

Novel MIME Type and Extension Based Packet Classification Algorithm in WiMAX

Siddu P. Algur

Department of Computer Science
Rani Chennamma University Belgaum, India

Niharika Kumar

Department of Information Science and Engineering
RNSIT Bangalore, India

Abstract—IEEE 802.16 provides quality of service by providing five different service classes. When a packet reaches the MAC layer, the packet classifier has to classify the packet such that the packet is associated with appropriate QoS. In this paper a packet classification algorithm is proposed that exploits the HTTP header of the application layer to determine the type of data and classify the data. The extension and MIME type (content type) headers are used to classify the packets. Further the algorithm is enhanced by considering the type of user as an additional parameter during packet classification. Simulations are done on variable bit rate video data. Based on the type of video the packets are classified as RTPS or nRTPS thereby providing differentiated QoS. Simulation results reveal that high priority video classified as RTPS traffic receive higher throughput compared to video of lower priority. The delay faced by high priority videos is correspondingly less compared to low priority video.

Keywords—QoS; WiMAX; MAC; Packet Classification; IEEE802.16e

I. INTRODUCTION

IEEE 802.16 also called as WiMAX provides last mile broadband wireless service. WiMAX was initially proposed as a broadband solution for fixed wireless devices. IEEE 802.16e adds mobility support to WiMAX. WiMAX supports broadband speed up to 100 Mbps within a service cell radius of about 5 KM.

WiMAX supports quality of service by providing five different service classes.

The first service class called Unsolicited Grant Services (UGS) is designed to support real time data streams that generate fixed size packets at periodic intervals. Voice over IP without silence suppression is an example for traffic that is categorized as UGS. The second service class called Real Time Polling Services (RTPS) supports real time data streams that generate variable sized packets on periodic basis. For example an MPEG video consists of P Frame, B Frame and I Frame. The third service class also called as Extended Real Time Polling Services (eRTPS) supports real-time service flows that generate variable sized data packets at periodic intervals, for example VoIP with silence suppression. The fourth service class is called as Non Real Time Polling Services (nRTPS). nRTPS supports delay tolerant data streams that generate variable size data packets. An example for one such type of traffic is the file transfer protocol data (FTP). The last service class, also called as Best Effort (BE), supports data streams

which do not require any service level. Ex Web browsing, Email etc.

When a packet is generated by the user it is encapsulated in the TCP/UDP packet which is further enclosed within an IP datagram before it reaches the data link layer. The packet classifier at the data link layer classifies the packet to one of five service classes. WiMAX does not specify any packet classification algorithm. Authors have proposed various packet classification algorithms. In [4] IP packet provides certain bits that can be used by packet classification algorithm. In [5] an IP address based packet classification mechanism is proposed. In this paper an extension and MIME type based packet classification for WiMAX is proposed. These methods exploit the HTTP protocol headers for packet classification. As per available literature an application layer based cross layer packet classification has not been studied before.

This paper is divided into following sections. Section II describes the existing packet classification algorithms. Section III proposes “extension” and “MIME type” based packet classification algorithms. Further a user based improvisation to the algorithm is suggested. Section IV provides a model for the system. Section V delves the simulation and a discussion on the simulation results is undertaken. Section VI concludes the paper.

II. EXISTING PACKET CLASSIFICATION ALGORITHM

IEEE 802.16e does not specify any packet classification algorithm. It is upto the vendor to implement an appropriate packet classification mechanism. One method of packet classification uses the type of service (TOS) field of the IP header. TOS field has 3 bits named delay, throughput and reliability. A value of 0 for the three bits indicates normal delay, normal throughput and normal reliability. A value of 1 for the bits indicates low delay, high throughput and high reliability. MAC layer can use the value of these fields to classify the packets into one of the five service classes. Though this technique provides a method of distinguishing and classifying data into different service classes, it does not provide fine grained support for packet identification and classification. [5] Provides an IP address based packet classification mechanism where the packet classification is performed based on the source and destination IP address. This method of packet classification necessitates the knowledge of ip address of the receiver to result in appropriate packet classification. This is a look up table based packet classification relevant for routers.

III. PROPOSED PACKET CLASSIFICATION ALGORITHM

Data packets enclosed and transmitted using http protocol have HTTP headers to identify the data. HTTP protocol specifies different type of headers to aid the sender and receiver in processing the data. Two headers that are of interest are the "GET" header in the HTTP request and the "Content-Type" header in the response header.

The packet classifier at the MAC layer in WiMAX can use the GET header and the "Content-type" header to classify the packets at the MS and BS as described below.

A. Extension Based Packet Classification

When an MS issues a HTTP GET request for data, it provides the URL of the data. The url contains the location of the data on the server. The URL ends with the type of content being requested. For example, in case of video packets the URL could end with ".mpeg", ".mp4", ".avi" etc. The packet classifier algorithm is described below:

Step 1: Extract the HTTP header from the packet.
Step 2: Extract the GET header from the HTTP headers.
Step 3: Identify the type of request based on the extension (ex: mp4, mpeg etc).
Step 4: Extract the source and destination port number and IP address from the packet.
Step 5: Assign a connection ID representing the flow as RTPS, nRTPS, eRTPS or BE.
Step 6: create a tuple containing six parameters as below:
<extension, connection ID, source IP, destination IP, source port, destination port>
Step 7: place the packet in the appropriate queue (i.e RTPS, nRTPS, eRTPS or BE).

The above packet classification shall be done for the first packet of the service flow. Subsequent packets of the service flow may not contain the get header. So, when subsequent packets arrive they shall contain the source IP, destination IP, source port and destination port. These four parameters in the packet shall be compared with the four parameters of the tuple and when matched the corresponding connection ID is retrieved. This connection ID represents the service flow to which the packet shall be forwarded (RTPS, nRTPS, eRTPS or BE). Hence the packet classification algorithm is able to classify the packets based on the extension of the request.

B. Content Type or MIME type based packet classification.

The response for an HTTP request contains the content requested by the user. The response packets also contain the HTTP headers. Along with the different headers like Cache Control, cookies, Age, Date, the server sends the Content type of the data. For example, if the video is of type avi the content type shall be video/x-msvideo. For MPEG video the mime type shall be video/mpeg and for quick time videos the mime type is video/quicktime. When the packets reach the base station (BS), BS can decode the MIME type to identify the content type and

classify the packets accordingly. The algorithm for packet classification based on the MIME type is as below:

Step 1: Extract the HTTP header from the response packet.
Step 2: Extract the Content type / MIME type header from the HTTP headers.
Step 3: Identify the type of content based on the content type/ MIME type (video/mpeg, video/quicktime, video/x-msvideoetc).
Step 4: Extract the source and destination port number and IP address from the packet.
Step 5: Assign a connection ID representing the flow as RTPS, nRTPS, eRTPS or BE.
Step 6: create a tuple containing six parameters as below:
<MIME type, connection ID, source IP, destination IP, source port, destination port>
Step 7: place the packet in the appropriate queue (i.e RTPS, nRTPS, eRTPS or BE).

This classification and tuple generation is done for the first response packet. For all subsequent response packets the source IP, destination IP, source port and destination port details from the packet is mapped with all the tuples. The matching tuple is extracted and the connection ID is obtained from the tuple. The packet is then forwarded to the queue corresponding to the connection ID (i.e RTPS, eRTPS, nRTPS or BE). Hence the packet gets classified as one of the service class packet based on the MIME type of the packet.

1) Advantage

MIME type and extension based classification provides the advantage of classifying the packets based on the extension and MIME type there by giving a finer control at packet classification. In the general packet classification mechanism all video packets are classified as RTPS traffic. By using the MIME type based packet classification, specific videos can be classified as high priority RTPS traffic and other videos can be classified as low priority nRTPS or BE packets. This lets the network operators to selectively degrade the quality of service for certain types of video traffic during dynamic network conditions. When the network is heavily loaded the high bandwidth MPEG packets can be downgraded to nRTPS/BE and bandwidth efficient video packets like mp4 can be treated as RTPS. This lets the BS serve many users at the same time. When the network condition improves, the MPEG videos can again be classified as RTPS packets to improve the QoS for MPEG users.

C. User oriented MIME type and extension based packet classification

MIME type and extension based bandwidth allocation can be further specialized to selectively classify the same MIME type data differently for different types of users, i.e. for certain users the MPEG videos can be classified as RTPS packets and for some other users the MPEG videos can be classified as nRTPS data. This lets network operators provide graded

quality of service based on the type of user and the type of traffic generated for/by the user.

The network could grade the users as priority users and regular users. Priority users could be users who pay more or utilize the network more often. Such users generate more revenue and also ensure a higher utilization factor for the network operators. Such users can be classified as priority users. Regular users could be the users who pay the regular network utilization fee.

The operator can maintain a mapping table of the MAC address of the user and the priority value associated with the user.

So, when a video/mpeg packet is being classified as RTPS, nRTPS or BE packet the BS uses the following algorithm to classify the packet. The request packet obtained from MS and the response packet being sent to MS together contribute to the packet classification algorithm. This two stage process is described below.

1) Stage 1: BS uses extension type and user priority for tuple creation

Step 1: Receive the HTTP request from MS.
Step 2: Extract the MAC address of user from MAC header
Step 3: From the priority table extract the priority associated with user (regular user or priority user)
Step 4: Extract the extension from the GET request to identify the type of data.
Step 5: Based on the extension type and the priority of the user, associate the request to a particular connection. Ex: high priority user generating MPEG packet is associated with RTPS connection and regular user generating MPEG packet is associated with nRTPS/BE connection.
Step 6: Generate a tuple containing seven parameters. One of the parameters (content type is null for now). The tuple looks as below:
<extension, content_type_null, connection ID, source IP, destination IP, source port, destination port>

2) Stage 2: BS uses the tuple for packet classification

The GET request is then forwarded to the network by the BS. When a response from the network arrives, the BS shall extract the source IP, destination IP, source port, destination port to identify the tuple. The tuple is then updated with the content type to form the complete tuple.

<extension, content_type, connection ID, source IP, destination IP, source port, destination port>

From the tuple extract the connection ID and place the packet in the queue corresponding to the connection ID (RTPS, nRTPS, eRTPS, BE). Hence a combination of extension content type and the type of user is used to classify the packet appropriately.

IV. SYSTEM MODELING

Let the system contain users that generate video traffic of different types. Some users generate lossless high resolution MPEG video and other users generate lossy mp4 video. The amount of data generated per video frame for an MPEG video shall be much higher compared to a MP4 video. The packet classifier shall read the content type of the data to determine if it is video/mpeg or video/mp4.

All packets of video/mpeg are assigned to the connection associated with the QoS for nRTPS traffic and all packets of video/mp4 are assigned to the connection associated with the QoS RTPS traffic.

As the MPEG packets keep arriving the packets are queued in the nRTPS queue and all mp4 packets get queued as RTPS traffic.

The base station allocates unicast request opportunities to RTPS queue and unicast/broadcast request opportunities to nRTPS queues to request for bandwidth. Each of the RTPS and nRTPS queues request for bandwidth by sending bandwidth request headers.

When the bandwidth request header reaches BS the base station allocates bandwidth by scanning through the bandwidth request headers. BS shall allocate bandwidth in the following order:

UGS >> RTPS >>eRTPS>> NRTPS >>BE

Since the RTPS connections are allocated bandwidth first, MP4 videos get higher share of bandwidth compared to MPEG videos. This result in better quality of service for mp4 traffic compared to mpeg traffic. The bandwidth allocation is performed using the following criteria:

$$BWAllot_{mp4} = \begin{cases} BWReq_{mp4} & \text{if } BWReq_{mp4} < MRTR, BWAvail \\ MRTR & \text{if } MRTR < BWReq_{mp4} < BWAvail \\ BWAvail & \text{if } BWAvail < BWReq_{mp4} < MRTR \end{cases} \quad (1)$$

Where MRTR is the minimum reserve traffic rate.

After an mp4 video is allocated bandwidth, the leftover bandwidth is calculated as:

$$BWAvail = BWAvail - \sum_{i=1}^n BWAllot_{mp4}^i \quad (2)$$

Where n represents the number of mp4 connections that have been allocated bandwidth in the current frame.

Once all the mp4 connections are allotted bandwidth, the connections for MPEG videos are allocated bandwidth as below:

$$BWAllot_{mpeg} = \begin{cases} BWReq_{mpeg} & \text{if } BWReq_{mpeg} < MRTR, BWAvail \\ MRTR & \text{if } MRTR < BWReq_{mpeg} < BWAvail \\ BWAvail & \text{if } BWAvail < BWReq_{mpeg} < MRTR \end{cases} \quad (3)$$

The available bandwidth is calculated as:

$$BWAvail = BWAvail - \sum_{k=1}^p BWAllot_{mp4}^k \quad (4)$$

Where “p” represents the number of MPEG videos that have already been allocated bandwidth

The term BW_{avail} represents the available bandwidth after allocating bandwidth to “n” mp4 videos followed by p MPEG videos, i.e.

$$BW_{avail} = BW_{avail} - \sum_{i=1}^n BW_{Allot}_{mp4}^i - \sum_{k=1}^p BW_{Allot}_{mp4}^k \quad (5)$$

V. RESULTS AND DISCUSSION

Simulations were carried out on MATLAB. Simulation parameters are given in the table

TABLE I. SIMULATION PARAMETERS

Parameter	Value
Uplink Bandwidth	10 Mbps
Traffic Type	RTPS high priority data
	nRTPS low priority data
Packet Arrival Pattern	Variable bit rate
Average arrival rate	50 kbps

Simulations were carried out to find the improvement in throughput. Figure 1 shows the throughput results.

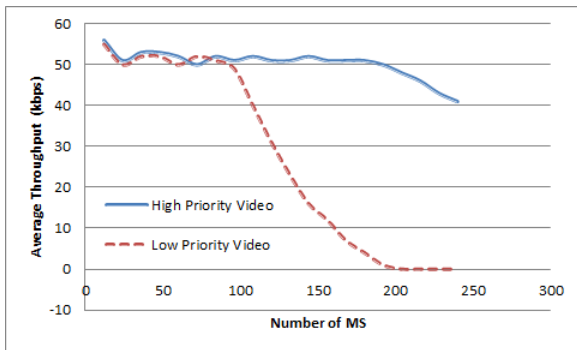


Fig. 1. Average throughput for high priority and low priority video data..

Figure 1 shows that initially there is sufficient bandwidth to support all the MS. Hence both high priority and low priority data show similar throughput results. As the number of MS increases there isn't sufficient bandwidth to support all the traffic since high priority video is classified as RTPS traffic, high priority video gets allocated bandwidth. The left over bandwidth is allocated to the low priority video which is categorized as nRTPS traffic.

As the packets keep accumulating, MS has to decide whether to transmit the delayed data or to drop the packets. In case of video packets delayed transmission results in jittered video which is not suitable for viewing. Hence all delayed packets shall be dropped. Figure 2 shows the simulation results for the data drop in case of high priority and low priority video traffic.

Increase in the number of MS results in more MS contending for limited bandwidth. Since high priority video is

classified as RTPS traffic and allotted bandwidth before low priority video, high priority data experiences lower data drop compared to low priority video.

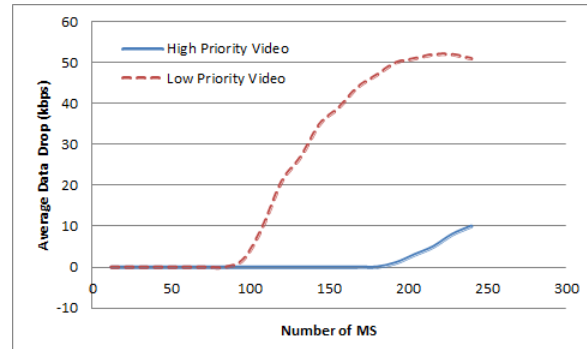


Fig. 2. Average data drop for high priority and low priority video data.

Simulations were carried out to observe the impact of priority of the user and the priority of the data traffic. The order of classification can be as given in Table II.

Data Type	Classification Type
High priority user generating high priority video (HUHT)	RTPS
Low priority user generating high priority video (LUHT)	eRTPS
High priority user generating low priority video (HULT)	nRTPS
Low priority user generating low priority video (LULT)	BE

Figure 3 shows the simulation results for average throughput.

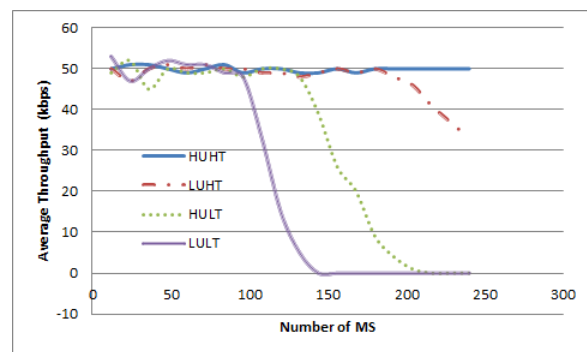


Fig. 3. Average throughput for different classification methods v/s number of MS.

Since HUHT traffic is classified as RTPS traffic, HUHT is allocated bandwidth first then followed by LUHT which is followed by HULT and LULT. As the number of MS increases there isn't bandwidth to support all the video traffic. Hence lower priority traffic is dropped to maintain the throughput of higher priority traffic.

Figure 4 shows the corresponding data drop for the four types of classification.

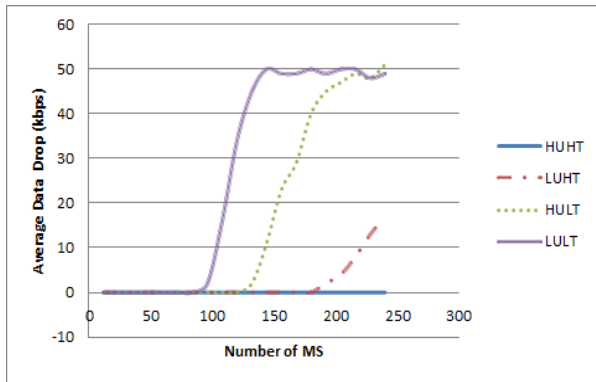


Fig. 4. Average data drop for different classification v/s number of MS.

VI. CONCLUSION

This paper proposes a packet classification algorithm in WiMAX that utilizes the GET and Content type/ MIME type

headers of HTTP. By exploiting the application layer headers a fine grained packet classification mechanism can be achieved which in-turn leads to a controlled and coordinated quality of service. Additionally the algorithm is extended to add user based packet classification which provides the necessary flexibility to the network providers. Simulation results show a graded quality of service for video traffic.

REFERENCES

- [1] Air Interface for Fixed Broadband Wireless Access Systems, IEEE Std. 802.16, 2004.
- [2] Air Interface for Mobile Broadband Wireless Access Systems, IEEE Std. 802.16e, 2005.
- [3] IEEE Std. 802.16-2009, IEEE standard for local and metropolitan area networks part 16: air interface for broadband wireless access systems.
- [4] RFC 791 <http://www.ietf.org/rfc/rfc791.txt>.
- [5] P. Gupta, N. McKeown, Packet classification on multiple fields, Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication, pp. 147-160, 1999.
- [6] Matlab <http://www.mathworks.in/products/matlab/>