2009 Third International Conference on Next Generation Mobile Applications, Services and Technologies

A Comparative Study of Bandwidth Requirements of VoIP Codecs Over WiMAX Access Networks

Ashraf A.Ali [1]
Athens Information Technology
asal@ait.edu.gr

Spyridon Vassilaras
Athens Information Technology
svas@ait.edu.gr

Konstantinos Ntagkounakis
Intracom Telecom
knta@intracom.gr

Abstract—**WiMAX, as a wireless broadband access technology, aims at providing the end users with broadband services identical to that provided by DSL. The data rate requirements of VoIP connections over WiMAX links greatly depend on the choice and parameterization of the employed codec, the uplink scheduling algorithm, the header overhead of all network layers as well as applied header compression techniques. The knowledge of the exact data rate requirements of key services is very important for network dimensioning and planning. In this paper we provide a study of the bandwidth and delay requirements of VoIP services in WiMAX networks. This includes an analysis of the effects of Robust Header Compression, Payload Header Suppression, Voice Activity Detection with Discontinuous Transmission and other bandwidth affecting factors. Finally, the overhead and induced delay by each one of the WiMAX uplink scheduling algorithms for real time services is analyzed in this paper**.

Keywords- VoIP; WiMAX; 802.16; Ethernet; codecs

I. INTRODUCTION

The WiMAX wireless technology is a last mile solution for wireless broadband access. Due to its wireless nature, it is a cost effective solution to provide DSL like services for both stationary and mobile end users. The 802.16-2004 standard [1], usually referred to as 802.16d, supports only fixed broadband wireless access systems. The more recent 802.16-2005 (or 802.16e) standard [2] supports both fixed and mobile broadband access at the expense of lower transmission range. Consequently, WiMAX will compete with both wired technologies (especially in green field deployments) and 3G and beyond cellular technologies for the broadband access market.

A WiMAX network is based on infrastructured network architecture with Base Stations (BS) covering a wide area in a cellular topology. A mobile or fixed Subscriber Station (SS) is connected to a BS through a wireless link. As the Carrier Ethernet technology gains increased popularity among Telecom Operators, Carrier Ethernet backhauling becomes the solution of choice: Ethernet frames are received by the base station and then encapsulated directly into WiMAX frames. The WiMAX standard supports Ethernet encapsulation in the Convergence Sublayer (CS) which treats the Ethernet frames as Service Data Units (SDUs) for the lower layers. These SDUs are classified based on their header fields into different Connection Identifiers (CID) and Service Flow Identifiers (SFID) before being forwarded to the WiMAX Common Part Sublayer (CPS). The CPS packs or fragments the received SDUs and adds a MAC header to form Payload Data Units (PDUs). Finally, the PDUs are forwarded to the scheduler in order to be scheduled in the data burst fields inside the WiMAX frame. The encapsulated fields in the WiMAX PDU are shown in Figure 1. An alternative to the Ethernet CS is the IP convergence sublayer which is used in case we wish to transmit IP frames over WiMAX without Ethernet encapsulation. It is important to note that we can use both convergence sublayers over a connection if we have both Ethernet and IP SDUs. Clearly, header overhead in IP SDUs is less than in Ethernet SDUs.

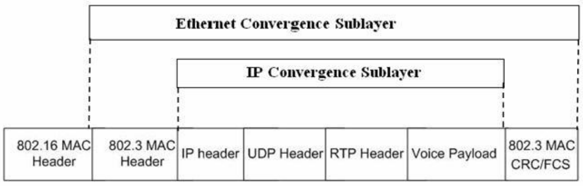

Figure 1 VoIP over Ethernet over WiMAX frame

As WiMAX networks are all-IP networks, voice services over WiMAX are implemented as Voice over IP (VoIP). The data rate generated by VoIP codecs differs from one codec to another, as there is a tradeoff between the voice quality, generated date rate, and complexity of the codec. Since wireless resources are scarce, the need to deploy bandwidth efficient codecs with acceptable voice perception quality and moderate complexity is of great importance for WiMAX access networks. In addition, as digitized voice is packetized in small chunks the header overhead in VoIP is significant. In this paper, we examine the CS data rate required by a VoIP over Ethernet over WiMAX and IP over WiMAX flows. We compare the bandwidth requirements of all widely used codecs and analyze the performance of various rate reduction techniques (such as Voice Activity Detection and Header Suppression). As VoIP is expected to be a key application over WiMAX networks, this analysis is very important for network dimensioning and planning, call admission control and optimization of the application layer protocol implementation and parameterization.

The rest of this paper is organized as follows: Section II presents different types of several voice codecs which are used in various digitized voice application and derives their data rate requirements for VoIP over WiMAX flows. The performance of different suppression and compression techniques is investigated in Section III. The effects of using the Voice Activity Detection and Silence Suppression with Discontinuous Transmission techniques are detailed in Section IV. An overview of the scheduling algorithms and delay analysis is given in Sections V and VI respectively. Simulation results are presented in section VII and finally conclusions are driven in Section VIII.

[1] Corresponding author

978-0-7695-3786-3/09 $26.00 © 2009 IEEE
DOI 10.1109/NGMAST.2009.47

II. VoIP Codecs

A Voice Codec (or vocoder) is used at the subscriber side to convert the analogue voice waves to digital pulses and vice versa. There are different codec types based on the selected sampling rate, data rate, and implemented compression algorithm. In order to determine the bandwidth requirements of VoIP connections, we will start by presenting common VoIP Codecs and their associated characteristics. Among these codecs, G.711 and G.729 are the most widely used codecs in existing networks.

The **G.711** codec is currently used in a wide range of applications. Its voice sampling rate is 8 kHz and each sample is encoded with 8 bits resulting in a constant 64 kbps bit rate and offering a very good voice quality. Samples are packed into frames every 10 ms. In order to further reduce the bit rate generated by this codec without damaging the voice quality in packet based multimedia applications such as VoIP, a modified version of the base algorithm was proposed in G.711 Appendix II [3]. This version makes use of Discontinuous Transmission (DTX), Voice Activity Detection (VAD), and Comfort Noise Generation (CNG) algorithms. Table II.1/G.711 in [3] shows that bandwidth savings can reach up to a 39% reduction compared to the basic G.711 codec. Obviously, the data rate generated by this version of the codec is variable in time.

The **G.729** codec also generates speech frames every 10 ms containing 80 voice samples (collected at a sampling rate of 8000 samples per second) [4]. However, it requires a 5 ms look-ahead delay before producing any new frame. Annex A of the standard (G.729 a) has the same main features of G.729 but is considered as a reduced complexity version. It has been developed for multimedia simultaneous voice and data applications. In addition, a silence suppression scheme is developed as Annex B of the standard (G.729 b). This version defines a voice activity detector (VAD) and comfort noise generator (CNG) for G.729 in order to reduce the transmission rate during silence periods.

Enhanced Variable data-Rate Coding (**EVRC**) and **Adaptive Multi-Rate Coding** (**AMR**) are examples of codec types that can trade off bit rate for voice quality. The AMR codec is used in GSM and UMTS networks and can operate in 8 different modes with bit-rates of 4.75, 5.15, 5.90, 6.70, 7.40, 7.95, 10.2 or 12.2 kbps [5]. The codec generates one frame every 20 ms. Depending on the desired bit rate each frame contains 95, 103, 118, 134, 148, 159, 204 or 244 bits respectively. On the other hand, EVRC dynamically changes the data rate from the full rate to 1/8th of the full rate based on the voice activity. EVRC generates one frame every 20 ms with a voice sampling rate of 8 kHz and 16-bit encoded samples. It then uses compression and VAD to reduce the frame length in one of three different sizes: 171 bits (Rate 1), 80 bits (Rate 1/2), or 16 bits (Rate 1/8).

G.723.1 is an example of low rate codecs. It operates at two bit rates either 5.3 or 6.3 kbps and can switch between the two rates at the frames boundary. The frame size is 30 ms with look-ahead time of 7.5 ms, resulting in a total algorithmic delay of 37.5 ms. The encoder operates on blocks (frames) of 240 samples each (30 ms at an 8-kHz sampling rate). After compression, each frame holds 189 bits in case of 6.3 Kbps rate and 158 bits in case of 5.3 Kbps rate [6].

G.722.1 is a digital wideband codec algorithm which provides an audio bandwidth of 50 Hz to 7 kHz, operating at a bit rate of 24 kbps or 32 kbps. It operates on 20-ms frames (320 samples) of audio with look-ahead time of 20 ms. The frame size is 480 bits for 24 Kbps and 640 bits for 32 Kbps. In G.722.1 Annex C the sampling rate doubles from 16 to 32 kHz, and it can generate three different packets sizes; 480 bit, 640 bit, and 960 bit over the same frame duration. Hence, the data rate also changes based on the packet size [7].

The **internet Low Bit rate Codec** (**iLBC**) is suitable for robust voice communication over IP. The codec is designed for narrow band speech and results in a payload bit rate of 13.33 kbps with an encoding frame length of 30 ms and 15.20 kbps with a frame length of 20 ms. The iLBC codec enables graceful speech quality degradation in the case of lost frames, which occurs in connection with lost or delayed IP packets.

In order to derive the overall data rate requirements of VoIP flows over Ethernet, we need to add all header overhead: Ethernet MAC header, IP header, UDP header and RTP header. We will refer to the final calculated data rate as the default Ethernet Convergence Sublayer rate. Similarly, we will calculate the needed data rate for encapsulating the Voice frames over IP packets without counting the added Ethernet headers. We will refer to the calculated rate as IP CS data rate. Table I compares all described codecs in terms of total MAC SDU size and default Ethernet CS and IP CS data rate.

As shown in Table I the pure injected Ethernet rate ranges from 21.8 kbps to 113.6 kbps depending on the codec used. Such values are considered relatively high especially when a large number of users need to maintain VoIP calls within a small geographical area. Therefore, different techniques, described in the following sections, are being employed in order to reduce the data rate required for one VoIP stream.

III. Header Suppression and Compression

As discussed previously, the need for reducing the overall bandwidth requirements of a VoIP flow becomes very important in wireless networks. In this Section, we investigate methods for reducing the header overhead of VoIP packets. In Payload Header Suppression (PHS) the redundant parts due to the higher layers in the payload header of MAC SDU are sent only once and then suppressed in the following SDUs. Then at the receiver side the suppressed header fields are reinstated.

When PHS is enabled, each MAC SDU is prefixed with a payload Header Suppression Index (PHSI) which references the Payload Header Suppression Field (PHSF). The suppressed fields and thus the total size of the packet after header suppression differ from one packet to another according to certain PHS rules. Commonly, the Ethernet Header (14 bytes) is suppressed. 37 bytes are suppressed from the IP header in case that IPv6 is used and 8 bytes are suppressed in case IPv4 is used which represent the source and destination addresses.

TABLE I. VOIP DATA RATE FOR ETHERNET CS AND IP CS.

Codec	G.711	G.729	G.723.1[A]	G.722.1[B]	G.722.1[C]	G.722.1C[D]	EVRC[E]	EVRC[F]	EVRC[G]	AMR[H]	iLBC[I]
Sample Time (ms)	10	10	30	20	20	20	20	20	20	20	20
Frame Size[J]	640	80	158	480	640	960	172	80	16	95	303
Packets Per Second	100	100	33	50	50	50	50	50	50	50	50
IP,RTP,UDP headers	320	320	320	320	320	320	320	320	320	320	320
Ethernet Header[K]	176	176	176	176	176	176	176	176	176	176	176
Total Size	1136	576	654	976	1136	1456	668	576	512	591	799
Ethernet CS data rate[L]	113.6	57.6	21.80	48.8	56.80	72.8	33.4	28.8	25.6	29.5	39.9
IP CS data rate[M]	96	40	15.9	40	48	64	24.6	20	16.8	20.7	31.1

A. 5.3 Kbps B. 24 Kbps C. 32 Kbps D. 48 Kbps E. Rate 1 F. Rate 1/2 G. Rate 1/8 H. 4.75 Kbps I. 15.2 Kbps
J. All sizes measured in bits K. Including the CRC header and 802.1Q header L.Calculated in Kbps M. calculated in Kbps

Moreover, in the UDP header 4 bytes are suppressed that represent the source and destination port addresses. Finally, from the RTP header we can suppress the SSRC identifier (4 bytes) [8].

Based on the above suppression rule the total header size (Ethernet, IP, UDP, RTP, CRC) can be reduced from 58 bytes to 30 bytes in case we use IPv4. On the other hand, the total header size is reduced from 78 to 59 in case of IPv6. However, we need to add one additional byte for the Payload Header Suppression Index (PHSI).

Another way to save bandwidth and reduce the overhead is by using header compression. The CS may use one of many different compression algorithms to compress the header and decompress it again at the receiver side. There are many header compression protocols among which the Robust Header Compression (ROHC) protocol is one of the best algorithms because it has the ability to tolerate high loss rate in wireless connections [9]. The protocol allows for different degrees of compression efficiency according to its operating mode. There are three modes of operation; the Unidirectional, the Bidirectional Optimistic and the Bidirectional Reliable mode.

The basic concept of ROHC is to exploit the redundancies in packet headers. IP, UDP, and RTP headers all have many redundancies within the same packet (intra-redundancy) and between different packets within the same service flow (inter-redundancy). Moreover, the changes in many fields are incremental between consecutive packets. So we can efficiently suppress and compress many fields in the headers. The compression algorithm of ROHC is quite complicated and as mentioned previously the compression ratio and efficiency depends on redundancy among packets. However, it was found that the compression ratio for IPv4/UDP/RTP headers is 2.5% to 5% which means that the 40 bytes header is compressed to 1~2 bytes. For simplicity we will assume that the size of the header after compression is 2 bytes. On the other hand using the high efficiency compression on IPv6/UDP/RTP headers reduces the header size from 60 to 4 bytes. So, generally speaking ROHC compresses the header to less than 10% of its original size [10]. However, it is important to not that the Ethernet frame headers are not compressed by ROHC so the Ethernet header is kept intact.

Figure 2 shows the compression efficiency of ROHC and PHS when the Ethernet CS is used. The ROHC compression ratio is higher than that of PHS and the percentage gain differs from one codec to another. Naturally, codecs that have large header size compared to the payload size (like G.729) are the ones that benefit the most by header compression and suppression techniques. Similarly, Figure 3 shows the compression efficiency for both PHS and ROHC in case the IP CS is used.

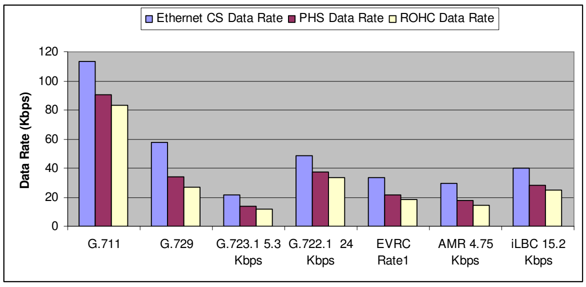

Figure 2 Compression effeciency of PHS and ROHC over the Ethernet Convergence Sublayer

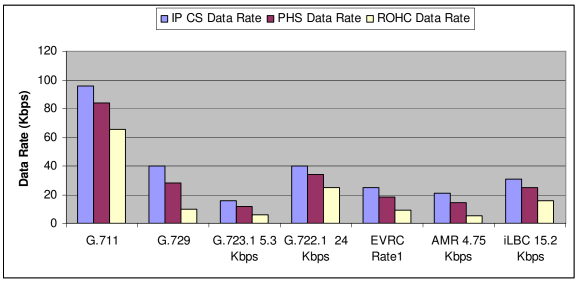

Figure 3 Compression effeciency of PHS and ROHC over IP Convergence Sublayer

It is important to note that the WiMAX MAC header and CRC can not be suppressed or compressed; hence we do not count the size of MAC header and CRC in the calculation of compression efficiency and we limit our analysis in WiMAX convergence sublayer. The bandwidth saving percentage for different codecs compared to the default codec rate (without using ROHC or PHS) is shown in table II. We notice from the table that codecs with small payload size such as G.729 has the highest compression efficiency. As shown in Table II, the saving percentage of ROHC is higher than that in PHS for both Ethernet and IP convergence sublayers.

TABLE II. DATA RATE SAVING PERCENTAGE

Codec	**Ethernet CS**		**IP CS**	
	PHS	***ROHC***	***PHS***	***ROHC***
G.711	20.4%	26.7%	12.5%	31.6%
G.729	40.2%	52.7%	30.0%	76.0%
G.723.1 5.3 Kbps	35.7%	46.7%	25.1%	63.5%
G.722.1 24 Kbps	23.7%	31.1%	15.0%	38.0%
EVRC Rate1	34.7%	45.5%	24.3%	61.7%
AMR 4.75 Kbps	39.2%	51.6%	28.9%	73.4%
iLBC 15.2 Kbps	29.1%	38.1%	19.2%	48.8%

IV. VOICE ACTIVITY DETECTION

Apart from reducing the header size, we can also reduce the size of the payload inside the frame. Voice Activity Detection (VAD) or Silence Suppression (SS) is a technique used in many codecs in order to reduce VoIP bandwidth. VoIP frames will not be generated continuously for each user, since there are silent periods and talk periods which follow exponential distribution as shown in table III.

TABLE III. EXPONENTIAL VOICE ACTIVITY MODEL

Parameter	**Value**
Mean Duration of ON-Period (1/α)	352 ms
Mean Duration of OFF-Period (1/β)	650 ms

Each source will alternate between OFF-state and ON-state as shown in Figure 4.

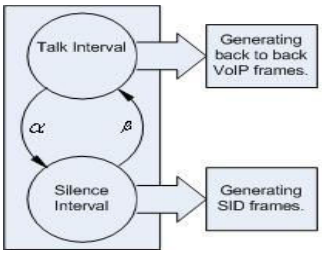

Figure 4 Voice Activity Model

It is important to note that during silence periods, the codec sends small Silence Insertion Descriptor (SID) frames in order to convey the information of background noise to the codec at the receiver side. The SID frame compresses information for sounds that are audible to the human ear. Although the payload of a SID frame is small, we need to consider IP/UDP/RTP headers and the link layer header to calculate the overall packet size. Moreover, SID frames are not sent back to back. They are usually sent either upon considerable change in the background noise or periodically. Figure 5 assumes that the SID frames are sent periodically at a predefined fixed interval T. In the general case SIDs are sent only in a fraction of these intervals denoted by F_{DTX}.

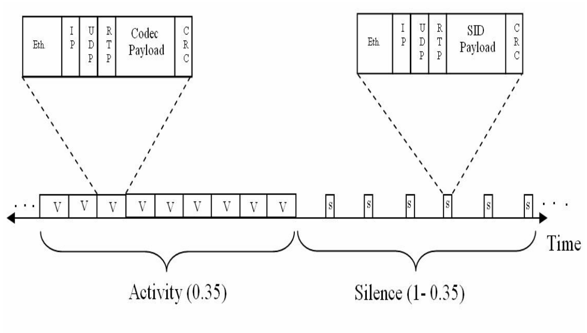

Figure 5 Voice and SID Frames

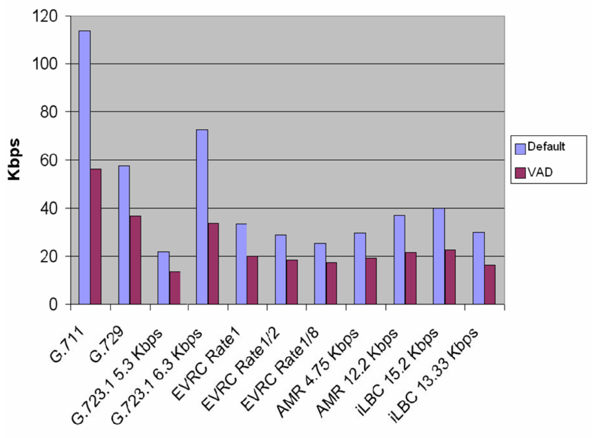

Figure 6 Default Ethernet Rate Vs. VAD Data Rate

We calculate the data rate during voice activity periods, silence periods and the total average rate as follows:

$$R_V = \frac{L_V}{T} \qquad (1)$$

$$R_{SID} = \frac{L_{SID}}{T} \qquad (2)$$

$$R_{Avg} = R_V \cdot \eta + R_{SID} \cdot F_{DTX} \cdot (1-\eta) \qquad (3)$$

Where R_V, L_V, T, R_{SID}, L_{SID}, R_{AVG} and η are: the rate during voice activity periods, the size of a VoIP frame, the period of VoIP and SID frames generation, the rate during a silence interval T which contains a SID frame, the length of a SID frame, the overall average rate and the voice activity ratio respectively. The save in bandwidth using VAD is shown in Figure 6 using exponential traffic models and a F_{DTX} ratio of 0.50.

V. Uplink Scheduling Algorithms Overview

A wide range of scheduling algorithms have been defined for 802.16 such as UGS, UGS-AD, rtPS, nrtPS, Lee's Algorithm and BE scheduling. Most of these algorithms exhibit certain limitation such as waste of uplink resources, additional access delay and MAC overhead when used to support VoIP services with variable data rate and silence suppression [11]. In this section we will focus on the uplink scheduling algorithms for real-time services.

In the **Unsolicited Grant Service** (**UGS**) the scheduler provides to the subscriber station periodic real time fixed grants. Consequently there is no delay at the subscriber side since it has the needed resources all the time. The size of each grant should be enough to fit the size of the data for the associated service with all other headers (such as the generic MAC header and Grant management sub header).

Another scheduling service is the **real time Polling Service** (**rtPS**). In this scheduling service the BS assigns periodic variable size data packets at real time. So, this service requires more overhead but is considered more efficient in using the available bandwidth than UGS since it allows the subscriber station to determine the exact needed bandwidth at every poll. The rtPS algorithm supports variable data rates; the base station assigns resources that are sufficient for the requesting mobile station. This approach has many advantages: the base station only assigns the required bandwidth without wasting resources, so data are transferred more efficiently than other fixed bandwidth allocation algorithms such as UGS. Moreover, there is no need for polling in the silence period since it operates over the minimum needed bandwidth. However, rtPS needs continuous polling each time the mobile station wants to send data, this introduces MAC overhead and additional access delay.

The **extended real time Polling Service** (**ertPS**) is designed as a compromise between the previous two services [12]. To this end, it provides unicast grants in an unsolicited manner like UGS to save the latency of the bandwidth request. However, all the allocations are made dynamically similarly to the rtPS service. ertPS tries to overcome all the problems in other scheduling algorithms, such as the waste of uplink resources, the additional MAC overhead, and the additional access delay. It was developed especially for variable data rate with silence suppression VoIP services. In this algorithm when the size of voice data packet decreases, the voice user informs the BS of his voice status using the grant management sub-header. Thus, only for one frame the user is assigned more bandwidth than needed; in subsequent frames the BS reduced the bandwidth assigned to this user according to the reduced voice packet size. In case that the size of the voice data packet increases, the user informs the BS of his status using bandwidth request (BR) bits in the bandwidth request header. Therefore, as a result of this process, there is no waste of uplink resources as is the case in the previous algorithms.

VI. Subscriber to Core Network Delay analysis

In this section we will analyze the delay of VoIP from the subscriber station to the core network. In ITU-T G.114 [13] it is prescribed that the mouth to ear delay for real time services such as VoIP should not exceed 150 ms in order for voice quality not to be significantly affected. In this section we will present some delay components that contribute to the total mouth to ear delay.

A. *Codec and Packetization Delay*

The ITU-T G.114 standard describes the codec delay. The codec waits for a certain amount of speech samples before processing and compressing them in a frame. The frame generation interval differs from one codec to another. Moreover, some codecs, as described in previous sections, require a look-ahead time to improve the compression efficiency. So, the codec delay is equal to the frame generation interval plus the look-ahead delay (if required). We can neglect the delays due to encapsulation and adding headers before passing the frame to the MAC layer since they are very small. Table IV shows the look-ahead delays of some important codecs.

TABLE IV. CODEC LOOK-AHEAD DELAYS.

Codec Type	G.711	G.729	G.723.1	G.722.1	EVRC
look-ahead delay (in ms)	0	5	7.5	20	10

B. *Scheduling Algorithm Access Delay*

In the downlink, the BS scheduler looks into each peer node and tries to schedule data from its connections to the downlink sub-frame. The service of connections is performed in a round robin way. As long as there is free space in the downlink sub-frame, the scheduler will assign a data burst to the connection. Once the free space in the downlink sub-frame is over, the scheduler will continue scheduling bursts in the next available physical frame [14]. So, as long as there is enough space for the data in the downlink sub frame, the scheduler will send it immediately.

In the uplink, the BS scheduler will first reserve space for contention slots. The information about the bandwidth needs is stored in the peer node objects in the MAC layer. When scheduling uplink transmissions, the scheduler will look at this information from connections, and will use a round robin procedure as for the downlink part. During this process, it will also add 1 OFDM symbol between bursts. The SS scheduler is responsible for taking the data transmission opportunities allocated by the BS and for assigning them to the appropriate incoming connections. If bandwidth is assigned to a given connection and not entirely filled up by it, the SS scheduler will fill the burst with the data from other outgoing connections. If the SS has not an outstanding request, the SS scheduler looks at all outgoing connections and generates bandwidth requests, sent in the contention phase [14]. Each request is of the aggregated type.

The access delay depends on the scheduling service. We will analyze mainly three real time scheduling services that were described in the previous Section: UGS, rtPS, and ertPS.

1) UGS access delay

As mentioned previously, in the UGS algorithm, the BS will provide the SS with periodic fixed size grants that are negotiated during the connection establishment phase. So, the subscriber station can always send the data to the base station. Assuming that the VoIP packet generation period is an integer multiple of the WiMAX frame duration, each VoIP packet will arrive with a fixed delay x_1 after the beginning of the frame. If $x_1 \leq T_{DL}+T_{ttg}$ (ttg stands for transmission transition gap) as shown in Figure 7, then in order to minimize the UGS scheduling delay the UL slot for transmission of this packet should be assigned as early as possible in the UL. However, since more than one VoIP or other delay sensitive flows may be present, there will be some additional delay x_2 until this particular flow can be transmitted. On the other hand, if $x_1 > T_{DL}+T_{ttg}$ the transmission should be scheduled as soon as possible after the arrival of the packet. (Recall that due to the periodicity of VoIP packet generation x_1 is known at the beginning of the frame.) The additional delay x_3 incurred in this case is expected to be smaller than x_2 as x_1 grows and the number of contending packets decreases. However, a small number of packets arriving too close to the UL end might be forced to wait for the next UL in order to get transmitted. Equation 4 shows the scheduling delay incurred by the UGS algorithm.

$$\mathrm{T}_{\mathrm{tx\_ugs}} = \begin{cases} T_{\mathrm{DL}} + T_{ttg} - x_1 + x_2, & \text{if } x_1 \leq T_{\mathrm{DL}} + T_{ttg} \\ x_3 & \text{if } x_1 > T_{\mathrm{DL}} + T_{ttg} \end{cases} \quad (4)$$

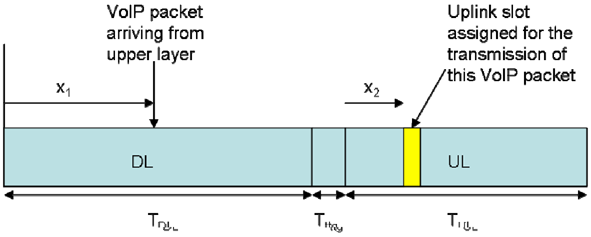
VoIP packet arriving from upper layer
Uplink slot assigned for the transmission of this VoIP packet
x1
x2
DL
UL
TDL
Tttg
TUL

Figure 7. Packet scheduling delay in UGS

Note that if Partial Usage Sub Carrier (PUSC) is used in the uplink interval, the slot duration in the uplink is two OFDMA symbols. Hence, x_2 and x_3 can only be multiples of 2 OFDMA symbol transmit time.

In conclusion, assuming a lightly loaded cell, the minimum UGS scheduling delay is 0 and the maximum is a little larger than $T_{DL}+T_{ttg}$. For heavily loaded systems, queuing delay has to be taken into account in which case the maximum delay can exceed the WiMAX frame duration.

2) rtPS and ertPS access delay

rtPS, as mentioned previously, is used to support real-time service flows that generate variable size data packets periodically. Hence, the BS assigns uplink resources that are sufficient for unicast bandwidth requests to the voice user. This period is negotiated in the initialization process of the voice sessions. Generally, this process is called a bandwidth request process, or polling process. This bandwidth request process always causes MAC overhead and additional access delay. Hence, the rtPS algorithm has larger MAC overhead and access delay than the UGS and ertPS algorithms.

The transmission delay over rtPS scheduling is given in the following equation:

$$T_{tx\_rtPS} = T_{MF} + T_{tx\_ugs} \quad (5)$$

This means that the transmitter has to wait one MAC frame time (T_{MF}) for polling the BS and then transmits the data in the uplink field similar to the UGS case.

On the other hand, in ertPS the user requests the bandwidth for sending the voice packets using the extended PBR (Piggyback Request) bits of the Grant Management sub header. Therefore, the access delay for this algorithm follows the same equation as that of UGS since it uses piggybacking which affects the overhead but not the access time. Only when an increase in VoIP packet size occurs extra bandwidth needs to be requested similarly to the rtPS case. Thus, the ertPS access time is similar to that in the case of UGS except from the relatively rare instances where additional bandwidth needs to be requested.

VII. SIMULATION RESULTS

In this section we will present simulation results that illustrate the scheduling algorithm added overhead. We used the network simulator 2 (ns2) in order to study the uplink scheduling algorithms and their performance. The open source module in [15] was used for simulating Point to Multipoint Point (PMP) WiMAX networks. The simulation parameters are shown in Table V.

A VoIP application with exponential ON and OFF periods has been developed based on the exponential traffic model in ns2. Based on the packet size, rate, the silence interval, and talk interval, we can simulate any VoIP codec that implements VAD with any silence and activity periods distribution.

TABLE V. SIMULATION PARAMETERS

Total number of subcarriers	2048
WiMAX frame duration	5 ms
RTG duration	0.02941 ms
TTG duration	0.02941 ms
OFDMA symbol duration	0.10084 ms
DL subframe	37 OFDMA symbols
UL subframe	12 OFDMA sumbols
Simulation Area	500 m * 500 m

In these simulation experiments, five subscriber stations with different codec types and scheduling algorithms maintain VoIP connections to the same BS for a simulation duration of ten seconds. Estimated performance metrics are the throughput of the scheduling service and the overhead of the bandwidth request headers introduced by each scheduling type.

The bandwidth request overhead along with the data throughput in terms of sent packets are shown in Figure 8. Note that the granularity of the bandwidth overhead measurements is not fine enough to observe variations due to increased packet data size.

The overall system throughput in the downlink and uplink directions averaged over ertPS, rtPS and UGS scheduling connections is shown in Figure 9. We notice that although we used the same scheduling services in both the uplink and downlink directions the downlink traffic is more stable and offers service to the subscribers in steady behavior. On the other hand, the uplink traffic throughput fluctuates according to traffic needs and the allocated bandwidth (depending on the scheduling service used).

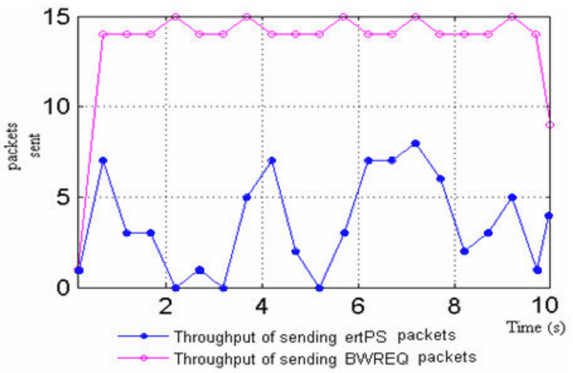

Figure 8. ertPS bandwidth request overhead and exponential traffic throughput

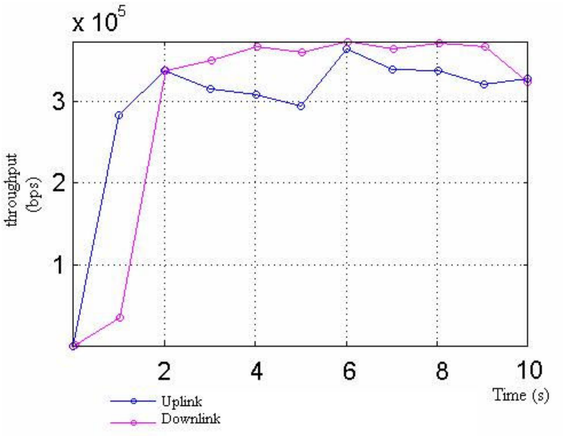

Figure 9. System throughput

VIII. Conclusions

In this paper we calculated the data rate required by VoIP flows over Ethernet and IP convergence sublayers of WiMAX. Calculated these data rates is a prerequisite to estimating the cell capacity and performing call admission control in order to decide if we can accommodate more calls in the system. Moreover, it is important for network dimensioning and planning to know the number of calls that can be served simultaneously. Bandwidth saving techniques were presented and evaluated. Although for Ethernet services the saving in bandwidth achieved by ROHC is higher than that achieved by PHS, ROHC adds much more complexity and is rarely used in practice. Moreover, the subscriber to core network delay was calculated. Finally, the overhead due to bandwidth request messages in ertPS uplink scheduling algorithm was evaluated using ns-2 simulations.

IX. References

[1] IEEE Standard “802.16-2004. Part 16: Air interface for fixed broadband wireless access systems”, June 2004.

[2] IEEE Standard “802.16-2005. Part 16: Air interface for fixed and mobile broadband wireless access systems”, December 2005.

[3] ITU-T standard "G.711/Appendix II : A comfort noise payload definition for ITU-T G.711 use in packet-based multimedia communication systems" 2/2000.

[4] ITU-T standard "G.729 coding of speech at 8 kbps using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP)" 01/2007.

[5] 3GPP TS 26.090 V5.0.0, “AMR Speech Codec; Transcoding functions,” Jun. 2002.

[6] ITU-T Recommendation “G.723.1 -Annex A, General Aspects of Digital Transmission Systems: Dual Rate Speech Coder For Multimedia Communications Transmission at 5.3 and 6.3 kbit/s Annex A: Silence Compression Scheme”, 1996.

[7] ITU-T Recommendation “G.722.1 Annex C: The First ITU-T Superwideband Audio Coder". Claude Lamblin and Catherine Quinquis. IEEE Communications Magazine Oct 2008.

[8] Loutfi Nuaymi, Nabil Bouida, Nabil Lahbil and Philippe Godlewski, "Headers Overhead Estimation, Header Suppression and Header Compression in WiMAX". Wireless and Mobile Computing, Networking and Communications, 2007.

[9] IETF RFC 3095, "Robust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed, " C. Bormann et al, Jul 2001.

[10] Qinqing Zhang, "Performance of Robust Header Compression for VoIP in 1xEV-DO Revision A System", Global Telecommunications Conference, IEEE Volume , Pages:1 – 5 , Issue , Nov. 27 2006.

[11] Howon Lee, Taesoo Kwon, and Dong-Ho Cho, "An Enhanced Uplink Scheduling Algorithm Based on Voice Activity for VoIP Services in IEEE 802.16d/e System". Communication Letters, IEEE, Volume 9, Issue 8 Aug 2005.

[12] Howon Lee, Taesoo Kwon and Dong-Ho Cho, "Extended-rtPS Algorithm for VoIP Services in IEEE 802.16 Systems". IEEE Communications Conference, Volume 5. 2006.

[13] ITU-T Recommendation G.114, “One-way Transmission Time,” May 2003.

[14] Taesoo Kwon, Howon Lee, Sik Choi, Juyeop Kim, and Dong-Ho Cho, "Design and Implementation of a Simulator Based on a Cross-Layer Protocol between MAC and PHY Layers in a WiBro Compatible IEEE 802.16e OFDMA System", Communications Magazine, IEEE Volume 43, Issue 12, Dec. 2005 .

[15] J.Chen et all., "Design and Implementation of WiMAX Module for ns-2 Simulator" in Proceedings of the Workshop on ns-2: the IP network, 2006.