In fact, the observed streaming BW can be depicted by a state-transition diagram (Fig. 8). Streaming BW over about 723 Kbps (Turbo Rate) can only be observed during the first few seconds, but once the transition is made to High Rate, it will never go back to Turbo Rate. Transitions between Low, Medium, and High Rates can happen as the available bandwidth of the link changes. Transitions across two states occur rarely.

We also note that the extremely small measurements below 110 Kbps were observed only 0.43% of the time, and that the client stalled or nearly stalled at 10 Kbps or less only 0.086% of the time.

Jitter is calculated as a standard deviation of the inter-frame playback time over the entire clip. We present a histogram and CDF of jitter measurements in Fig. 9. All our trials produced smooth or average streams, according to measured

jitter. About 63% of trials played smooth streams, and another 30% produced average streams with jitter under 160 ms, which is a “smoother” half of the average category. All smooth clips with jitter less than 50 ms played fully with the highest coded bit rate. Both coded bit rate and streaming BW show a strong negative correlation with jitter, indicated by correlation coefficients of -0.95 and -0.94, respectively.

We next conduct network-layer analysis of the video streaming traffic. Traces of four streaming runs are captured on both server and client side and compared with respect to PIAT. Server side capture is done at the network interface after the packets have passed through the buffer and have been handled by various system calls. We therefore cannot directly measure packet generation rate of the multimedia server, we rather use the traces to infer server’s behavior.

RealNetworks media stream includes distinct audio and video packets, whose traffic structure differs. Summary of PIAT statistics is shown in Table 6. Audio packets are sent on average every 100 ms by the server. Packets arrive with a slightly higher PIAT at the receiver, but the standard deviation of the sent and received PIAT is comparable. Video traffic is sent at about 10 times the rate of audio traffic, and the video PIAT is more variable than for the audio. However, video packet arrivals have smaller PIAT variation than when being sent out, according to the PIAT standard deviation.

Table 6. PIAT of audio and video streams.

	AUDIO		VIDEO	
	Server	Client	Server	Client
<i>Mean (ms)</i>	0.0992	0.1064	0.0102	0.0107
<i>Min (ms)</i>	1.7E-05	1.5E-05	1.3E-05	9.0E-06
<i>Max (ms)</i>	1.0156	1.7811	0.5626	0.5050
<i>StDev (ms)</i>	0.0710	0.0816	0.0137	0.0121

To better understand the summary results we plot the histograms of the PIAT at both sender and receiver for all test runs. For audio, Fig. 10 shows that the packets are sent out according to a fairly spread-out distribution with the peak at 80 ms and the tail extending past 500 ms. The distribution is more smooth at the client and extends approximately over the same range. Clearly, WiMAX network does not have a strong impact on the PIAT of the audio stream. We also note that the audio packet size is always the same: 1452 bytes.

It takes more than one data packet to carry the video frame. In fact, the sample clip consists of about 2700 frames transported by 8615 data packets over 90 seconds. It takes on average 3.19 packets per frame that the server must send in every 33.33 ms interval. Since the data packets are not the same size, and the frame sizes are also not the same, we analyze the trace data to extract the packets that belong to the same video frame using their media timestamp. We find two common cluster structures of fragmented frames into packets. Each cluster consists of 4 frames, one large and three small ones. The small frames always consist of two data packets. The large frame most commonly comes in two sizes, 4 or 5 packets, but sometimes more. Large frame has a size of over 5000 bytes, and the small ones range between 900 and 1500 bytes.

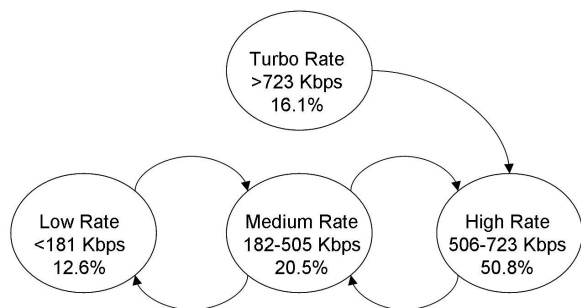


Fig. 8. Transitions between streaming BW levels and average time each level was observed.

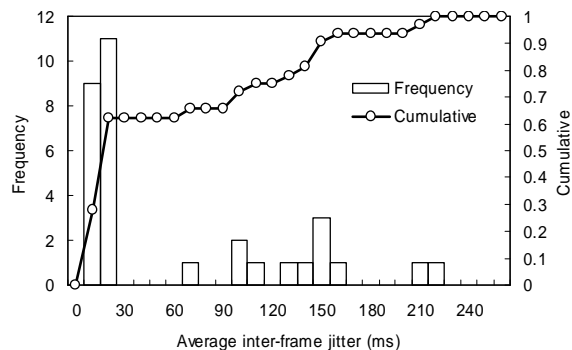


Fig. 9. Histogram and CDF of observed average jitter over UDP for 32 test runs.

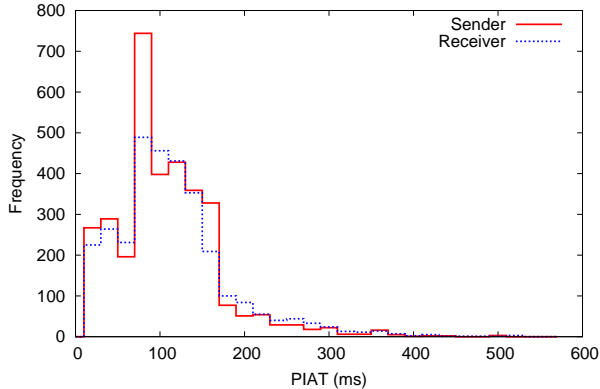


Fig. 10. Distributions of sample PIAT for audio stream.

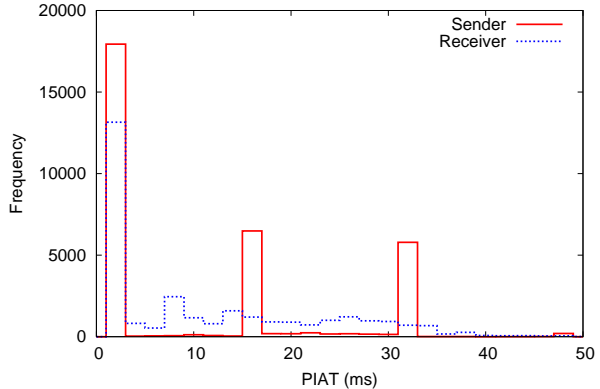


Fig. 11. Distributions of sample PIAT for video stream.

The distribution of the video data PIAT is shown in Fig. 11. About 94% of packets depart the server at three dominant intervals: 1, 16, and 32 ms. These three intervals are not related to the three coded bit rates, they occur even when only the highest coded bit rate is streamed and played at the client. The intervals can be explained by the frame generation requirements of the particular video clip. The frame rate of the sample video clip is about 30 FPS, therefore frames should be sent out every 33.33 ms on average, corresponding to the 32 ms interval. Also, it takes about 17 ms to send a large packet onto the wire at the coded rate of 603.5 Kbps, accounting for the 16 ms interval. Server attempts to match the sending rate to the coded bit rate, rather than to the physical link rate. The 1 ms interval accounts for multiple packets sent at the same time, as required by the streaming rate and frame size.

The receiver-side distribution completely lacks the three peaks corresponding to the sender-side distribution. One peak still exists at 1 ms, whereas the rest of the packets (57%) arrive mostly at intervals of 2 to 36 ms. RealPlayer’s buffering and the streaming protocol’s recovery mechanisms efficiently deal with the significant network impact to still provide high quality experience.

Our network-level analysis reveals that the RealNetwork media streaming format has evolved, i.e. the traffic structure has changed since the earlier study [12]. In particular, we do not observe audio frame fragmentation into multiple packets, nor clusters of back-to-back audio packets in the network. We only observe full-size UDP packets with audio data. The video traffic structure has subtle differences as well, for example, packets arriving at 1 ms intervals do not necessarily belong to the same frame. We do observe familiar clustered structure of video frames, however with more packets per frame due to higher coded bit rates than used in [12].

## 5.2 Streaming over TCP

Due to space constraints, we briefly note that the streaming performance over TCP strongly depends on user location. For reasons of reduced TCP throughput at campus location, as explained in our earlier work [9], video streams are too poor to watch. However, it is possible to achieve good streaming performance over TCP at some residential locations.

## 6. SUMMARY AND CONCLUSIONS

In this paper, we present the measurement results of multimedia applications from a commercial deployment of a WiMAX-based BWA network. We emulate single user experience by running single sessions of VoIP traffic and video streaming while capturing traffic traces and measurements from the applications.

We find that the overall performance of VoIP over a relatively new WiMAX technology is acceptable, but that voice quality will not be the same as on the wired networks. The average PESQ MOS values are comparable to the ones obtained from a live UMTS network. However, the reference point of our measurements was much lower so that relative quality degradation due to network environment is smaller in our case. When compared to the testbed results, both UMTS and WiMAX testbeds had higher MOS and R-factor [5, 11], which shows a need for more live measurements in order to fully understand and consequently improve the new wireless technologies. The main result from VoIP experiments is that the WiMAX user should expect voice quality degradation by a full rating with respect to source signal, for example from “High” to “Medium”.

Performance of video streaming is consistently very good over UDP, with stable frame rate and high quality video playing for about 2/3 of the time. Inter-frame jitter is very low allowing for smooth playback throughout nearly entire video clip. Our results largely match those obtained in 1xEV-DO network, when stationary client is considered [14].

In addition to performance results, we confirm the changing nature of popular multimedia applications, Skype and RealNetworks media server. While the results we present are specific to the particular network deployment, they provide initial hints about the nature and impact of a live WiMAX-based system. Our future work continues by investigating ways to improve the performance of applications over WiMAX links.

## REFERENCES

- [1] Chen, K.-T., Huang, C.-Y., Huang, P. *et al.*, "Quantifying Skype user satisfaction," Proc. SIGCOMM 36, 399-410 (2006).
- [2] RealNetworks, Inc., <http://www.realnetworks.com/>, June 20, 2008.
- [3] Haskins, W., Aussie ISP wails on WiMax, TechNewsWorld, <http://www.technewsworld.com/story/62289.html>, June 25, 2008.
- [4] IEEE std 802.16e-2005 and IEEE std 802.16-2004, IEEE standard for local and metropolitan area networks], (2005).
- [5] Pellegrini, F. D., Miorandi, D., Salvadori, E. *et al.*, "QoS support in WiMAX networks: Issues and experimental measurements," CREATE-NET TR N. 200600009, 2006.
- [6] Cicconetti, C., Lenzini, L., Mingozzi, E. *et al.*, "Quality of service support in IEEE 802.16 networks," IEEE Network, 20(2), 50-55 (2006).
- [7] Ramachandran, S., Bostian, C. W., and Midkiff, S. F., "A link adaptation algorithm for IEEE 802.16," Proc. WCNC 3, 1466-1471 (2005).
- [8] Yang, X., Venkatachalam, M., and Mohanty, S., "Exploiting the MAC layer flexibility of WiMAX to systematically enhance TCP performance," Proc. Mobile WiMAX Symposium, 60-65 (2007).
- [9] Halepovic, E., Wu, Q., Williamson, C. *et al.*, "TCP over WiMAX: A measurement study," Proc. MASCOTS, to appear (2008).
- [10] Perez, J. A., Donnet, B., and Bonaventure, O., "Preliminary analysis of the TCP behavior in 802.16 networks," Proc. WEIRD Workshop on WiMAX, Wireless and Mobility (2007).
- [11] Hossfeld, T., Binzenhöfer, A., Fiedler, M. *et al.*, "Measurement and analysis of Skype VoIP traffic in 3G UMTS systems," Proc. IPS-MoMe (2006).
- [12] Kuang, T., and Williamson, C., "Hierarchical analysis of realmedia streaming traffic on an IEEE 802.11b wireless LAN," Computer Communications, 27(6), 538-548 (2004).
- [13] Mahanti, A., Williamson, C., and Arlitt, M., "Remote analysis of a distributed WLAN using passive wireless-side measurement," Performance Evaluation, 64(9-12), 909-932 (2007).
- [14] Sessini, P., Leventer, M., and Mahanti, A., "Video to go: The effects of mobility on streaming media in a CDMA2000 1xEV-DO network," Proc. MMCN 6504, 650408 (2007).
- [15] Johannesson, N. O., "The ETSI computation model: A tool for transmission planning of telephone networks," IEEE Communications, 35 (1) 70-79 (1997).
- [16] Cole, R. G., and Rosenbluth, J. H., "Voice over IP performance monitoring," SIGCOMM Comput. Commun. Rev., 31(2), 9-24 (2001).
- [17] Wang, Y., Claypool, M., and Zuo, Z., "An empirical study of RealVideo performance across the internet," Proc. ACM Internet Measurement Workshop, 295-309 (2001).
- [18] Baset, S. A., and Schulzrinne, H. G., "An analysis of the Skype peer-to-peer Internet telephony protocol," Proc. INFOCOM, 1-11 (2006).
- [19] Lindblom, J., "A sinusoidal voice over packet coder tailored for the frame-erasure channel," IEEE Transactions on Speech and Audio Processing, 13(5), 787-798 (2005).
- [20] Signalogic, Speech codec WAV samples, [http://www.signalogic.com/index.pl?page=codec\\_samples](http://www.signalogic.com/index.pl?page=codec_samples), April 2, 2008.
- [21] YouTube, Wikipedia, The Free Encyclopedia, <http://en.wikipedia.org/w/index.php?title=YouTube&oldid=221635399>, 26 June 2008 21:34 UTC.