UNIVERSITY
OF TRENTO

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

A CROSS-LAYER SCHEME FOR TCP PERFORMANCE
IMPROVEMENT IN WIRELESS LANS

Dzmitry Kliazovich and Fabrizio Granelli

March 2004

Technical Report <u>DIT-04-025</u>

A Cross-layer scheme for TCP Performance Improvement in Wireless LANs

Dzmitry Kliazovich and Fabrizio Granelli
DIT - University of Trento
Via Sommarive 14, I-38050 Trento (Italy)
E-mail: [klezovic,granelli]@dit.unitn.it

Abstract – **The growing importance of wireless networks requires an effort from the research community for the optimization of the transmission protocols, in order to achieve the best performance. This paper presents a novel approach for the performance enhancement of the most commonly used transport protocol (TCP) in a wireless environment. Cross-layer collaboration, achieved through the introduction of an ARQ Snoop agent within the protocol stack of a wireless host, creates the possibility for the exploitation of link layer ARQ messages within the TCP acknowledgement scheme, to the aim of enhancement of the level of achievable throughput.**

Keywords – Cross-layer Design, TCP over Wireless, Wireless-cum-wired Network Design.

I. INTRODUCTION

Wireless networks are becoming increasingly popular in the world of telecommunications. As a consequence, a relevant effort is being devoted to the provisioning of reliable data delivery for a wide variety of applications over different wireless infrastructures.

The Transmission Control Protocol (TCP) is a connection-oriented byte stream transport protocol used by many end-user applications. In fact, nowadays the overwhelming majority (over 85%) of Internet traffic is TCP-based.

The problem that arises in the usage of TCP over wireless networks comes from the fact that wireless links have different characteristics with respect to wired ones, in terms of less reliability and time-variant behaviour, fading / shadowing problems, node mobility, hand-offs, limited available bandwidth and large RTTs. Since the only reaction provided by TCP to the event of a non-successful packet delivery is the congestion control mechanism, TCP implementations perform poorly in wireless environments.

Thus, a relevant amount of research work has been carried out for the optimization of the TCP performance over wireless networks. The majority of the solutions proposed by the research community are either at the link or at the transport layer.

The presented references are only those papers which are considered to be more closely related to the proposed work, grouped for the sake of clarity into four main classes depending on their features. For a more precise survey, the reader should refer to [1] and [2].

Connection Splitting Solutions. The end-to-end TCP connection is divided into fixed and wireless parts, so that it becomes possible to have more degrees of freedom in the optimization of the TCP over the wired as well as over the wireless link.

Disadvantages of the proposed solutions are mainly in the attempt to perform the splitting of TCP connection in a transparent way from the point of view of the TCP layer on the wired host. This leads to a growing complexity of the Base Station (BS) (as the most common place for splitting) and to higher complexity in terms of efficiency in hand-off handling, prevention of end-to-end semantics of TCP connection, and enlarged software overhead caused by the TCP part of the stack involved at the intermediate point. Well-known solutions in this area are I-TCP [9] and M-TCP [10].

Link Layer Solutions – the attempts to make the wireless link layer look similar to the wired case for TCP. The most relevant and interesting proposal in this category is the snoop protocol [3]: the snoop

agent is introduced at the BS performing local retransmissions using information sniffed from the TCP traffic passing through the BS.

TCP Modifications. There are several studies on the advantages of usage of well-known TCP mechanics over wireless environments, such as Selective acknowledgements (SACK), Fast-Retransmit, and Forward acknowledgement (FACK).

Cross-Layer Design. The relatively large number of solutions described above are trying to solve the problems within a single layer while at least transport and link layer should be jointly taken into account in order to achieve an optimal solution in adaptation of natively-wired TCP to the wireless scenario.

This idea can be included in the wide range of recently-proposed solutions for optimizing wireless network design that are labelled "Cross-Layer Design" [4] methodologies, that break the layering principles by allowing interdependence and joint design of protocols crossing different layers.

However, in spite of the theoretical advantages described above, a relatively low number of propositions aimed at TCP optimization is available in this category. The most promising approach is the Inter-layer Collaboration Protocol (ILC-TCP) [5]. The main concept behind it is the introduction of a State Manager in parallel to the protocol stack for gathering information from TCP, IP and Link/Physical layers and then, if necessary, for providing the collected information upon a request from the other layers.

Another approach is presented in [6], where feedback is allowed between network and transport as well as user/application and transport layers. In this case, information on connection/disconnection from the network is used to improve TCP awareness of wireless network status and, as a consequence, its performance.

The objective of this paper is to present a novel cross-layer approach, called *Link Layer ARQ Exploitation TCP* (LLE-TCP), for the performance improvement of TCP over wireless links. To this aim, a "snoop agent" is introduced, whose purpose is to create a possibility of collaboration between link and transport layers. Therefore, the main performance advantages are achieved through the optimization of interlayer Automatic Repeat Request (ARQ) scheme functionality.

The paper is organized as follows: Section II presents a detailed description of the proposed scheme. Performance evaluation of LLE-TCP through the simulation is presented in Section III, and finally conclusions and outlines of future work on the topic are proposed in Section IV.

II. THE PROPOSED SCHEME (LLE-TCP)

This section is aimed at providing a detailed description of the proposed scheme and its main features.

The considered scenario is a WLAN employing the IEEE 802.11 protocol [7] at the MAC and physical layers. Such an environment is chosen due to the relevance and diffusion of such standard in the framework of local area wireless networks. The Distributed Coordination Function (DCF) is used to discipline access on the wireless channel. Fig. 1 depicts the baseline network scenario which is considered as a reference in the remainder of the section.
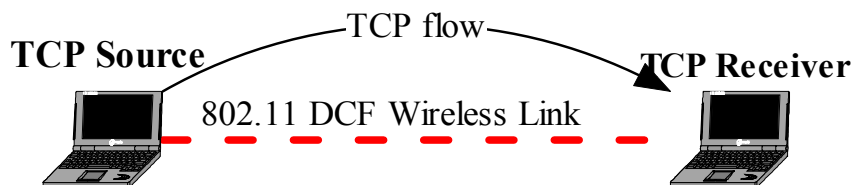

Fig 1. Network scenario used for the LLE-TCP detailed description.

A. Motivations for the Design of LLE-TCP

IEEE 802.11 standard [7] represents the leading MAC (Medium Access Control) solution for wireless local area networks (WLAN). 802.11 provides reliable link layer data transmission by handling packet delivery problems through an ARQ scheme. On the other hand (and level of the protocol stack), the majority of end-user applications use TCP for end-to-end reliable data delivery. Such combination leads to the usage of two ARQ schemes (for MAC and TCP protocols) together within the protocol stack of a single wireless host.

This situation brings to a non-optimal utilization of the available network resources. For the successful delivery of one TCP data packet, in fact, the sender node must wait for an ARQ confirmation at the MAC level, and then for the TCP ACK packet at the transport layer from the receiver. Moreover, the TCP ACK is just one more ordinary data packet from the point of view of the link layer, which is to be delivered and acknowledged by involving one more time the usage of the link layer ARQ procedure as well as the available transmission resources. This implies that ARQ is involved three times at different layers of protocol stack for the delivery of a single TCP data packet.

In this paper, we propose to exploit the information of the link layer ARQ for a more efficient acknowledgement of TCP packet delivery. This means that, when a TCP packet is successfully delivered at the link level, the TCP ACK for the transport layer will not be sent through the channel, but it will be automatically generated locally at the sender side. In order to support this functionality, no changes are needed at the TCP protocol, but a new entity needs to be introduced: the ARQ Snoop agent.

B. The ARQ Snoop agent

The ARQ Snoop agent is a software module designed for the exploitation of the link layer ARQ scheme by TCP in wireless LAN environments. It operates within the protocol stack between the TCP and MAC layers. Fig. 2 shows the logical position of the ARQ Snoop agent in the protocol stack of a wireless node.

For the purpose of a better explanation of the core idea of the proposed approach, a standalone TCP data packet transmission using the designed scheme is presented in Fig. 3.

1) Agent interaction with the Link Layer. An interface to the link layer ARQ procedures is defined in order to provide information about the packet delivery to the destination host. For that reason, two events are specified:

- PACK_DELIVERED event – for an indication of a packet successful packet delivery. This event is generated upon the reception of the ACK frame at the link level as the indication that a data packet is successfully received by the destination node.

- PACK_UNDELIVERED event – for the notification that (after all possible retransmission attempts) the link layer is not able to deliver the packet.

The link layer interface with the ARQ Snoop agent is unidirectional: it only provides the mentioned events and it does not receive any data from the agent.

2) Agent interaction with the Transport layer. The ARQ Snoop agent sniffs all the packets outgoing from the Transport Layer on both sender and receiver sides.

On the sender side, whenever a TCP data packet is detected, the agent stores the relevant information from it, while the packet itself gets through to the lower layers. After that, the Snoop agent remains waiting for an event from the link layer which will indicate the successful or unsuccessful packet delivery.

In case of PACK_DELIVERED event, an ordinary TCP ACK packet is generated by the agent and sent up to the TCP entity of the sender.
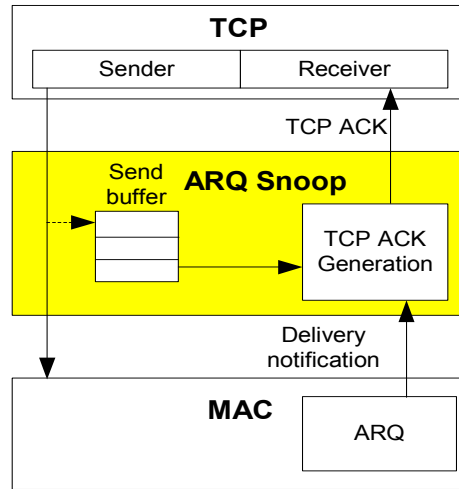
Fig. 2. A view of the protocol stack of the wireless node, where the position of ARQ Snoop agent as well as its relationship with the transport and MAC layers is illustrated.

On the receiver side, generally speaking, the ACK generated by TCP is dropped by the agent. This means that all the packets which have only the ACK flag set and no data content are dropped. Since TCP allows to encapsulate an acknowledgement into data packets (simply by setting the ACK flag and mentioning an acknowledgement number), such packets with data and ACK are not considered by ARQ Snoop agent and left to go through without taking any action.

Fig. 3 compares the TCP data packet delivery procedure in the original and the proposed schemes. LLE-TCP avoids the transmission of TCP ACK packet in the direction opposite to the actual data flow (an aspect that impacts as a significant overhead in the original TCP scheme in terms of channel usage and probability of collisions). Into more detail, after a successful transmission of the TCP Data packet by the MAC protocol, the PACK_DELIVERED event is sent to the ARQ Snoop agent of the Sender Node (Node 0) through its link layer interface. Upon the reception of this event, the agent generates a TCP ACK packet according to the selected TCP acknowledgement scheme. At the same point in time, the TCP ACK packet generated by the TCP layer of the Receiver Node (Node 1) is dropped by the ARQ Snoop agent of the same node in order to prevent its effective transmission on the wireless link.

The yellow part of the framework in Fig. 3 presents the behaviour of LLE-TCP, while the rest of the picture graphically represents the additional time required for the standard TCP ACK packet transmission (which can be clearly considered as producing a relevant overhead with respect to LLE-TCP).

3) Acknowledgement generation module. From the establishment of a TCP connection until its termination, the ARQ Snoop agent traces the TCP packet flow by storing the information about current sequence number of packets passing through the agent. When the PACK_DELIVERED event occurs, an ordinary TCP ACK packet containing no data is generated for the TCP layer. The agent should keep track of the sequence numbers of TCP packets which have already been acknowledged. In order to take into account the possibility of a TCP data packet reception with the ACK flag set from the receiver side, the agent analyzes the acknowledgement number of that packet. In case that number has already been acknowledged by the Agent, the ACK flag must be unset. After that, the packet is forwarded to TCP.

Since the PACK_DELIVERED event occurs exactly after an ACK packet is received by the link layer, the Round Trip Time (RTT) calculated by TCP will be different from the actual one. This situation could impact on the behaviour of TCP. Even if the results available in Section III outline that TCP behaviour using LLE-TCP does not differ from the original one, a more detailed investigation on this point will be a matter of future work on LLE-TCP.
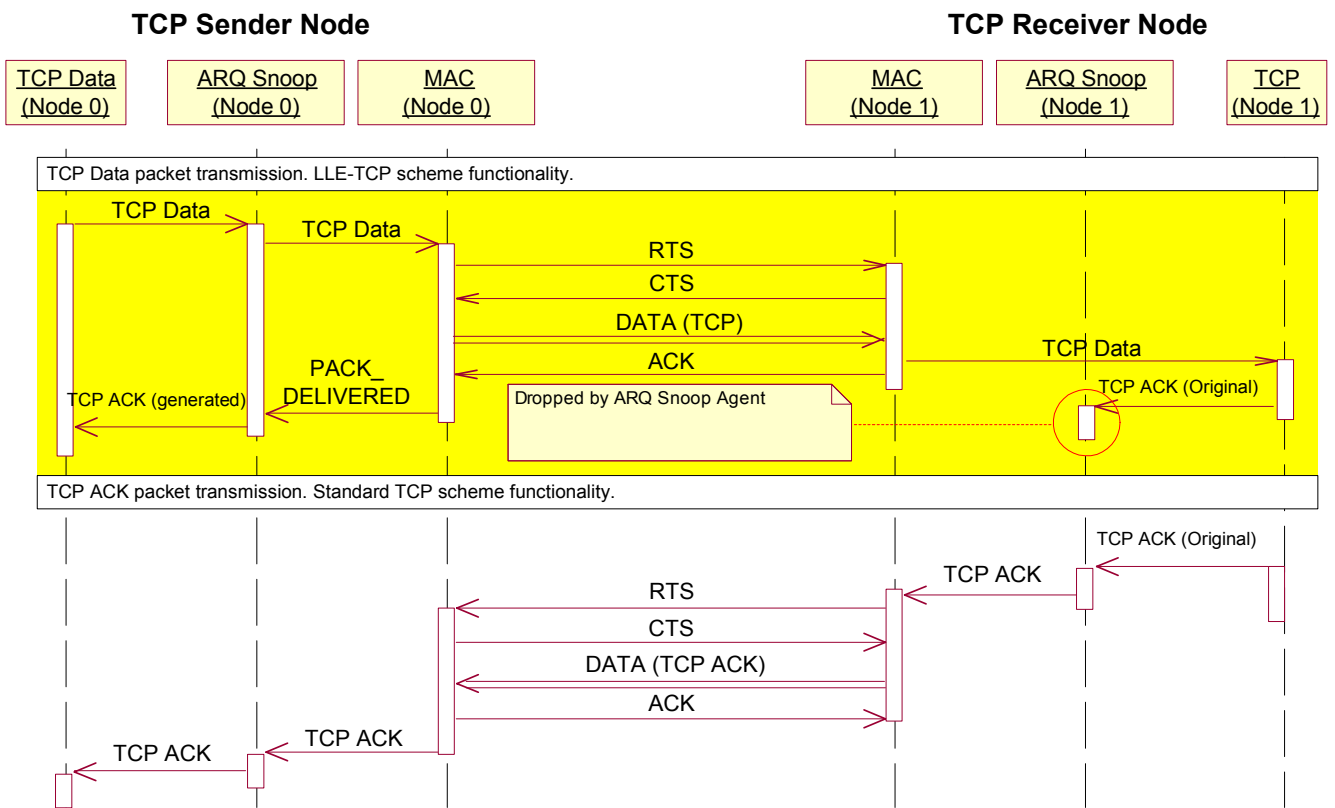
TCP Sender Node

TCP Data (Node 0)	ARQ Snoop (Node 0)	MAC (Node 0)

TCP Receiver Node

MAC (Node 1)	ARQ Snoop (Node 1)	TCP (Node 1)

TCP Data packet transmission. LLE-TCP scheme functionality.

- TCP Data
- TCP Data
- RTS
- CTS
- DATA (TCP)
- ACK
- TCP Data
- PACK_DELIVERED
- TCP ACK (Original)
- Dropped by ARQ Snoop Agent
- TCP ACK (generated)

TCP ACK packet transmission. Standard TCP scheme functionality.

- TCP ACK (Original)
- TCP ACK
- RTS
- CTS
- DATA (TCP ACK)
- ACK
- TCP ACK
- TCP ACK

Fig. 3. Comparison of the behavior of original TCP and LLE-TCP schemes. The picture presents the exchange of packets/frames for the delivery of a single TCP packet.

C. LLE-TCP Integration Scenarios

LLE-TCP is presented in this paper mainly for the case of a single wireless link like between two TCP-connected nodes. However, nowadays wireless networks can differ from that simple scenario. Therefore, an implementation of LLE-TCP in commonly used scenarios like Fixed-to-mobile, Multi-hop or Infrastructure mobile networks will bring obvious advantages. The criterion for the utilization of LLE-TCP is the presence of a TCP destination node within one wireless hop.

D. Advantages of LLE-TCP

Throughput and medium busy time. The main advantage of LLE-TCP is the reduction of medium busy time for the transmission of TCP ACK packets over the wireless link, which clearly brings to a corresponding relevant improvement in the TCP throughput. The level of improvement achieved with the respect to the existing solutions fully depends on the size of TCP data packet as well as on the channel error rate.

Different network scenarios. LLE-TCP can be utilized on wireless links in different network scenarios. In this paper, for the sake of clarity, the description of LLE-TCP is presented only for a WLAN environment, employing access through the DCF. However, in principle, LLE-TCP is not limited to DCF: it can be easily adapted to PCF, since the local generation of TCP acknowledgements requires only the event

interface (refer to section II.B.) from the link layer. The reader should note that LLE-TCP does not require any modification to the IEEE 802.11 link layer itself as well as TCP layer of the stack.

Channel error rate. A passive reduction of channel error rate (compared to the case of standard TCP operation) is an additional outcome of utilization of the LLE-TCP scheme, due to the absence of TCP ACK packets in the channel (they are generated locally at the sender side, and therefore channel errors have no impact on them).

E. Limitations of LLE-TCP

Performance. Probably the main limitation of LLE-TCP is the requirement for resources on the sender side for traffic analysis and acknowledgement generation: assuming multi-flow and multi-user environments (like Infrastructure BSS), the requirements for the resources from the Access Point can increase dramatically. An implementation of LLE-TCP inside protocol stack of PC operation system can put more light for the requirements of LLE-TCP in various operation scenarios and it will be the first-order future task for the authors.

III. EXPERIMENTAL RESULTS

The performance of the proposed protocol is analyzed by simulating experiments using the ns-2 network simulator [8]. Most of the results are achieved in grid topology, where two static nodes are linked through a single TCP connection with a TCP Data packet size of 1KBytes (which is considered in most works to be a default value for experiments). An ARQ Snoop agent is added to both nodes through the modification of the source code. One node (the TCP Sender) continuously sends data. Its ARQ Snoop agent provides the possibility for the operation of an ARQ scheme for TCP relaying on the information indicating the packet delivery result incoming from the link layer. At the other side of the link, the second node (the TCP Receiver) replies with TCP acknowledgements, which are dropped by its own ARQ Snoop agent.

In spite of the fact that LLE-TCP can operate in various multi-flow and multi-node network scenarios, this section evaluates its performance on single wireless link with two nodes. The reason is that such an environment can fully present the benefits achieved by utilization of LLE-TCP. LLE-TCP was also tested in the other scenarios, showing good agreement with the case presented in the paper.

The throughput of TCP connections is chosen as the main parameter for performance analysis of LLE-TCP. The resulting throughput values are in most cases compared against standard TCP scheme over a 1 Mbit/sec IEEE 802.11 wireless link with two-ray ground propagation model. The resulting ns-2 node structure is depicted in Fig. 4, where only yellow blocks are enabled.
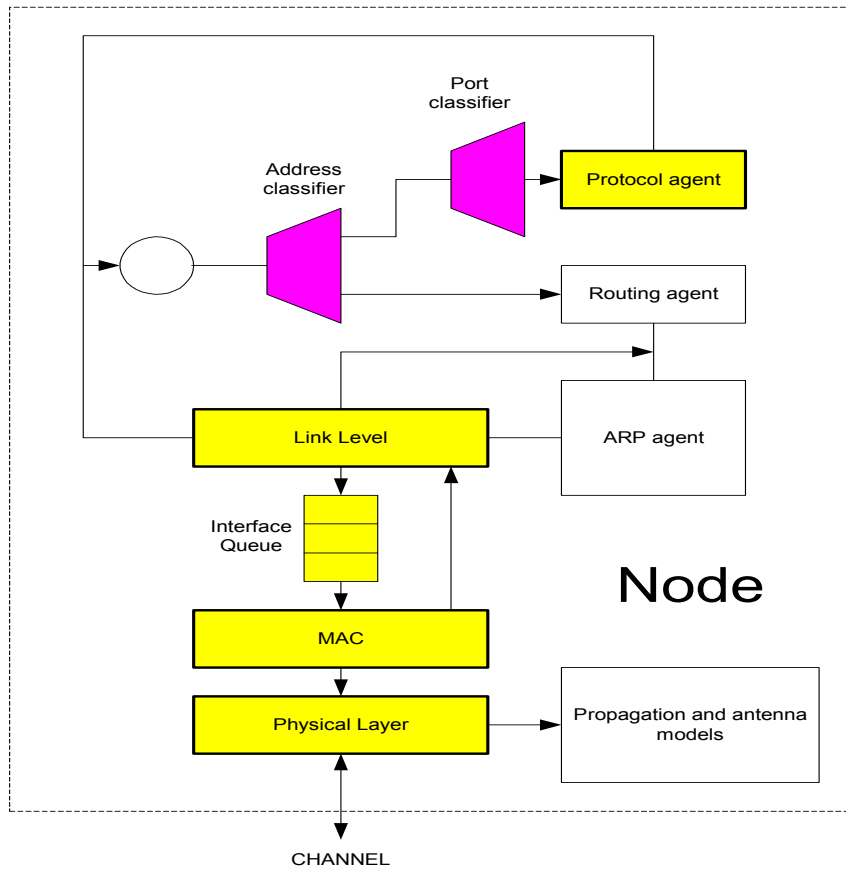
Fig. 4. The ns-2 node structure used in the simulations.

Fig. 5 shows the throughput of TCP Tahoe against the proposed scheme. As the packet error rate increases, LLE-TCP has an increasing advantage against standard TCP. The average throughput improvement of LLE-TCP on TCP Tahoe with a standard acknowledgement scheme for a low percent of packet errors (in the interval from 0 to 5) of the total channel bandwidth is more than 28% and up to 120% at high (18-20%) packet error rates (see Fig. 6).
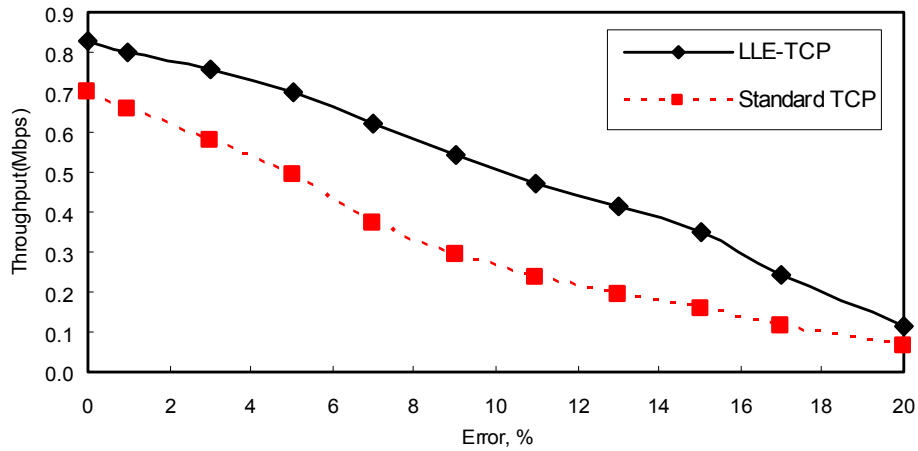
Fig. 5. Performance comparison of LLE-TCP scheme against the standard TCP implementation (TCP Tahoe).
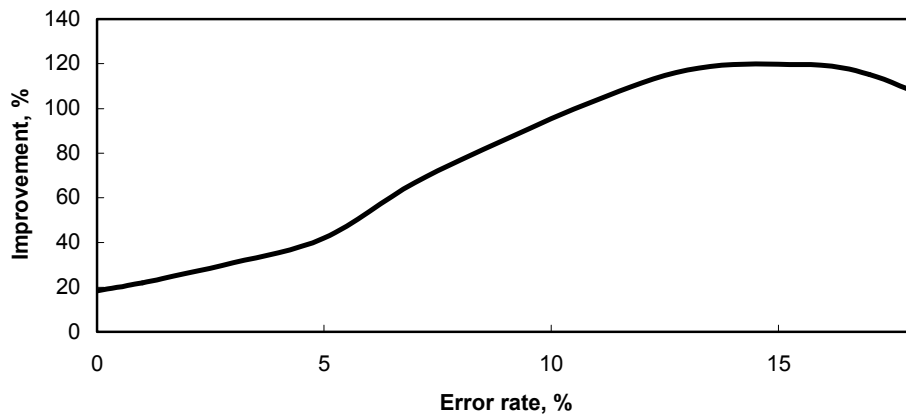

Fig. 6. Improvement achieved by LLE-TCP against TCP Tahoe on a 802.11 WLAN.

A. LLE-TCP Performance with different TCP packet sizes

The goal of this sub-section is to show the dependence of LLE-TCP performance on the size the TCP packets. The range of variation of the packet size is chosen from 40 bytes as the smallest size (TCP ACK packet size) up to 1500 bytes.

Fig. 7 illustrates the throughput of the proposed scheme in TCP connections using different packet sizes. The exponential growth of the improvement with the reduction of the TCP data packet size shows the advantages of the proposed scheme not only in case of TCP flows with a standard packet size of 1K, but also in the case of small packets (which are the most commonly used on the Web for such applications like HTTP-based web browsing).

For the maximum packet size of 1500 bytes the throughput of LLE-TCP has an improvement of only 13% against the standard TCP scheme, while for the default packet size of TCP of 1KB it is 18%. For a packet size of 40 bytes LLE-TCP achieves a throughput improvement of 104%.
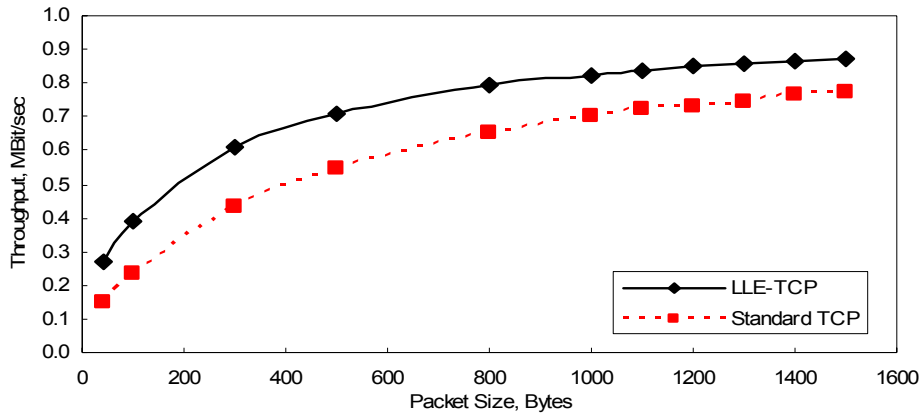
Fig. 7. Comparison of throughput level of LLE-TCP against TCP Tahoe for different packet sizes.

B. File Transfer Performance

A file consisting of 700 TCP packets each of 1KB is selected to compare the advantages of LLE-TCP against standard TCP. Wireless channel error rate is equal to 0 through all the experiment. Fig. 8 presents the throughput for LLE-TCP in comparison with standard TCP scheme for the transmission of the mentioned file. The transfer was started three seconds after the beginning of the simulation. The amount of the time taken for the delivery of the file by LLE-TCP (7.37 sec) scheme is 18.1% less then time taken by standard TCP scheme (9 sec), thus providing an additional advantage of the proposed approach.
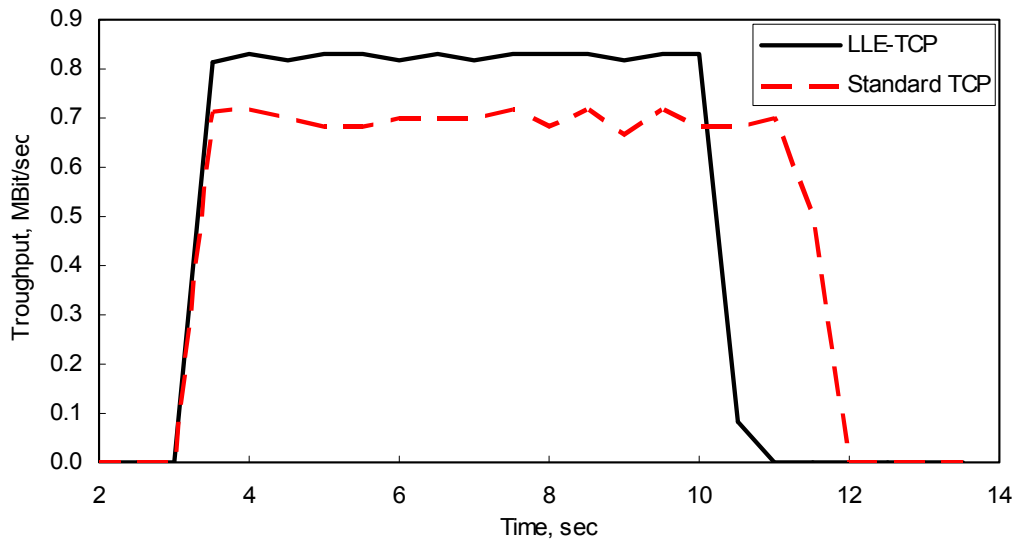

Fig. 8. Throughput level of LLE-TCP and Standard TCP for the transmission of a 700 KB file.

C. Comparison with other schemes

The results of LLE-TCP are compared with other approaches in order to determine the value of the proposition with respect to the state-of-the-art.

o   I-TCP was selected for comparison as the most famous proposition in the Splitting Connection area able to achieve a significant level of improvement. Another advantageous characteristic of I-TCP against other approaches is in providing an improvement even in case of usual operation of wireless link, i.e. when there are no specific wireless problems for TCP (like frequent disconnections and user mobility).

o   ILC-TCP protocol was chosen in the Cross-Layer Design area, which is the area in which also LLE-TCP can be classified.

Actual implementations of the selected protocols are not included in the present work, but the numerical results are directly those reported in the corresponding papers: [9] for I-TCP and [5] for ILC-TCP.

The average results regarding the performance enhancements achieved by I-TCP, ILC-TCP and LLE-TCP are presented in the Table I, for operation in a usual mobile environment as well as in the case of disconnections, user mobility, handoffs and other conditions.

TABLE I. COMPARISON OF LLE-TCP AGAINST COMPETING APPROACHES

Type of TCP	Achieved Improvement	
	Normal	Other conditions
LLE-TCP	18 %	Up to 120 %
ILC-TCP	-	Up to 30 %
I-TCP	About 7 %	Up to 53 %

IV. CONCLUSIONS AND FUTURE WORK

This paper presented a novel approach for the performance enhancement of TCP over wireless infra-structures (such as WLANs). The advantages in terms of throughput jointly achieved by cross-layer collaboration technique and link layer ARQ exploitation for improving the acknowledgement scheme of the TCP are presented. The performance of the proposed scheme is evaluated through extensive simulations with different implementations of TCP by comparing the results with standard TCP scheme of the protocol stack as well with other approaches in this area. Results underline significant advantages of the designed scheme in any channel error condition.

At the present moment, in order to confirm the excellent results of the simulation in the ns-2 network simulation environment, we are considering the possibility of the implementation of the proposed scheme in the actual protocol stack of a 802.11-enabled PC, so that it'll be possible to conduct experiments in a real wireless environment.

REFERENCES

[1] Pentikousis, K., "TCP in Wired-cum-Wireless Environments", in IEEE Communications Surveys, Vol. 3, No. 4, Fourth Quarter 2000.

[2] Daniel, L., "Introduction to TCP and its Adaptation to Networks with Wired-cum-Wireless Links". Seminar on Transport of Multimedia Streams in Wireless Internet, September 2003.

[3] Belakrishnan, H., Seshan, S., Amir, E., Katz, R., "Improving TCP/IP Performance over Wireless Networks," Proceedings of the 1st (MOBICOM), Berkeley, CA, November 1995, p. 2-11.

[4] Qi Wang, Abu-Rgheff, M.A., "Cross-layer signaling for next-generation wireless systems," IEEE Wireless Communications and Networking Conference (WCNC 2003), Vol. 2 , pp. 1084-1089, 16-20 March 2003.

[5] Chinta, M., Helal, S., "ILC-TCP: An Interlayer Collaboration Protocol for TCP Performance Improvement in Mobile and Wireless Environments," Proceedings of the Trird IEEE Wireless Communications and Networking Conference (WCNC), New Orleans, Louisiana, March 2003.

[6] Raisinghani, V.T.; Singh, A.K.; Iyer, S., "Improving TCP performance over mobile wireless environments using cross layer feedback," 2002 IEEE International Conference on Personal Wireless Communications, pp. 81–85, 15-17 Dec. 2002.

[7] "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," IEEE 802.11 standard, 1997.

[8] NS-2 simulator tool home page. http://www.isi.edu/nsnam/ns/, 2000.

[9] Bakre, A., Badrinath, B.R., "I-TCP: Indirect TCP for mobile hosts". Distributed Computing Systems, Proceedings of the 15-th International Conference, pages 196 – 143, June 1995.

[10] Brown, K., Singh, , S., "M-TCP: TCP for Mobile Cellular Networks," Computer Communication Review, Vol. 27, No. 5, October 1997.