Performance of TCP/IP data transfer over the DECT radio interface

Andrea Baiocchi, Roberto M. Cautelier, Maria G. Di Benedetto, Cecilia Poscetti, Maurizio Simeoni (*)

(*) Authors are listed in alphabetical order
University of Roma "La Sapienza" - INFOCOM Dept. - Via Eudossiana 18 - 00184 Roma, Italy
Contact author e-mail: andrea@infocom.ing.uniroma1.it

Abstract

DECT technology can play an important role in local as well as public access scenarios, at least in a short-medium term perspective, to provide means for the rapid deployment of moderate bandwidth (up to a few hundreds of kbit/s) data and multimedia services over radio interfaces. In this view, the extension to DECT mobile users of currently available data services, today mainly based on a TCP/IP architecture, must be analyzed.

In the present study, we elaborate a detailed simulation model of a DECT radio interface in a (mainly) indoor business environment, by taking into account all major functions performed by DECT layers and by the TCP/IP protocols and considering both single bearer and multibearer MAC connections. The impact of the radio channel unreliability and of the bearer and connection handover mechanism of DECT on TCP transfer performance are analyzed and reported to lead to a significant increase of delays, though no negative interaction appears to arise between lower layer management functions and TCP flow control.

1. Introduction

A significant effort is being spent for extending data and multimedia service availability to mobile users, in the local environment (wireless LANs being specified in ETSI and IEEE 802), in the radio local loop (e.g. wireless ATM) and in the public access domain (e.g. Personal Communication Services in the U.S.A., UMTS and MBS in Europe).

The Digital European Cordless Telecommunications (DECT) [1] air interface defines a radio access intended for a broad range of applications, e.g. domestic or business private environments or cordless access to public networks. DECT technology can play an important role in all of these scenarios, at least in a short-medium term perspective, to provide means for the rapid deployment of moderate bandwidth (up to a few hundreds of kbit/s) data services over radio interfaces [2]. In this view, the extension to DECT mobile users of currently available data services, today mainly based on a TCP/IP architecture, must be analyzed [3][4][5][6].

In the above context, a detailed simulation of a realistic DECT scenario to assess the performance of TCP data transfer is a preliminary step in the process of understanding which are the most critical parameters to be selected at the various layers (MAC, Data Link Control, TCP) and of pointing out possible shortcomings of currently used TCP versions, specifically designed for fixed networks with high performance transmission media.

In the present study, we elaborate a detailed simulation model of a DECT radio interface in a (mainly) indoor business environment, in the framework of the COBUCO project [7], which is going to develop, install and use a UMTS demonstrator including mobile IP hosts and exploiting TCP/IP based applications.

All major functions performed by DECT layers (Physical, MAC, DLC) and by the TCP are taken into account. An advanced scenario is considered, with both narrowband data connections and variable bit rate data connections, using slot bundling in the DECT radio interface, besides voice connections; so, both single bearer and multibearer MAC connections are multiplexed in the radio interface. A multiple cell scenario is considered, hence accounting also for connection handover.

The focus of this work is to study: i) the performance of standard TCP over unreliable channels (both as to quality and bandwidth availability, due to handover failures); ii) the interaction of the error protection and recovery procedures of DECT protocols with flow control run by TCP entities. The most evident findings of the presented preliminary results are that: a) a significant reduction of useful TCP throughput results from point i) above, but DECT is essentially responsible for that; b) point ii) seems not to be an issue.

As for the rest of the paper, Section 2 sets the scene, identifying considered physical and protocol architecture of the simulated network. Section 3 describes the simulation model and Section 4 presents preliminary performance results.

2. Reference scenario

The classical scheme of a TCP/IP communication envisages an information exchange between two hosts across an IP network (actually relying on a multiplicity of different network and on IP internetworking).

The variation considered in the present work consists of making one of the hosts a mobile one by means of a radio access. The radio interface is based on DECT; in particular, it uses a multibearer DECT connection, to achieve relatively large bandwidth. The fixed part of the DECT subnetwork comprises 3 transmission equipment (Radio Fixed Points, RFPs) and a centralized Base Station (Central Common Fixed Part, CCFP), dealing with higher layer functions. An Interworking Unit (IWU) is required to access the external IP network: the IWU acts as an IP router interfacing the DECT subnetwork on one side and the external subnetwork on the other side.

The chosen configuration may represent a large private network environment (e.g. wireless LAN), interconnected to outside world (see Figure 1). The protocol architecture corresponding to the user plane of the considered communication is depicted in Figure 2.

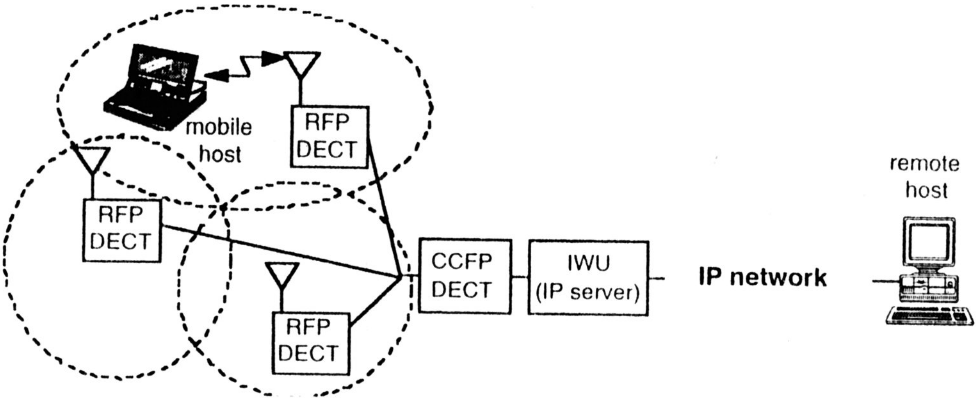

Figure 1 - Network configuration.

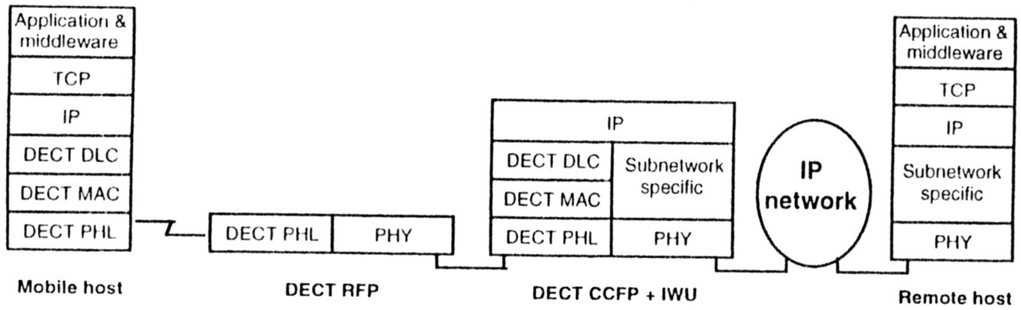

Figure 2 - Protocol architecture of the considered network.

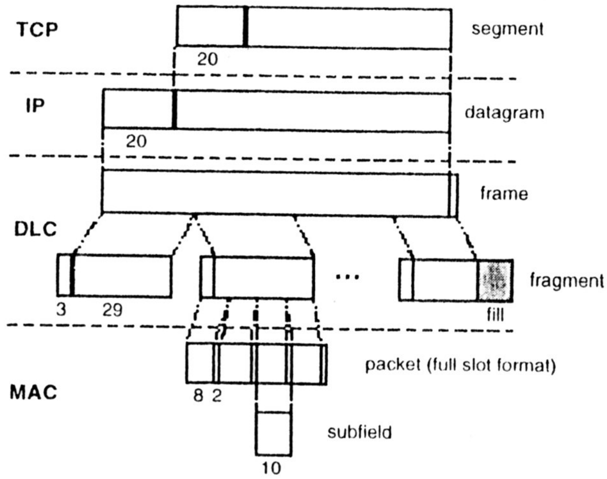

Figure 3 - Protocol Data Units of the considered protocol architecture and their relations (all lengths are in bytes).

The protocol data units (PDU) of the different layers are shown in Figure 3 with their respective names and sizes in bytes. For each PDU, the protocol control information fields are shaded. Given L user bytes to transfer in the TCP layer, the number of required MAC packets is at least (i.e. provided no retransmissions in DLC and MAC layers take place) $ceil((L+42)/29)$, where L is in bytes and $ceil(x)$ is the least integer not less than x.

The following procedures are implemented for the simulation of a mobile TCP user:

i) protection mechanism provided by DECT MAC layer (redundancy and stop&wait ARQ);
ii) fragmenting, sequencing and go-back-N ARQ of DECT DLC layer;
iii) error and flow control of TCP: the implemented TCP is as described in [8].

In more detail, the implemented MAC procedures correspond to the Ip Correction channel [1], i.e. a part of the MAC packet B field is divided into 10 bytes subfields, each protected by a CRC field (see Figure 3). If errors are detected on at least a subfield of a packet, retransmission is repeated the next frame; each time, correct subfields are stored by the receiving side, until the MAC packet can be reconstructed without errors. A maximum number of 10 transmission attempts is allowed. It is also assumed that the undetected error probability of MAC CRCs is negligible, so that all error detection codes added by upper layers (DLC LU2, IP headers, TCP segments) have essentially no impact on performance.

The DLC LU2 service is considered. The protocol at the frame level is essentially void, except of a 2 bytes checksum field appended to DLC frames. The main part of the DLC LU2 service is fragmenting of DLC frames into small data units (see Figure 3), comprising a three bytes overhead, for useful payload length indication, send sequence numbers and receive sequence numbers. Sequence integrity is taken care of, by means of a Go-back-N ARQ protocol, with a transmission window of 63 fragments and two timers. A first timer is set to 10 TDMA frames: when it expires and the sequence is not complete, retransmission is asked for and the second timer is started, lasting 20 TDMA frames. When this second timer expires, fragment recovery is given up and the next DLC frame is considered.

As for TCP, all main functions are implemented [8] (variable window management with slow start and congestion avoidance, timers, acknowledgements, fast retransmission and fast recovery algorithms, round trip time estimation). The fast retransmission algorithm envisages that a segment is retransmitted even though the retransmission time-out (RTO) is not expired yet, when at least three consecutive acknowledgements carrying the same value are detected. Fast recovery defines a faster way to increase the transmission window than allowed by slow start and can be used for retransmissions triggered by the fast retransmission algorithm (not those due to RTO timeouts). The RTO computation is made on the basis of the classical Jacobson algorithm [13].

Functions associated with IP header are neglected (e.g. header error checksum) and the fixed network segment is idealized as for information integrity and delay variation (note that if delay is almost constant, round trip time estimation mechanism of TCP manage to track it quite successfully).

3. Simulation model

We consider a system consisting of three RFPs (cells) which are visited by Mobile Terminals (MTs) of different types during the simulations. Both intra-cell (bearer) and inter-cell (connection) handovers are considered, but not external handover, i.e. the three RFPs belong to the same MAC cluster. The RFPs are assumed to be capable of handling each up to the entire DECT TDMA capacity.

MTs are classified according to the physical packet type used for their connections in the DECT air interface and the number of bearer involved on them. We assume that each MT is capable of supporting only one connection at a time. Single bearer connections can be half, full or double slot based, but only full slot multibearer connections are considered. So, a MT can be of only one of the following types respectively: P08j, P32, P80 or Data. All connections are assumed to be symmetric bidirectional. For each type of MT, call arrivals are modelled as Poisson processes. Call durations are exponentially distributed, except of Data MTs, whose calls have the following duration ϑ distribution: $P(\vartheta=1 \text{ min})=0.333$; $P(\vartheta=2 \text{ min})=0.333$; $P(\vartheta=3 \text{ min})=0.333$; $P(\vartheta=30 \text{ min})=0.001$.

The simulation model comprises two main parts: i) a "background" environment, loading the DECT air interface and made up by narrowband data or voice MTs (P08j, P32 and P80) and wideband data MTs (Data); for these MTs only DECT MAC functions are emulated; ii) a single special Data MT, referred to as "TCP user", for which layers on top of MAC up to transport layer (TCP) are also simulated. In the first part of this Section, the modelling of MAC functions is considered; in the second part, the specific assumptions in the modelling of the TCP user are discussed.

The simulation program runs on a TDMA frame by TDMA frame basis and emulates the major MAC layer procedures through decision taking algorithms, as the MTs remain or pass from one state to another (Figure 4).

At the beginning of the simulations, all MTs are set to the Idle_Locked state and they are distributed in the three cells at random. While in this state, MTs are just allowed to pass to the Call_Setup state. The Call_Setup state is not defined as a part of the DECT standard but has been included in this work to indicate that a given MT has a connection setup procedure in progress. Thus, that first transition can be thought of as the arrival of a new call request in the DECT system.

At call set up and each time the DECT connection is to be modified, the mobile terminal uses the so called channel list, where physical DECT radio channels are labelled as being quiet (measured signal level < –93 dBm) or busy or somewhat in between the two extremes. Hence, we define three channel labels, quiet (Q), busy (B) and interfered (I). The channel list is updated by the mobile terminal itself. To account for measure imperfections and inter-cell interference, we assumed the following distribution for the channel list updating outcomes: $P(\text{Idle}\rightarrow\text{Q})=0.8$; $P(\text{Idle}\rightarrow\text{I})=0.2$; $P(\text{Idle}\rightarrow\text{B})=0$; $P(\text{Reserved}\rightarrow\text{Q})=0$; $P(\text{Reserved}\rightarrow\text{I})=0.5$; $P(\text{Reserved}\rightarrow\text{B})=0.5$.

Immediately after a MT enters the Call_Setup state, the corresponding Call_Setup timer (3 s) starts. A channel list generation algorithm provides each MT with a statistically modified version of the states of reservation of the channels, simulating the construction of the individual channel lists. Once a MT has obtained the permission for attempting a bearer setup in a certain TDMA frame, it selects a channel from its channel list and requires the channel to the RFP it is locked to. The FP decision logic evaluates if the channels selected by a

MT are idle or have already been reserved by other MTs, and detects if collisions take place. Frame by frame, idle channels at any RFP are assigned to the MTs that select them if no collision occurs. When a MT selects a channel that was previously reserved, the corresponding bearer setup attempt is considered to have failed. The MT can repeat the above procedure up to a specified maximum number of times before the expiration of its setup timer. For multibearer connections (Data users) after each successful bearer setup, a new bearer set up can be attempted, subject to not exceeding its target number of bearers.

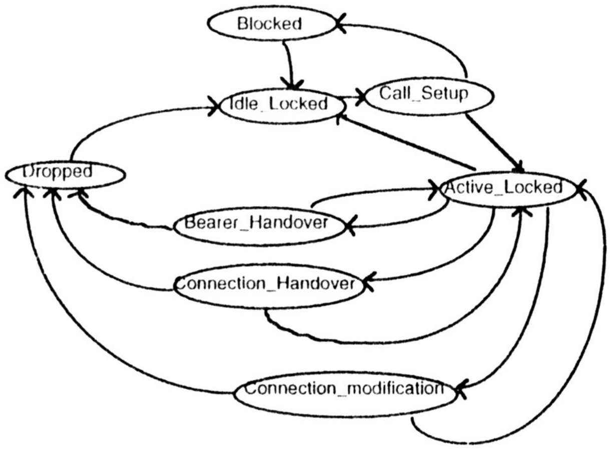

Figure 4 - MAC layer MT evolution state diagram.

If the Call_Setup timer expires and the MT has not yet established the minimum acceptable number of bearers, it enters the Blocked state. Otherwise, it enters into the Active_Locked state. When a MT enters the Blocked state, it remains there for a TDMA frame before finally returning to the Idle_Locked state. In the Blocked state all bearer previously established for the connection are released.

A MT returns to the Idle_Locked state from the Active_Locked state to represent the natural termination of its connection.

A MT enters the Bearer_Handover state each time it has at least one of its bearers needing to be handed over to another physical channel. This situation is generated in a probabilistic way, considering that for a particular bearer the time between the requirement of a new bearer handover and the last successfully finished one is a random variable with an exponential pdf. For a MT, only one bearer handover at a time is resolved and therefore bearers needing the bearer handover procedure are sequentially served. A completely updated channel list is immediately provided to a MT as it passes from the Active_Locked state to the Bearer_Handover state. The channel list is periodically updated every 30 frames until the MT leaves that state. A single bearer handover procedure ends when either the MT obtains the reservation of a new physical channel or the respective timer expires. At the expiration of each timer the number of surviving bearers in the connection is checked . If it falls below the minimum acceptable number, then the MT enters into the Dropped state and in the following frame all old bearers of the connection are released. On the other hand, each time a channel is substituted by a new one by means of a bearer handover, it is no longer reserved in the respective base station.

We assume that a MT hands over a connection only after selecting a new RFP, i.e. when passing from one cell to another. A 3-states Markov chain has been implemented, so that each time a transition on the chain occurs, a connection handover procedure is triggered. The procedure is essentially the same used for the Bearer_Handover case and can be viewed as a sequence of individual bearer handover procedures over the whole set of bearers of a given connection. When a MT enters this state a connection handover timer (3 s) starts. The channel list is updated immediately after the MT passes from Active_Locked to Connection_Handover and then periodically updated every 30 frames. The MT tries to set up new bearers on channels within the new base station. At the expiration of the timer, the number of established bearers with the new base station is checked . If it is less than the minimum acceptable number of bearers, then the MT enters into the Dropped state and in the following frame all bearers belonging to the old and the new connections are released. Each time a channel is reserved with the new base station, a channel with the old one is released.

A MT can be forced to enter the Connection_Modification state from the Active_Locked state at any time. The new desired minimum and target number of bearers for transmissions on both direction must then be specified and a special routine decides the best strategy that must be implemented in order to reach that specifications. The TCP user enters this state to restore the original target number of bearers, when the actual number of bearers falls below a threshold *bth*. A series of single bearer setups are then carried out.

As regards the TCP user, it is assumed that a bidirectional symmetric multibearer DECT connection is set up when the TCP connection is initialized. The multibearer DECT connection of the TCP user is set up with a target number of 10 bearers. Each component bearer is a full duplex full slot bearer (one full slot per frame in both uplink and downlink). The parameter *bth* is set to 4, i.e. the TCP user moves into the Connection_Modification state whenever its number of bearers falls below 4. Moreover, should the MAC connection of the TCP user be dropped, it is assumed that a new connection set up is started anew immediately

(Call_Setup state); on the contrary, the TCP connection is assumed to be never broken.

The BER and bearer handover processes are modelled as follows. A two state Markov chain is defined, associated to each bearer of the TCP user, the two states being labelled as B (bad) and G (good). Transitions between the two states occur at TDMA frame rate, according to the probabilities $P(B \rightarrow G)=0.2$ and $P(G \rightarrow B)=0.02$. The BER (probability of bearer handover required) is 3e–3 (1e–4) in the G state and 1.7e–4 (1e–3) in the B state [2]. Bit errors are assumed as independent of one another; hence the subfield error ratio (SER) can be easily computed, yielding 0.0135 in the G state and 0.2137 in the B state. The Markov modulation approach aims at modelling the effect of variable quality transmission channel typical of radio interfaces.

New TCP segments are generated according to a Poisson or Batch Poisson arrival process, both at mobile TCP MT and at the remote fixed host; in case of batch arrivals, the batch size is geometrically distributed. TCP segments waiting for acknowledgement are stored in a so called TCP buffer; also DLC fragments are stored in a buffer. Both buffers are assumed to have infinite size (i.e. buffer overflow is neglected).

4. Performance evaluation

Extensive simulation experiments have been carried out. The results discussed here refer to throughput and performance of the TCP user. To this end, we define a reference scenario, corresponding to a moderate overall load of the DECT air interface and to the TCP model parameter values given at the end of the previous Section. The TCP segments are generated according to a simple Poisson process of rate λ (seg/s) in the reference case.

We consistently assumed that the TCP payload has a constant length of 512 bytes. With the assumed protocol stack, this requires 20 MAC packets at least (i.e. without any retransmission) to carry a TCP segment. A single full duplex bearer can therefore carry at most 5 seg/s (a DECT TDMA frame lasts 10 ms). Since the TCP user exploits a MAC multibearer connection with a target number of 10 full duplex bearers, the maximum theoretical throughput of the TCP user is 50 seg/s. Due to the TCP user MAC state machine evolution (see Figure 4), the actual number of bearers varies during the lifetime of the MAC connections, hence the upper bound of the throughput, which can be computed as the mean number of bearers actually used by the TCP user times their theoretical capacity (5 seg/s). This upper bound is denoted as *THteo*.

The graphs in Figure 5 show the throughput behaviour as a function of the TCP segment generation rate λ for the reference case and for variations of this case (with doubled BER, doubled bearer handover probability, higher background load and batch arrivals with average batch size *g*). Each plot compares *THteo* with the actual throughput, *TH*, and the ideal throughput, *THid*. The last one is defined as $THid=min\{\lambda,THteo\}$, i.e. it represent the ideal value of TH as determined by the offered load and the available MAC connection capacity.

Except of the double BER and high background load cases, the limiting value of *TH* tends to something ranging between 76% and 80% of *THteo*. In the double BER case, as expected, the reduction factor falls to 68%. In the high background load case, the reduction factor is somewhat higher, i.e. about 83%, probably because of a saturation effect (*THteo* is lower than in all other cases). As a numerical example, in the reference case from a nominal throughput of 50 seg/s (204.8 kbit/s of user data; note that almost one third of the gross MAC rate of 320 kbit/s corresponding to 10 full duplex bearers, is lost due to static overhead), the DECT MAC layer procedures and loading reduces the maximum achievable throughput to slightly more than 30 seg/s (122.88 kbit/s). The retransmission processes in the MAC and DLC layers further cause the limiting throughput to fall down to about 25 seg/s (102.4 kbit/s). The TCP layer timers associated with segments acknowledgements appear not to expire in all the simulations carried out; this happens because the round trip time tracking algorithm and RTO estimation process manage to adapt to the varying delays across the air interface (due to retransmission). On the other hand, even with the doubled BER scenario, the MAC and DLC recovery procedures described in the previous Section manage to be successful.

The delay performance of the TCP user are shown for two cases in Figure 6. The plot on the left hand side shows the histogram of the TCP segment queuing plus transfer delay, *D*. Note that about 12% of samples fall beyond 10 s. The most noteworthy result of this graph is that, even though the average of *D* is rather low (below 1 s) there is a relevant amount of segments that are delayed several seconds, i.e. the distribution of D seem to exhibit the so called heavy tail phenomenon. This is confirmed by the fact that the negative exponential distribution with the same mean has a remarkably faster decaying tail. This is the more surprising as one considers that Poisson arrivals have been generated for the TCP segments! Clearly, much higher delays result in case of batch Poisson arrivals, where with the same $\lambda=20$ seg/s, about 20% of samples of *D* fall beyond 25 s.

Finally, the graph on the right of Figure 6 shows a sample path of transfer delays of TCP segments across the air interface in the reference case, with $\lambda=20$ seg/s and 10 full duplex bearers. In this situation, 20 ms are required at least to transfer a TCP segment. In the central 5 seconds of the run (from TDMA frame 500 to 1000) the BER has been increased by an order of magnitude. In this way, we get SER=0.127 in the G state and SER=0.913 in the B state. The transfer delay exhibits a significant increase as well, even though no TCP timeout occurs.

Our interpretation of the presented performance results, though referring to a limited set of cases, is that the throughput loss with respect to a wired access of the same nominal capacity is tied to the specific air interface and not to internal TCP mechanism that work improperly in the radio environment. On the contrary, delay may be an issue for the provision of real time services (e.g. browsing). Many more simulations are needed to definitely confirm this intuition, especially a comparison with a different approach, where no special means for error recovery are set up in the radio access lower layers (e.g. use of In MAC channel and DLC LU1 service), thus reducing overhead to essentially zero, but relying only on TCP error recovery.

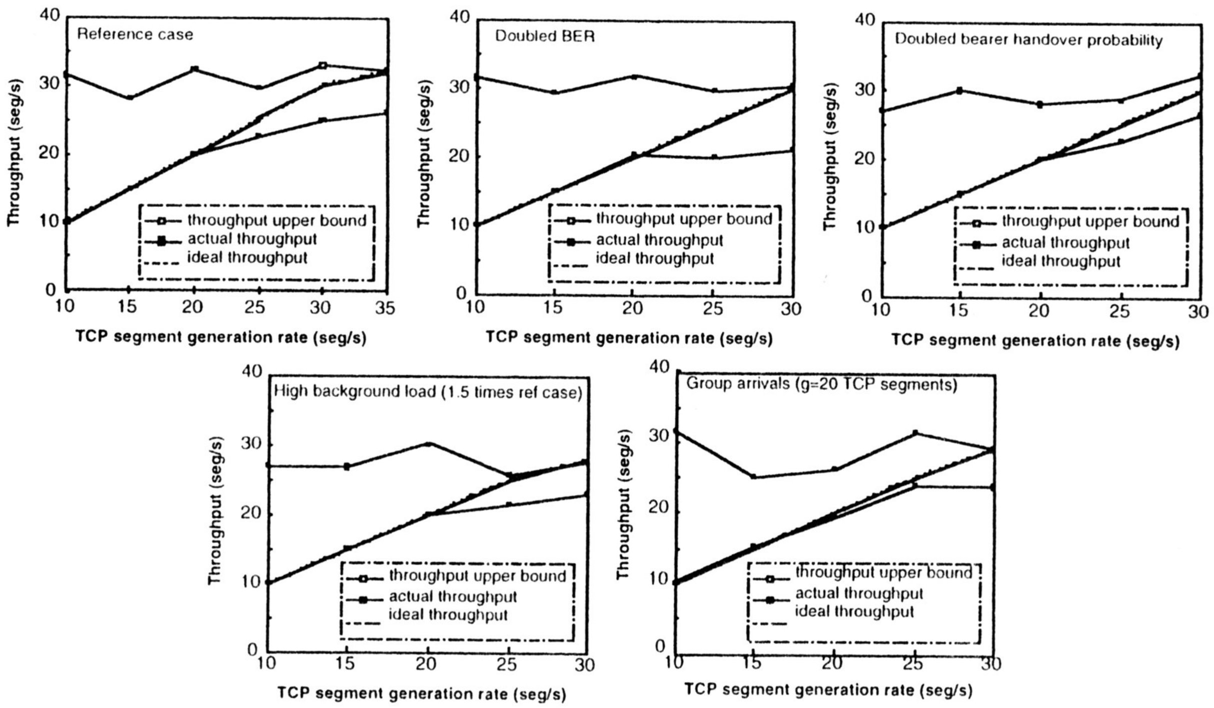

Figure 5 - TCP user throughput versus the TCP segment generation rate λ in various cases.

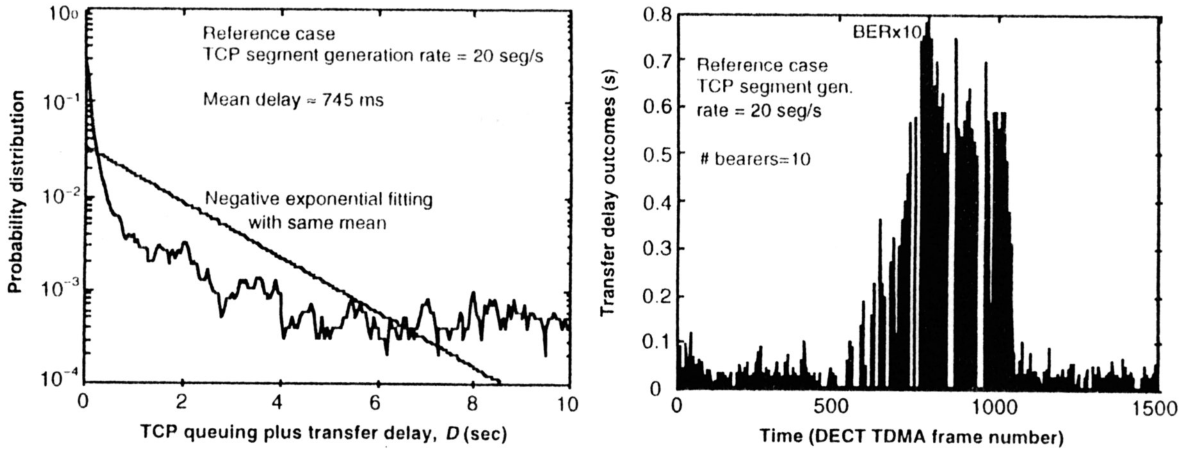

Figure 6 - TCP user delay performance.

References

[1] "Digital European Cordless Telecommunications", ETSI Standard ETS 300 175 Parts 1÷9, 2nd edition, 1996.

[2] P. Wong, A. Lasa, F. Halsall, G. Schultes: "Performance of multibearer connections for varied data services in a TDMA system", proceedings of the IEEE ICC'94, New Orleans, 1-5 May 1994, pp. 593-597.

[3] G. Bao: "Performance evaluation of TCP/RLP protocol stack over CDMA wireless link", Baltzer Wireless Networks Journal, March 1996, Vol. II, pp. 229-237.

[4] R. Caceres, L. Iftode: "Improving the performance of reliable transport protocols in mobile computing environments", IEEE Journal on Selected Areas in Communications, June 1995, Vol. 13, n. 5, p. 850.

[5] P. Manzoni, D. Ghosal, G. Serazzi: "Impact of mobility on TCP/IP: an integrated performance study", IEEE Journal on Selected Areas in Communications, June 1995, Vol. 13, n. 5, pp. 858-867.

[6] M.C. Chuah, O.C. Yue, A. De Simone: "Performance of two TCP implementations in mobile computing environments", proceedings of the IEEE Globecom'95, Singapore, 13-17 November 1995, pp. 339-344.

[7] H. Herbrig, R. Rheinschmitt: "DECT/ATM based UMTS demonstration and trial system", Proceedings of the first Mobile Summit, Granada (Spain), 27-29 November 1996, Session B3.

[8] W.R. Stevens: "TCP/IP illustrated. Vol. 1: the protocols", Addison Wesley, 1994.