Optimization of TCP/IP over 802.11 Wireless Networks in Home Environment

Toni Janevski[1] and Ivan Petrov[2]

[1] University "Sv. Kiril i Metodij", Faculty of Electrical Engineering and Information Technologies, Karpos 2 bb, 1000 Skopje, Macedonia
tonij@feit.ukim.edu.mk
[2] Macedonian Telecom, Key Customers Business Centre, Orce Nikolov bb, 1000 Skopje, Macedonia
ivan.petrov@telekom.mk

Abstract. Internet connectivity today is based mainly on TCP/IP protocol suite. Performance of the Internet transport protocols may significantly degrade when end to end connection includes wireless links where packets delays and losses are caused by mobility handoffs and transmission errors. In this paper we perform analysis of the achievable throughput for different TCP versions, such as TCP Tahoe, TCP Reno, TCP New Reno, TCP Vegas and TCP SACK, in IEEE 802.11 wireless networks. The analysis showed the strong impact of Medium Access Control parameters, such as number of retransmissions and interface queue length in 802.11 networks on the obtained throughput.

Keywords: TCP, Throughput, Transmission Protocols, Wireless Network.

1 Introduction

Rapid increase in number of active wireless hot spots is providing people to be connected in almost every building and every street, in their homes as well as in their offices. Large amount of nomadic users are using the IEEE 802.11a/b/g/n wireless access technology for checking emails, web surfing, video/audio streaming, and P2P file sharing, either in infrastructure mode (when connected to Internet) or in ad-hoc mode (in office, home or personal environment). Mobile users are starting to use the benefit of these applications as well. Also, such wireless scenario will be applicable in the home environment for interconnecting communication electronic devices, entertainment devices, various sensors in the home or in the car etc. In all such applications, most of the traffic will be based on TCP/IP protocol suite, either for data traffic i.e. non-real time traffic, such as www, ftp, email etc., or for real time traffic, such as voice over IP and IPTV.

It is well known that TCP/IP protocol stack is the most widely used one in the today's Internet world. However, its parameters were carefully tuned in order to maximize its performance on wired networks where packet delays and losses are caused by congestion [1-5]. In the wireless networks, delays and losses are mainly caused by mobility handoffs and transmission errors due to bad wireless channel conditions. With the recent developments in mobile wireless networking, the performance of the

R. Mehmood et al. (Eds.): EuropeComm 2009, LNICST 16, pp. 240–251, 2009.

Internet transport protocols in mobile wireless environment is becoming more important. We should mention that the protocols for wireless access have been designed in order to maximize the utilization of the wireless channel for web browsing and file downloading applications in an environment with restricted mobility, which is the main reason why the buffers and the local Medium Access Control (MAC) retransmissions are tuned in a way to maximize the throughput and the reliability for this kind of applications. In this context it is necessary to define and fine tune the technical standards in order to guarantee full interoperability between different digital applications in wireless environment as well as to provide proper wireless/wired interconnection. We focus our attention of the impact of diverse MAC layer and buffer settings of IEEE 802.11g wireless access technology over the Internet native transport protocol suite during the distribution of multimedia applications in realistic outdoor static as well as mobile multimedia scenario.

The paper is organized as follows: Section 2 gives brief overview of the transport protocols, discusses some related work and motivates the need for our approach. It briefly describes the 802.11 MAC protocol. Section 3 describes our simulation scenario and section 4 presents the simulation results. Section 5 concludes the paper.

2 Transport Protocols

Applications can be grouped in two major classes: downloading (using TCP) and real-time (using UDP). The first class is using reliable data transfer while the second class is based on quick delivery of packets. The performances that are measured by booth classes of applications are completely different. The first class is measuring the performance in terms of how much time is required to have the whole file transferred that is different from the second one where the performances are measured in terms of percentage of packets that reach the destination within a certain time interval. We can say that FTP, HTTP, SMTP are applications that belong to the first class and that the interactive on-line games, real-time IPTV, video/audio chatting, represent examples of applications that are part of the second class. So, we can distinguish the downloading and real-time applications by the transport protocol: TCP or UDP. The TCP protocol guarantees the reliable and in order delivery of every packet sent by using the congestion control functionality. Every TCP flow probes the link with higher and higher data rates eventually filling up the channel. We can be sure that the packets will be queued at the buffer associated with the bottleneck of the link until it overflows causing packet losses. In this situation TCP retransmits the lost packets, and halves its sending rate to diminish the congestion level. Finally, the regular increase of the sending rate is reestablished and so forth. This is not the case with the UDP transport protocol which is much simpler then TCP because packets are immediately sent toward the receiver with a data rate decided by the sender. UDP does not guarantee reliable and in order packet delivery, but its small overhead and lack of retransmissions make it less prone to introduce additional delays due to packet retransmissions as TCP does. This is the main reason why the UDP transport protocol is mainly used by real-time applications. The first TCP implementations were using cumulative positive acknowledgements and required a retransmission timer expiration to send a lost data during the transport. They were following the go-back-n model. In order to enable good user throughput and to control network congestion a lot of work has been done in order to improve its characteristics and with time TCP has evolved. Today's TCP implementations contain variety of algorithms that enables to control the network congestion and to maintain good user throughput in the wired network. Several variants of TCP can be found in the today's wired networks. TCP Tahoe, TCP Reno, TCP New Reno, TCP Vegas and TCP Sack are few of them that are going to be used in ours simulation scenarios. The most used variant of TCP in the real world today is TCP New Reno. However, every of these TCP variants have unique congestion and flow control mechanisms. A problem is defined in the coexistence of the TCP and UDP traffic in a given wireless channel, caused by the TCP congestion control functionality. TCP continuously probes for higher transfer rates, eventually queuing packets in the buffer associated with the bottleneck of the connection. The wireless connection can be shared by several devices and applications. In such case it is obvious that the connection level and the queue lengths may increase, thus delaying the packet delivery and hence jeopardizing the requirements of the real-time applications. Such situation is even worse because the wireless medium allows transmission of only one packet at a time and in most of the wireless networks it is not full-duplex as in wired links [6-10]. This means that packets should wait their turns to be transmitted. Interference, errors, fading, and mobility are causing additional packet losses, and the IEEE 802.11 MAC layer reacts through local retransmissions which in turn cause subsequent packets to wait in the queue until the scheduled ones or their retransmissions eventually reach the receiver. The back off mechanism of the IEEE 802.11 introduces an increasing amount of time before attempting again a retransmission. In the recent years there was a lot of research regarding the problems that TCP and UDP encounters in a wireless environment [11-15].

3 Simulation Scenario

The network layout of the simulation scenario that is subject of the conducted analysis in this paper is presented at Fig. 1.

We have used the network simulator NS2 (version ns-2.28) in order to simulate the outdoor environment presented in Fig. 1. We can notice that the network topology is consisting of four wired nodes (A0-A3), two wireless base stations (BS0-BS1) and four wireless nodes (n0-n3).

The wireless stations are configured to work according the IEEE 802.11g Standard. Wired connections are configured as given in Table 1. Maximum achievable bandwidth rate is 20Mbps, instead of the maximal 54Mbps for IEEE 802.11g standard, due to home environment.

The queue size value used in the simulation is calculated by multiplying the longest RTT (Round Trip Time) with the smallest link capacity on the path, which is the 20Mbps throughput effectively available over the wireless link. In Table 2 are presented several applications that are used during the simulation. In the simulation we have used real trace files for video chat and movie traffic. Two VBR H.263 Lecture Room-Cam are used for the Video chat and high quality MPEG4 Star Wars IV trace file is used for the movie [15]. In this simulation the game events have been generated at the client side every 60ms [11].

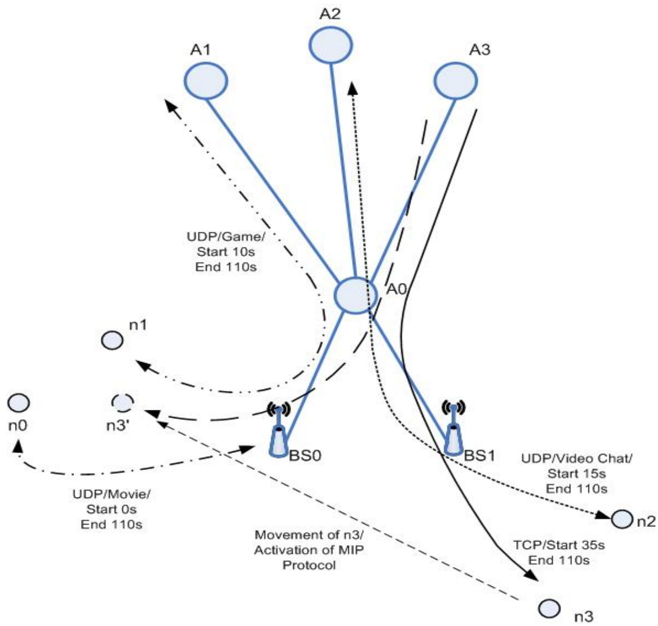

Fig. 1. Simulation Scenario

Table 1. Configuration of wired links simulated at scenario

Node 1	Node 2	Delay	Capacity
A1	A0	10ms	100 Mbps
A2	A0	20ms	100 Mbps
A3	A0	30ms	100 Mbps
A0	BS0	10ms	100 Mbps
A0	BS1	10ms	100 Mbps

Table 2. Types of applications and traffic simulated in the presented scenario

From	To	Type	Transport Protocol	Start	End
BS0	n0	Movie Stream	UDP	0s	110s
A1	n1	Game Traffic	UDP	10s	110s
n1	A1	Game Traffic	UDP	10.1s	110s
A2	N2	Video Chat	UDP	15s	110s
N2	A2	Video Chat	UDP	15.1s	110s
A3	N3	FTP	TCP	35s	110s

Table 3. Simulation parameters

Parameter	Values	Comments
MAC data retransmissions	1,2,3,4	Default value is set at 4
User-BS distance (m)	50,100	Common outdoor environment
MAC queue size (pkts)	25,50,100	Common values
Velocity	static; 4 m/s	Random choice
TCP Transport protocol	TCP Tahoe, TCP Reno, TCP Newreno, TCP Vegas, TCP Sack	Commonly used types of TCP protocols in wired networks.

At the server side updates were transmitted every 50ms toward the client. The payload generated by the client has been set to 42Bytes and the payload generated by the server has been set to 200Bytes. The rest of the packets were set to standard value of 512Bytes for TCP segments. The values for different parameters used in this scenario are listed in Table 3.

For the simulation we have used the shadowing model. The shadowing deviation (σ_{dB}) was set to 4 while the path loss exponent (β) was set to 2.7. These parameters are common for urban environment.

4 Analysis of Transport Protocols

In the following part we observe simulation results from scenario presented in Fig.1, obtained by using the configuration parameters of the links given in Table 1 and applications defined in Table 2. We study the behavior of the UDP/TCP applications and the TCP impact on real time applications in 802.11 wireless networks regarding the throughput as the most important performance metric for non-real-time flows (which use the TCP on transport layer).

The queue size at the MAC layer and the number of MAC layer retransmissions impact the TCP throughput. In Figs. 2 and 3 we show analyses of the throughput of an FTP application as a function of the distance, queue size and the number of MAC layer retransmissions. The results are shown for TCP Tahoe and TCP New Reno.

From the results one may conclude that the queue size of the MAC layer does not impacts the throughput for a given value of the number of MAC layer retransmissions (the curves for different queue sizes and same other parameters are overlapping). Hence, if we increase the number of the MAC layer retransmissions we shall obtain better throughput. The best throughput in this case is obtained when the number of MAC layer retransmissions is set to 4 retransmissions. One may notice that the queue size (i.e. IFQ) at the MAC layer drastically impacts the throughput. It is obvious that if we increase the queue size we will obtain better throughput. It is also obvious that the same throughput is achieved for given values of the MAC layer queue size when the number of the MAC layer retransmissions is set to 3 and 4 retransmissions. The best throughput is achieved when Interface Queue (IFQ) has value of 100 packets.

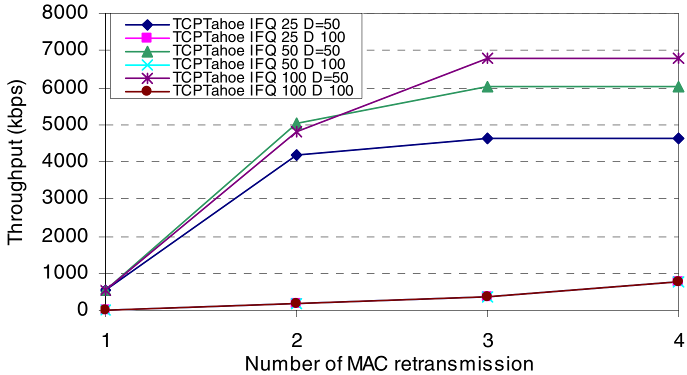

Fig. 2. TCP Tahoe throughput for different access point distances; different MAC queue sizes

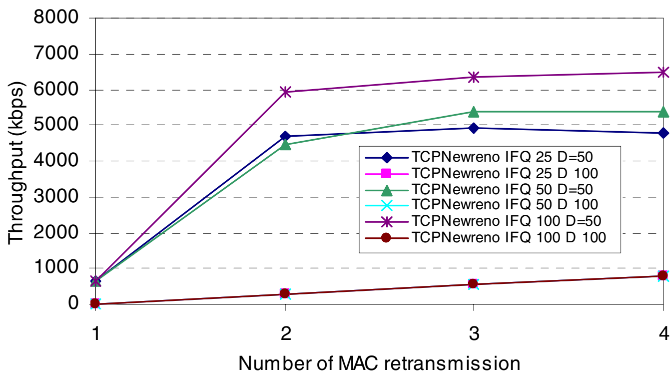

Fig. 3. TCP NewReno throughput for different access point distances; different MAC queue sizes

So far now we are able to conclude that the throughput for all of these transport protocols (i.e. TCP versions) is decreasing as a function of the distance and is increasing as a function of IFQ buffer size and the number of MAC layer retransmission. Drastically lower throughput is achieved at longer distances (i.e. 100m) for all used transport protocols. The queue size at distance of 100m does not impact the throughput as it is a case when the distance is shorter (50m). Highest throughput is achieved for up to four MAC retransmissions. Nearly the same throughput is achieved when MAC retransmissions are set to values of three and four for a given value of MAC queue size (in this scenario the wireless terminal is 50m away from BS).

If we compare the throughput achieved when node is 50m away from the BS for different values of IFQ buffer size and MAC layer retransmissions (Fig. 4) then we can notice that best throughput is achieved for three or higher number of MAC retransmissions. However, the results showed that there is no need for more than 3 retransmission on MAC layer, because there is no significant improvement in average throughput when number of MAC retransmissions is higher than 3 (e.g., 4, 5 etc.).

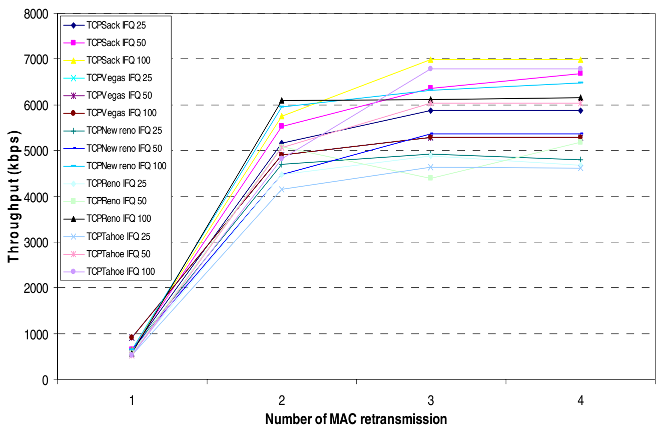

Fig. 4. Throughput of variety TCP protocols for different MAC queue sizes and different number of MAC retransmissions; Distance between n3 and the AP, BS1 is 50m

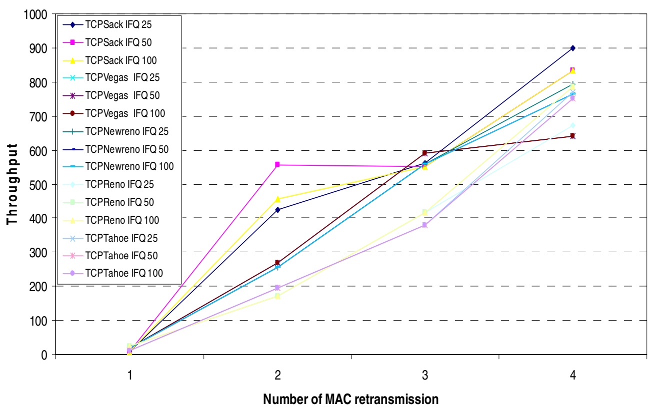

Fig. 5. Throughput of variety TCP protocols for different MAC queue sizes and different number of MAC retransmissions; Distance between n3 and the AP, BS1 is100m

At this point we may fix the number of retransmission to number of 3 at the MAC layer for all TCP versions. Furthermore, the best throughput is achieved with TCP SACK as a transport protocol. TCP Tahoe with larger buffer size (IFQ=100) is the second best case from all TCP versions and all IFQ values. On the other side, TCP Tahoe performs very poor for small IFQ values, i.e. it shows the worst performances for IFQ=25 (the smallest IFQ value in our analysis) when compared with all other cases in Fig. 4. Further, TCP New Reno has slightly better performance than TCP Reno. The overall worst performances are achieved with TCP Vegas excluding the case when the MAC queue size has value of 25 packets (in such case the throughput achieved with TCP Vegas is the second best, after the one achieved when TCP SACK is used as a transport protocol).

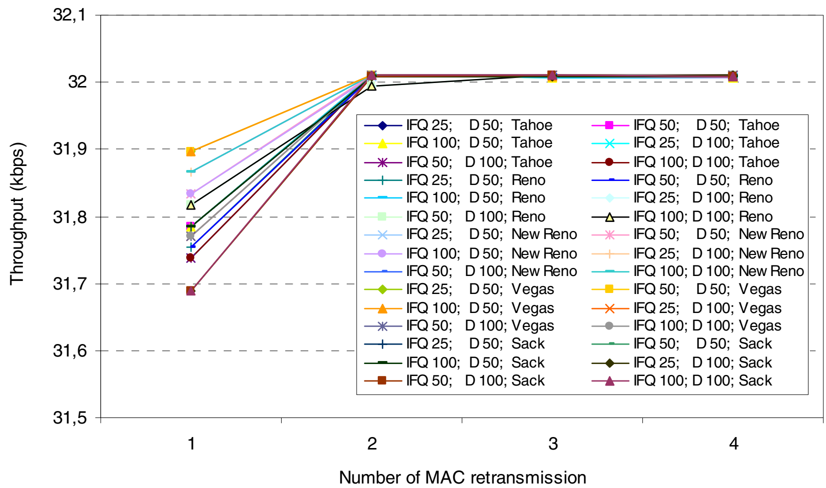

Fig. 6. Comparation of average throughput of the game traffic between the nodes A1-n1 when FTP flow is enabled for different TCP protocols

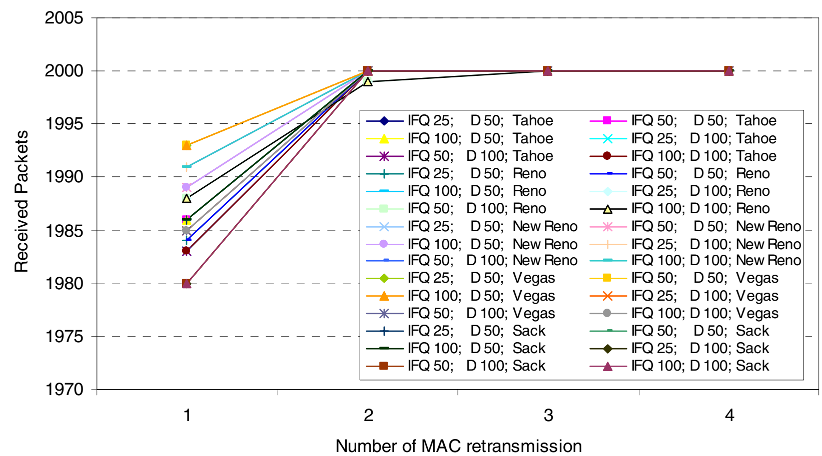

Fig. 7. Received packets of the game traffic between the nodes A1-n1 when FTP flow is enabled for different TCP protocols

In Fig. 5 we present the throughput achieved when the distance between wireless terminal and base stations is 100m. In this scenario, one may conclude that when are needed four MAC retransmission to achieve maximum performances regarding the wireless network and the TCP version. Longer the distance between the BS and the terminals means more MAC retransmissions to achieve the maximum performance in the 802.11 wireless networks. Again, best throughput is achieved with TCP Sack. The second best is achieved with TCP Reno except when the IFQ size is 25 packets (in such case TCP New Reno and TCP Tahoe show better throughput performances). The worst throughput is achieved with TCP Vegas for average of four MAC retransmissions. On the other side, when we use three MAC retransmissions the best throughput is achieved with TCP Vegas, while in such case the worst performance is achieved with TCP Tahoe.

In the following part of this section we provide analysis of the UDP traffic (generated by game application) in presence of background TCP traffic (generated by an FTP flow). If we analyze the results presented in Fig. 6 we will notice that the average throughput of the game traffic between the nodes A1-n1, when FTP flow is enabled, has constant value for different TCP versions when the number of the MAC layer retransmissions is bigger then two, for all possible scenarios. The same note is also valid for the number of received packets, which is presented in Fig. 7.

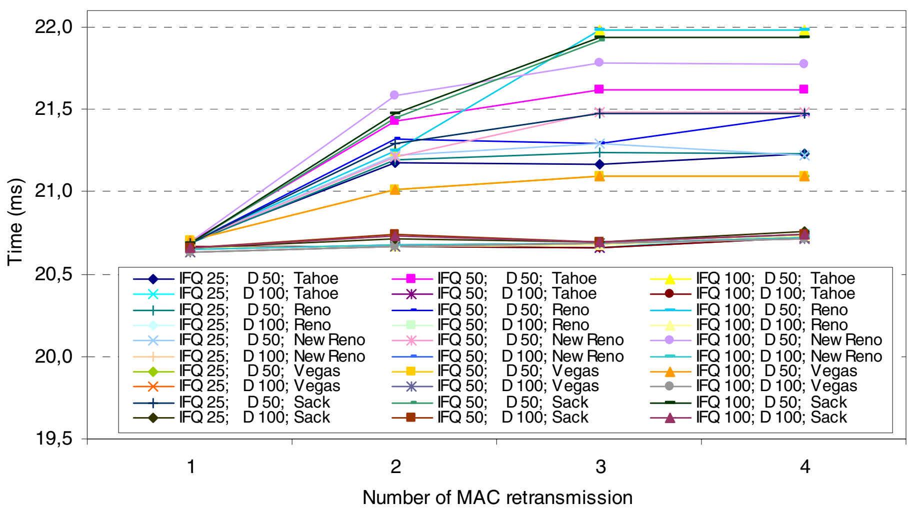

Fig. 8. Comparation of average delay of the game traffic between the nodes A1-n1 when FTP flow is enabled for different TCP protocols

Table 4. Average delay of the game traffic (A1-n1) in (ms) when the MAC retransmissions are set at value of three and the node n3 is 50m away from BS1

L=3	IFQ (pkts)	25	50	100
D=50m	TCP Tahoe	21.1711	21.6210	21.9826
	TCP Reno	21.2408	21.2902	21.9826
	TCP NewReno	21.2953	21.4861	21.7785
	TCP Vegas	21.0952	21.0952	21.0952
	TCP Sack	21.4726	21.9197	21.9386

The results of the average packet delay for game traffic (UDP traffic) when different TCP versions are used for the background TCP traffic (i.e. the FTP flow) are shown in Fig. 8. Numerical results for the packet delay of game traffic are given in Table 4. Lower delays are obtained for larger IFQ buffers and vice versa. The optimal number of MAC retransmissions for smaller IFQ buffers regarding all TCP versions is three, while two retransmissions are good choice for larger buffers, which is similar to conclusions regarding the throughput of the UDP game flow (Figs. 6 and 7). Hence, for UDP game traffic, the optimal number of MAC retransmission is two, which is independent from the distance (i.e. the same results are obtained for different distances between the wireless node and the AP).

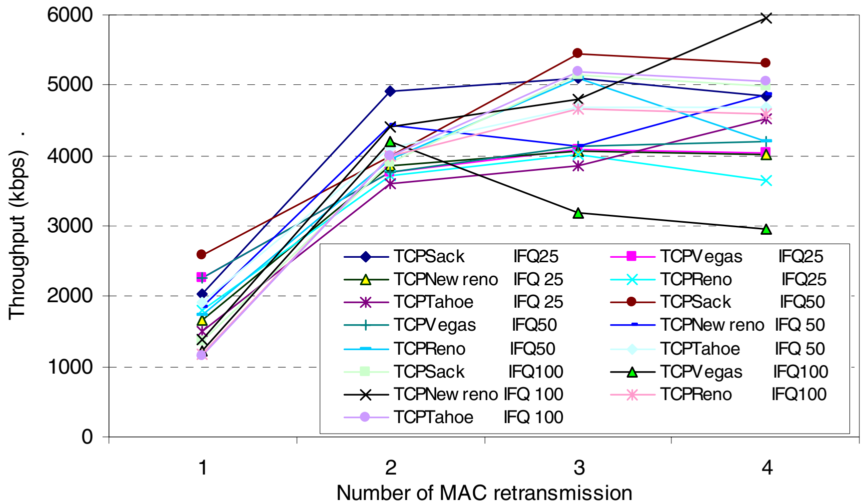

Fig. 9. Average Throughput of the FTP traffic when the node n3 is moving toward BS0 with speed V=4m/s

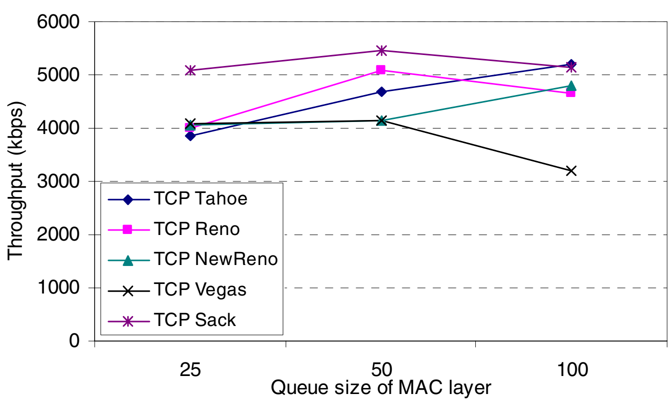

Fig. 10. Average Throughput of the FTP traffic when the node n3 is moving toward BS0 with speed V=4m/s, Number of MAC retransmissions are set at value three

After we have finished the simulations when the node has static position we have conducted the same analysis of the traffic when the node n3 is mobile. As given in Table 3 the mobile node n3 is moving with three different velocities. We have observed scenario when n3 is moving with velocity of 4 m/s (home environment). In Fig. 9 we present the average throughput of the TCP traffic. As one may expect, increasing the number of MAC retransmissions leads to increasing of the average throughput. According to the results optimum value for MAC retransmissions is three retransmissions.

The dependence of the average throughput upon the number of MAC retransmission in 802.11 wireless networks with user mobility in home environment is shown in Fig. 10. The results show that best performances can be achieved when one is using TCP SACK as a transport protocol. Worst results are obtained with TCP Vegas.

5 Conclusions

In this paper we have performed detailed traffic analyses regarding the performances of different TCP versions in 802.11 wireless networks. We have compared different transport protocols by using the throughput as a merit.

The results showed the high importance of the Medium Access Control (MAC) parameters in 802.11 wireless networks regarding the throughput.

The simulations have shown that the best setup in 802.11 wireless networks regarding to the TCP traffic (where TCP counts for most of the traffic in the Internet today): the optimal number of 802.11 MAC layer retransmissions is three, and the MAC queue size is 50 packets for most cases.

If the node becomes mobile the best throughput for TCP-based applications is obtained by using TCP Sack. If we take into consideration the behavior of the TCP traffic in the simulations, the best performance are obtained by using the TCP Sack transport protocol in 802.11 wireless environment. This leads to possibility to create an open transport protocol layer, especially for the case of ad-hoc networks in the home or in the office, when there is no direct communication with other hosts on the global Internet.

Future work is targeted to solutions for open transport layer protocols in future wireless terminals, which is a target for wireless local networks, but it is not limited to them in the wireless world.

References

1. Jacobson, V.: Congestion Avoidance and Control. In: SIGCOMM Symposium on Communications Architectures and Protocols, pp. 314–329 (1988)
2. Floyd, S.: SACK TCP: The sender's congestion control algorithms for the Implementation. Technical report, IETF, March 7 (1996)
3. Fall, K., Floyd, S.: Simulation Based Comparison of Tahoe, Reno and SACK TCP. Lawrence Berkeley National Laboratory One Cyclotron Road, Berkeley (1996)
4. Moraru, B., Copaciu, F., Lazar, G., Dobrota, V.: Practical Analysis of TCP Implementations: Tahoe, Reno, NewReno. In: 2nd RoEduNet International Conference (2003)

5. Mo, J., La, R.J., Anantharam, V., Walrand, J.: Analysis and comparison of TCP Reno and Vegas. In: Proceedings of Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. INFOCOM 1999. IEEE, Los Alamitos (1999)
6. Boyden, S., Mahanti, A., Williamson, C.: TCP Vegas Performance with Streaming Media. In: Performance, Computing, and Communications Conference. IPCCC 2007, Dept. of Comput. Sci., Calgary Univ., Alta (2007)
7. Nikitin, P.V., Celebioglu, O., Kukrer, V.(USA): TCP performance in mobile wireless environment: channel modelling and network simulation. In: SCI 2002 (2002)
8. Chiasserini, C.F., Garetto, M., Meo, M.: Improving TCP over wireless by selectively protecting packet transmissions. In: 4th International Workshop on Mobile and Wireless Communications Network, Bombay, India (2002)
9. Singh, A.K., Iyer, S.: ATCP: Improving TCP performance over mobile wireless environments. In: 4th International Workshop on Mobile and Wireless Communications Network, Bombay, India (2002)
10. Palazzi, C.E.: Residual Capacity Estimator for TCP on Wired/Wireless Links. In: WCC 2004 Student Forum IFIP World Computer Congress 2004, Toulouse, France (August 2004)
11. Palazzi, C.E., Pau, G., Roccetti, M., Ferretti, S., Gerla, M.: Wireless Home Entertainment Center: Reducing Last Hop Delays for Real-time Applications. In: ACM International Conference Proceeding Series, vol. 266 (2006)
12. Le Albayrak, S.L., Elkotob, M., Toker, A.C.: Improving TCP Goodput in 802.11 Access Networks. In: ICC 2007, IEEE International Conference on Communications (2007)
13. Lau, C.K.: Improving Mobile IP Handover Latency on End to End TCP in UMTS/WCDMA Networks, Thesis submitted at School of Electrical Engineering and Telecomunications, The University of New South Wales (March 2006)
14. Palazzi, C.E., Chin, B., Ray, P., Pau, G., Gerla, M., Roccetti, M.: High Mobility in a Realistic Wireless Environment: a Mobile IP Handoff Model for NS-2. In: Testbeds and Research Infrastructure for the Development of Networks and Communities, TridentCom 2007 (May 2007)
15. Movie trace files, `http://www-tkn.ee.tu-berlin.de/research/trace/ltvt.html` (Last accessed, January 2009)