

Hop Aware Scheduling Algorithm for TCP Performance Enhancement in Multi Hop Relay WiMAX

F. R. Ismael^{a*}, S. K. Syed Yusof^a, M. Abbas^b, N. Fisal^a

^aCoE Telecommunication Technology, Universiti Teknologi Malaysia (UTM), Johor, 81310 UTM Skudai, Malaysia

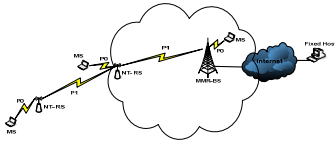
^bHead of Wireless Communication Cluster, MIMOS Berhad

*Corresponding author: fathi_sml@fkegraduate.utm.my

Article history

Received: 8 March 2012
Received in revised form: 10 April 2012
Accepted: 18 July 2012

Graphical abstract



Abstract

Mobile Multi-hop Relay (MMR) WiMAX networks allow the number of hops between the user and the MMR-Bs to be more than two hops when the non transparent relay station is used. Transmission Control Protocol (TCP) protocol is optimized to be used in wired networks where the main cause of loss is the congestion. TCP throughput is highly affected by packet losses or timeout occurrences because of its congestion control mechanism. Upon the occurrence of timeout, TCP decreases the transmission window which results in throughput degradation, even if such losses are not due to congestion. This paper presents link layer scheduling algorithm to enhance the TCP in IEEE 802.16j MMR WiMAX networks. The aim is to reduce the Retransmission Timeout (RTO) occurrences and enhance the throughput. The results showed that the hop aware scheduling algorithm gives higher throughput, and less timeout occurrence for different BER values.

Keywords: MMR WiMAX; IEEE 802.16j; TCP loss probability; TCP throughput; hop aware scheduling algorithm

© 2012 Penerbit UTM Press. All rights reserved.

1.0 INTRODUCTION

The single hop WiMAX products such as IEEE802.16e have limited coverage and provide poor Quality-of-Service (QoS) for indoor users as well as for users at cell boundaries. To address this issue, a new Relay Station (RS) had defined by the Multi-hop Relay (MR) task group. The RS can be used as an extension to the Base Station (BS) and relay traffic between the BS and the Mobile Station (MS). As a result, the channel quality received by the MS becomes better, and the overall network capacity will increase dramatically [1][2][3]. Additionally, RS significantly reduces the installation and operation cost compared with using micro-BS to cover these areas [4]. The relay operations are classified in two different modes: transparent relay (TR) and Non-Transparent Relay (NTR): TR is an element that is located in the network between the MMR-Bs and the MS, without the MS being aware of its existence. It does not transmit any control data. The MS receives directly all the control data such as preamble, Frame Control Header (FCH), Downlink (DL) MAP and Uplink (UL) MAP directly from the MMR-Bs. The TR usually does not extend coverage and limited to only two hops; its main function is channel quality improvement that as a result increases the link capacity [3]. NTR is a network element that the MS is totally aware of it. The MS which is IEEE 802.16e compatible is not aware of any relaying operation, from the MS point of view the

NTR is its conventional serving BS. So, the NTR should support most of the capabilities of a plain 802.16e BS like sending frame control data that includes preamble, UL/DL MAP, FCH. The NTR serves MSs that are beyond reach of the MMR-Bs. As so the NTR is perfect tool for achieving extension of cell and coverage and can support more than two hops. In addition, it can also act as a capacity enhancer because of significant improvement in the channel quality of both access and relay links [3]. However, adding the RS increases the complexity and delay of relaying information both signaling and user data across multiple hops. In addition, the functionalities of the BS should be extended in order to support incorporation of relays into the network. BS which incorporates these new functions is called a Mobile Multi-hop Relay Base Station (MMR-Bs). The MMR WiMAX can be considered as one of the best choices to connect users to the Internet in which it's required to ensure reliable transmission of data. The reliability can be achieved by using Transmission Control Protocol (TCP) in transport layer and Automatic Repeat reQuest (ARQ) in the link layer. TCP is used to carry a significant amount of elastic Internet traffic which needs reliability. It has been designed and optimized to be used in wired networks where the cause of the packet loss is only congestion, so it reacts to all types of losses as congestion loss using congestion control algorithms. TCP uses slow start, congestion avoidance,

fast retransmit, and fast recovery algorithms together to avoid and handle the congestion [5][6].

The paper is organized as follows: Section 2.0 introduces different categories of the TCP performance enhancements mechanisms. The proposed scheduling algorithm for TCP performance improvement stated in Section 3.0. The results and discussions of the work are given in Section 4.0. The conclusion is stated in Section 5.0.

2.0 RELATED WORKS

TCP has been tuned for traditional networks with wired links and stationary hosts. These protocols assume congestion in the network to be the primary cause for packet losses and unusual delay. When packets are lost in networks for reasons other than congestion, the TCP responds to it by invoking its congestion window, which results in an unnecessary reduction in end-to-end throughput, and degrades the overall network performance. Communication over wireless links is often characterized by high bit-error rates, and intermittent connectivity due to handoffs. TCP performance in such networks suffers from significant throughput degradation and very high interactive delays. There are some solutions proposed in the literature for different types of wireless networks, which are classified into three categories: end-to-end protocols, and split connection protocols, link layer protocols [7][8].

2.1 End to End Protocols

In the end-to-end approach, the TCP sender attempts to handle the losses by modifications to TCP in order to improve the performance on wireless links. The modifications allow the TCP to differentiate losses due to congestion from that due to increased error rate, or handoffs. Therefore this category maintains the end-to-end semantics of TCP. Nevertheless, their improvement of performance over regular TCP in wireless networks is expected to be limited, since they were not designed specifically to overcome the problems of the wireless networks. Furthermore, end-to-end protocols require modifications to TCP at the fixed host code and thus they do not satisfy compatibility [9][10].

2.2 Split-connection Protocols

This scheme split the TCP connection between the mobile host and the fixed host into two separate connections, a wired connection between the fixed host and the base station, and a wireless connection between the base station and the mobile host. They shield the mobility and wireless problems from the fixed host. The major advantage of split-connection protocols is that they provide backward compatibility with the existing wired network protocols, and they can handle disconnections efficiently. Their disadvantage is that they might not maintain the end-to-end semantics of TCP, which can cause major problems if not handled properly [11][12].

2.3 Link Layer Protocols

The link layer protocols maintain the end-to-end semantics of TCP. This approach tries to increase the quality of the lossy wireless link. Thus, it hides the characteristics of the wireless link from the transport layer and tries to solve the problem at the link layer by using error detection and correction techniques, error control techniques and scheduling algorithms. Therefore, link

layer protocols perform well in environments with high bit error rates. The reliability of the wireless link is increased by reducing the bit error rate using techniques such as Forward Error Correction (FEC) and Automatic Repeat reQuest (ARQ)[13][14]. Scheduling is a sequence of allocating time slots to users, where each possible transmission is assigned a time slot. In addition, it is responsible for the decision of which packets should be selected to transmit. WiMAX supports Unsolicited Grant Service (UGS), real-time Polling Service (rtPS), non real-time Polling Service (nrtPS) and Best Effort (BE). The QoS of these services can be achieved through scheduling schemes considering their priorities and their QoS requirements [15]. File Transfer Protocol (FTP) application which is considered as nrtPS in the WiMAX network uses Transmission Control Protocol (TCP) to ensure the reliability of data transferring. Most of the works done in the scheduling concerns the allocation of bandwidth or resources among different services in order to achieve the required QoS. However, no work has been done considering the hop level in the scheduling of nrtPS [16]. In the next section, we will propose hop aware scheduling algorithm for TCP performance enhancement.

3.0 PROPOSED SCHEDULING ALGORITHM FOR TCP IMPROVEMENT

Scheduling can be considered as one of the optimization stages of the network and it is left to the decision of the operator. Thus the selected scheduler should consider the network architecture in which it's implemented and the quality of service requirement of the end user. In the next sections, the network architecture and the proposed scheduling algorithm will be discussed.

3.1 MMR WiMAX Network Model and Assumptions

Figure 1 shows the MMR WiMAX network architecture consists of one MMR-BS connected to the Internet through backbone network, two Non Transparent RS, and three Mobile stations served either directly by the MMR-BS or one of the RSs. The direct link between the MS and the MMR-BS or RS is referred to as an access link, while the backhaul link between the RS and the MMR-BS referred to as the relay link. In this scenario the ARQ packet loss probability in the access link is P_0 , and in the relay link is P_1 . Here, we assume the same link BER for the access link and relay link. In addition, the users in all the hops always have data to download.

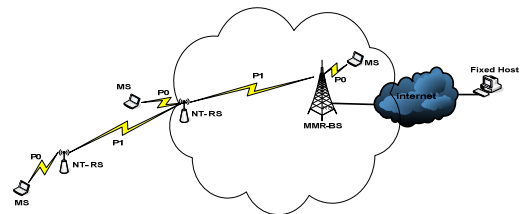


Figure 1 MMR WiMAX network topology

3.2 HOP Aware Scheduling Algorithm

Hop aware scheduling algorithm is an algorithm to share the available resources between the users in different hop levels. The main benefit of this algorithm is that it prevents the users in the first hop which have short delay time to occupy the available

resources and cause starvation to the multi-hop users. In this algorithm, the downlinks TCP Service Data Units (SDUs) that arrive at the MMR-BS from the Internet are fragmented to construct smaller ARQ packets. These ARQ packets going to the same hop level are stored in separate queues as shown in Figure 2. The packets within any queue are rearranged depending on their retransmission timeout that is determined using the equations from (1) to (3) and the number of retransmissions (N_{ij}) as in the equation (4). The TCP Retransmission Timeout (RTO) which calculated in the standard depending on the value of RTT as follow:

$$S(k) = (1 - \alpha) * S(k - 1) + \alpha * T(k) \quad (1)$$

$$V(k) = (1 - \beta) * V(k - 1) + \beta * |T(k) - S(k)| \quad (2)$$

$$RTO = S(k) + m * V(k) \quad (3)$$

$$W_{ij} = RTO_{ij} * 10^4 + N_{ij} \quad (4)$$

Where, $S(k)$ is the smoothed mean deviation of the Round Trip Time (RTT), $V(k)$ is the smoothed average of RTT , RTO is Retransmission out value, α is an exponential smoothing parameter ($\alpha = 1/8$), β is smoothing parameter ($\beta = 1/4$), m is a constant ($m = 4$), N_{ij} is the number of retransmissions and W_{ij} is the weight of the corresponding packet.

The algorithm gives the same weight or number of packets to all hops, beginning from the third then second hop packets and lastly, the first hop users. This procedure will continue unless there is an emergency case such as the RTO of one packet is about to expire or one of the queues is about to be full. If one packet has $RTO \leq 2 * RTT$, the forwarding procedure will be stopped and the packet is immediately sent. On the other hand, if one of the queues reaches 90% of its maximum capacity three packets will be sent from that queue out of normal round. The dropping criteria for this algorithm are the expiration of the RTO of the packet or a given queue is full when the packet arrives. If one packet is discarded all the ARQ packets belonging to the same TCP Service Data Unit should be dropped as well.

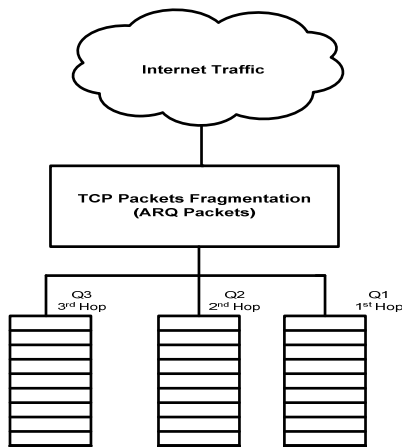


Figure 2 Hop Aware Queuing for MMR Network

3.3 The Pseudo Code of the Forwarding Algorithm

- 1: IF the hop number $H=1$
 $P_{ij} \rightarrow Q1$ % store the packet in the queue1
- 2: IF the hop number $H=2$
 $P_{ij} \rightarrow Q2$ % store the packet in the queue2

- 3: IF the hop number $H=3$
 $P_{ij} \rightarrow Q3$ % store the packet in the queue3
- 4: Else Drop the Packet
- 5: Sort Q1: $W_{i1j1} < W_{i2j2} < \dots < W_{injn}$ % rearranges the packets in Q1 depending on their weights
- 6: Sort Q2: $W_{i1j1} < W_{i2j2} < \dots < W_{injn}$ % rearranges the packets in Q2 depending on their weights
- 7: Sort Q3: $W_{i1j1} < W_{i2j2} < \dots < W_{injn}$ % rearranges the packets in Q3 depending on their weights
- 8: IF RTO of any packet in Q1 or Q2 or Q3 is $RTO \leq 2 * RTT$
 Transmit the corresponding packet
- 9: Else IF Q1 or Q2 or Q3 is 90% Full
 Transmit the urgent packets from the corresponding queue
- 10: Else IF Normal case exists
 Transmit one third of ARQ window size from the Q1
 Transmit one third of ARQ window size from the Q2
 Transmit one third of ARQ window size from the Q3
- 11: Continue the Algorithm

4.0 SIMULATION PARAMETERS, SCENARIO AND RESULT ANALYSIS

In order to test and evaluate the proposed hop aware scheduling algorithm, the simulator of MMR WiMAX Networks is developed. In which the functionality of TCP Reno and ARQ are implemented. In the following sections, the simulation scenarios, parameters and results are discussed.

4.1 Simulation Parameters and Scenario

Two simulation scenarios are conducted, the first one using the proposed hop aware scheduling algorithm (HASA) and the other using conventional First Come First Served (FCFS) [17]. In the first scenario, when the TCP segment arrived at the MMR-BS is fragmented into four smaller packets and stored in the corresponding hop queue and then scheduled using the algorithm mentioned in Section 3.2. In the other scenario the coming TCP segment is stored in one queue after fragmented to four fragments and then the First Come First Served (FCFS) algorithm is used to forward the packets. Simulation parameters are summarized in Table 1. The successfully delivered and acknowledged segments are calculated and then TCP throughputs for individual user and all users are plotted against the throughput from the conventional FCFS algorithm. Furthermore, the number of timeouts that occurs during the simulation time is determined and plotted for both cases. Lastly, the congestion window growths for the two scenarios are plotted for different BER values.

4.2 Results Analysis

In this section, different simulation results are discussed. Figure 3.1 shows the TCP throughput for individual users for the Hop Aware Scheduling Algorithm (HASA) (first three columns) and the First Come First Served (FCFS) algorithm (Last three columns). Meanwhile, Figure 3.2 shows the total throughput for

the users in all the networks. From these two figures we can conclude that the hop aware scheduling algorithm gives higher throughput up to 30% for different channel state.

Table 1 Simulation factors and level

Factors	Levels
TCP segment size (Bytes)	1024
ARQ packet size (Bytes)	256
Hop queue size (Bytes)	25600
FIFO queue size (Bytes)	76800
Frame Size (ms)	20
Number of hops	3
ssthresh(Bytes)	65536
Receiver Advertised Window (Bytes)	81920
Simulation Scenarios	2

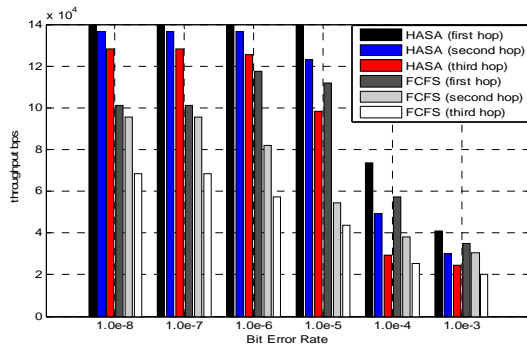


Figure 3.1 Individual user Throughput comparison

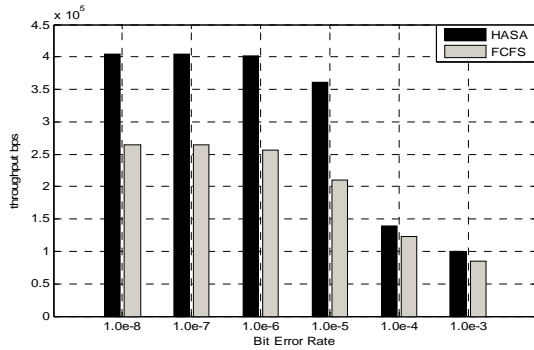


Figure 3.2 Total Throughput comparisons

Figure 4.1 shows the number of timeouts for individual users for the hop aware scheduling algorithm (HASA) (first three columns) and the FCFS algorithm (Last three columns). While the Figure 4.2 shows the total timeouts for the users in all the networks. From these two figures we can see that the hop aware scheduling algorithm is free of timeouts occurrence for the BER less than 10^{-6} and less than or equal two timeouts per individual user for the BER higher than 10^{-6} . It can be considered very little as compared with that of the FCFS algorithm which experience up to 5 timeouts.

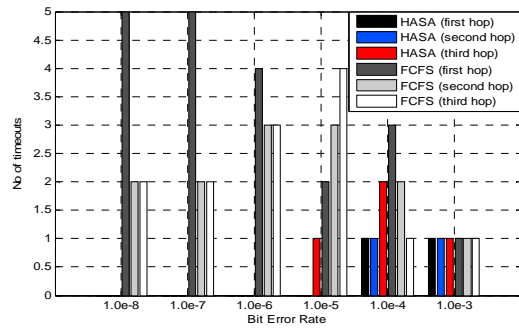


Figure 4.1 Individual Timeout Comparison

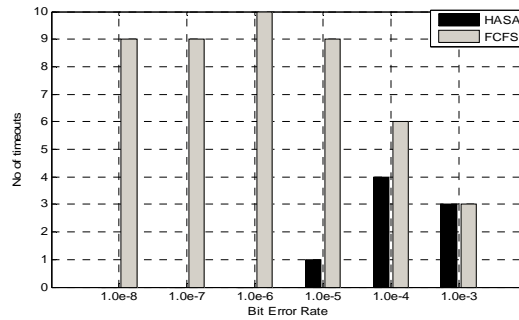


Figure 4.2 Total Timeout Comparison

The congestion windows of the same simulation scenarios for different BER are plotted in Figures 5.1, 5.2 and 5.3 respectively. The CWND of the proposed hop aware scheduling algorithm (HASA) is developed up to 50 packets for the BER less than or equal to 10^{-5} as shown in Figures 5.1 and 5.2. However, for the BER of 10^{-3} the CWND window is limited to less than 12 packets due to high error rate and hence long transmission time. However the CWNDs of the conventional FCFS algorithm are limited to small number of packets. The reason for this is that the hop aware scheduling algorithm experienced less timeouts as compared with the FCFS algorithm.

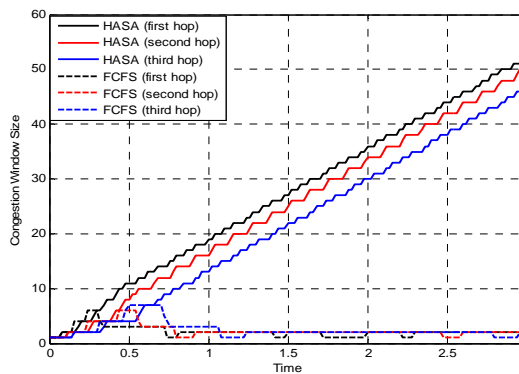


Figure 5.1 Comparison of individual CWND for BER= 10^{-8}

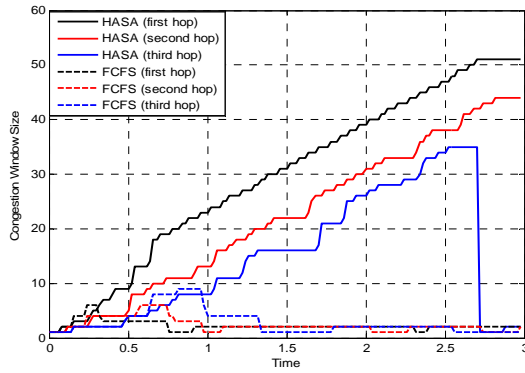


Figure 5.2 Comparison of individual CWND for $BER=10^{-5}$

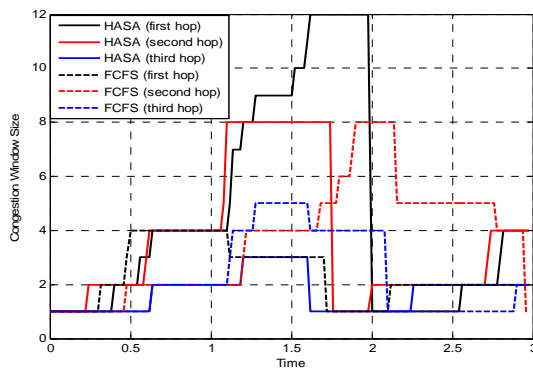


Figure 5.3 Comparison of individual CWND for $BER=10^{-3}$

5.0 CONCLUSION AND FUTURE WORKS

In this paper, we presented the link layer scheduling algorithm to enhance the TCP in IEEE 802.16j MMR WiMAX networks. The hop aware scheduling algorithm improves the TCP traffic performance in MMR WiMAX network and result in less number of timeout and higher throughput. However, the proposed algorithm does not consider the fairness among the users in different hop levels. Thus, we will consider this issue in the future works.

Acknowledgement

In this paper, we presented the link layer scheduling algorithm to enhance the TCP in IEEE 802.16j MMR WiMAX networks. The hop aware scheduling algorithm improves the TCP traffic

performance in MMR WiMAX network and result in less number of timeout and higher throughput. However, the proposed algorithm does not consider the fairness among the users in different hop levels. Thus, we will consider this issue in the future works.

References

- (1) Masato Okuda, Chenxi Zhu, and Dorin Viorel. 2008. Multihop relay extension for WiMAX Networks overview and benefits of IEEE 802.16j standard. *FUJITSU Sci. Tech. J.* 44(3): 292–302.
- (2) Vasken Genc, Sean Murphy, Yang Yu, and John Murphy. 2008. IEEE 802.16j Relay Based Wireless Access Networks: An overview. *IEEE wireless communications*.
- (3) Steven W. Peters and Robert W. Heath. 2009. The future of WiMAX: Multihop relaying with IEEE 802.16j. *IEEE communication magazine*. January.
- (4) Bharathi Upase, and Mythri Hunukumbure. 2008 Dimensioning and cost analysis of Multihop relay enabled WiMAX networks. *FUJITSU Sci. Tech. J.* 44(3): 303–317.
- (5) M. Allman, V. Paxson, and W. Stevens. 2001. TCP Congestion Control. Network Working Group, NASA Glenn/Sterling Software.
- (6) Carlo Cainin, and Rosario Firrincieli. 2004. TCP Hybla: a TCP enhancement for heterogeneous networks. *International Journal of Satellite Communications and Networking*. 22: 547–566.
- (7) ShiNing Li, JiPing Frang, and Fan Yu. 2004. On analysis and Comparison Performance of TCP in Wireless Networks. *Springer ApWeb 2004, LNCS 3007*. 341–352.
- (8) Hari Balakrishnan, Venkata N. Padmanabhan, Srinivasan Seshan, and Randy H. Katz. 1997. A Comparison of Mechanisms for Improving TCP Performance over Wireless Links. *IEEE/ACM Transactions on Networking*. 5(6): December.
- (9) Muhammad Saeed Akbar, Syed Zubair Ahmed, Muhammad Abdur Qadir. 2008. Performance Optimization of Transmission Control Protocol in Heterogeneous Wireless Network during Mobility. *IJCSNS International Journal of Computer Science and Network Security*. 8(8): August.
- (10) Claudio Casetti, Mario Gerla, Saverio Mascolo, M.Y. Sanadidi and Ren Wang. 2002. TCP Westwood: End-to-End Congestion Control for Wired/Wireless Networks. *Kluwer Academic Publishers, Manufactured in the Netherlands, Wireless Networks* 8. 467–479.
- (11) Milosh Ivanovich and Philip W. Bickerdike. 2008. Telstra Corporation On TCP Performance Enhancing Proxies in a Wireless Environment. *IEEE Communications Magazine*. September.
- (12) Liang Hu and Lars Dittmann. 2009. Optimizing TCP Performance over UMTS with Split TCP Proxy. *ICST Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*.
- (13) Dzmityri Kliazovich, Fabrizio Granelli, Mario Gerla. 2007. Performance improvement in wireless networks using cross-layer ARQ. *Elsevier Computer Networks*. 51: 4396–4411.
- (14) Yikang Xiang, Tao Jin, Jijun Luo, Egon Schulz, and Carmelita G. 2008. User Plane Protocol Optimization in Cellular Networks with Decode-and-Forward Type of Relay. 978-1-4244-1645-5/08, IEEE.
- (15) Y. Li, et al. 2010. QoS-aware fair packet scheduling in IEEE 802.16 wireless mesh networks. *International Journal of Communication Systems*. 23: 901–917.
- (16) Carlos G. Bilich. TCP over WiMAX Networks. Wireless Access networks project number II, *University of Trento, Trento*, TN 38100 italy.
- (17) A. Belghith and L. Nuaymi. 2009. Scheduling Techniques for WiMAX," *Current Technology Development of WiMAX Systems*. 61–84.