ETSI TECHNICAL REPORT

ETR 226

October 1995

Source: ETSI TC-RES

Reference: DTR/RES-10-06

ICS: 33.060.20

Key words: LAN, radio

Radio Equipment and Systems (RES); HIgh PErformance Radio Local Area Networks (HIPERLAN); Architecture for Time Bound Services (TBS)

ETSI

European Telecommunications Standards Institute

ETSI Secretariat

Postal address: F-06921 Sophia Antipolis CEDEX - FRANCE
Office address: 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE
X.400: c=fr, a=atlas, p=etsi, s=secretariat - **Internet:** secretariat@etsi.fr

Tel.: +33 92 94 42 00 - Fax: +33 93 65 47 16

Contents

Blank page

Foreword

This ETSI Technical Report (ETR) has been produced by the Radio Equipment and Systems (RES) Technical Committee of the European Telecommunications Standards Institute (ETSI).

ETRs are informative documents resulting from ETSI studies which are not appropriate for European Telecommunication Standard (ETS) or Interim European Telecommunication Standard (I-ETS) status. An ETR may be used to publish material which is either of an informative nature, relating to the use or the application of ETSs or I-ETSs, or which is immature and not yet suitable for formal adoption as an ETS or an I-ETS.

Introduction

HIgh PErformance Radio Local Area Network (HIPERLAN) is a short range radio-communications sub-system intended for use with computer systems. As used in this document the term "HIPERLAN" refers to the specification as given in ETS 300 652 [2].

HIPERLAN offers users at least 10 Mbit/s service data rate and supports a range of services such as asynchronous Local Area Network (LAN) and Time Bounded Services (TBS). Due to the performance of HIPERLAN, many different services (usual data transfer, voice, video, multi-media) are likely to share the single medium of HIPERLAN.

This ETR provides a description of an architecture to support TBS with HIPERLAN.

This ETR is structured as follows:

- clause 4 reviews the overall HIPERLAN architecture;
- clause 5 lists and parameterises the different TBS which are likely to operate on HIPERLAN;
- clause 6 is a short survey of existing techniques to ensure access delay constraints in a distributed transmission system;
- clause 7 proposes a HIPERLAN architecture for the data transfer service. A general architecture is developed and many options are presented. Actually the proposed architecture is not only reserved for time bounded service but for all the data transfer. The architecture proposes a unification and handles all packets using the Medium Access Control (MAC) facilities in the same way;
- clause 8 present guidelines for the utilisation of TBS;
- clause 9 presents a general conclusion of this study;
- annex A is a technical annex where models are presented. Analytical models and a simulation model are both presented and results of these models and simulations help to validate the architecture.

Blank page

1 Scope

This ETR investigates the architectural design of HIPERLAN regarding the support of Time Bounded Services. The ETR continues the architecture design provided in the HIPERLAN System Definition see ETR 133 [1].

The HIPERLAN architecture described herein is confined to the lower two Open Systems Interconnection (OSI) layers. Functions of higher layers are required for operation and interworking of a complete system and are outside the scope of HIPERLAN. Therefore they are not considered in this ETR.

The architecture described in this ETR serves as the basis and reference for the draft prETS 300 652 [2].

2 References

For the purposes of this ETR, the following references apply:

[1] ETR 133: "Radio Equipment and Systems (RES); HIgh PErformance Radio Local Area Networks (HIPERLAN); System definition".

[2] prETS 300 652: "Radio Equipment and Systems (RES); HIgh PErformance Radio Local Area Network (HIPERLAN); Functional specification".

[3] ISO 7498: "Information processing systems - Open Systems Interconnection - Basic Reference Model".

[4] ISO/IEC 10039 (1991): "Information technology - Open Systems Interconnection - Local area networks - Medium Access Control (MAC) service definition".

[5] ISO 8802-2 (1989): "Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 2: Logical link control".

[6] "Rate Monotonic", Liu and Layland. ACM 1973.

[7] "Message-Based Priority Functions In Local Multiaccess Communication Systems", Raphael Rom, Fouad Tobagi. North Holland Publishing Company Computer Networks (1981) pp 273-286.

[8] "Data Networks", Bertsekas, Gallager, Prentice Hall. 1988.

[9] "Supporting Real-Time Application in an Integrated Services Packet network: Architecture and Mechanism", David D. Clark, Scott Shenker, Lixia Zhang. ACM 1992.

[10] "Dynamic Bandwidth Allocation in a Network", Kurt Maly, C. Michael Overstreet, Xia-ping Qiu, Deqing Tang. ACM 1988, pp 13-24.

[11] "A Versatile Access Scheduling Scheme for Real-Time Local Area Networks", Ken Chen. INFOCOM 1992.

[12] IEEE Std 802.4: "Information Processing Systems - Local Area Networks - Part 4: Token-Passing Bus Access method and Physical Layer specifications".

[13] IEEE Std 802.5: "Information Processing Systems - Local and Metropolitan Area Networks - Part 5: Token Ring Access methods and Physical Layer Specifications".

3 Definitions, abbreviations and symbols

3.1 Definitions

For the purposes of this ETR, the following definitions apply:

acknowledgement: The operation in which the (one hop) destination node transmits after the packet a low bit rate burst containing some information to acknowledge the reception of the packet.

acknowledgement overhead: The duration of the acknowledgement burst.

ad-hoc network: Applications of HIPERLAN in which no fixed network infrastructures exist.

asynchronous service: The service provided by HIPERLAN to support asynchronous traffic between HIPERLAN nodes.

asynchronous traffic: Data traffic that characteristically has a statistical arrival and delay distribution. This typifies most LAN data traffic.

attached subnetwork: A subnetwork to which a HIPERLAN subnetwork is attached for the purpose of communication to either an end system or another subnetwork.

block: An elementary subset of interleaved, coded data in a packet.

block length indicator field: The field in the information data in a packet which indicates the number of blocks in the packet.

channel: An instance of medium use that can coexist with other instances of medium use, with each providing service to a separate set of HIPERLAN nodes.

channel cycle: Period of time on the channel during which each contending node attempts at most one channel access.

checksum field: The field in the information data in the packet which contains the octets used to check the correct reception and integrity of the useful data in the packet.

connection set: A subset of HIPERLAN nodes (within one or more HIPERLANs) that are in direct radio communication range of each other.

Data Link Control (DLC): Layer 2 of the ISO/OSI reference model.

data rate: Instantaneous user data rate at the MAC/DLC layer interface. This is an average rate over a multi-symbol burst of activity, and not an average over a time-period including several bursts.

data transfer service: MAC service handling packet from MAC user to transmission.

destination HIPERLAN Physical Service Access Point (HPhSAP) field: The field in the information data of the packet which indicates the physical address of the destination (one hop) of the packet.

elimination phase: Phase in channel access mechanism which follows priority phase.

elimination overhead: Duration of the elimination phase in channel access mechanism when packet transmission follows in the same channel cycle.

end system: A system which contains application processes, which from an OSI point of view are considered as sources and sinks of information. Communication protocols are expected to support the needs of these applications processes.

forwarding: A dynamic routing mechanism that enables a HIPERLAN node to provide connectivity within one single HIPERLAN.

NOTE: forwarding operates independently of the underlying physical medium.

HIgh PErformance Radio Local Area Network (HIPERLAN): A HIPERLAN is a set of HIPERLAN nodes that have the same HID, and which have the logical ability to interchange traffic. The term is also used to denote a situation where nodes with the same HIPERLAN IDentifier (HID) form a network.

HIPERLAN IDentifier (HID): A HIPERLAN subnetwork identifier used to differentiate HIPERLANs from each other. All members of a given HIPERLAN use the same HID.

HIPERLAN sharing: When two HIPERLAN nodes are able to exchange messages over the radio medium, they are sharing the medium. Nodes that are not in radio contact with each other (either direct or by means of forwarding), are not sharing the medium. If two nodes that are not sharing the medium and have the same HID get within radio range, a HIPERLAN sharing function resolves the conflict (see also: Medium).

HIPERLAN MAC Protocol Data Unit (HMPDU) length indicator field: The field in the information data in the packet which indicates the number of octets of useful data in the packet (excluding padding).

HMPDU type indicator field: The field in the information data in the packet which indicates the HMPDU type to which the packet corresponds.

HMPDU residual MPDU life time: The field in the information data in the packet which indicates the Residual Transit Delay (RTD) of the packet.

HMPDU destination address: The field in the information data in the packet which indicates the address of the terminal destination of the packet (multi-hop).

HMPDU source address: The field in the information data in the packet which indicates the address of the original source of the packet (multi-hop).

HMPDU sequence number field: The field in the information data in the packet which indicates the sequence number of the packet assigned by the original source node.

HMPDU user priority and lifetime: The fields in the information data in the packet which indicate the User Priority (UP) and the Transit Delay (TD) of the packet assigned by the original source node.

HMPDU key identifier and initialisation vector: The fields in the information data in the packet which indicate the key and the initialisation vector used by the original source node to encrypt the packet.

HMPDU sanity check: The field in the information data in the packet which contains octets used to check correct encryption of the packet.

identifier field: The field in the information data in the packet which indicates the HIPERLAN identifier (HID).

interworking: Interaction between dissimilar subnetworks, end systems, or parts thereof, providing a functional entity capable of supporting end-to-end communications.

Local Area Network (LAN): A group of user stations each of which can communicate with at least one other using a common transmission medium commonly managed.

MAC Service Data Unit (MSDU): The unit of data delivery between MAC users.

Medium Access Control (MAC): The sub-layer of the ISO 8802-2 [5] reference model between the Physical Layer and the Logical Link Control (LLC).

packet: The set of data modulated in a single and continuous burst of energy transmitted by a node.

padding octet: Octets of non significant bits added to the useful data octets in order to make the overall data in the packet a multiple of blocks with total length above a certain minimum threshold.

padding length indicator field: The field in the information data in the packet which indicates the number of padding octets inserted in the packet.

Physical Layer (PHY): Layer 1 of the ISO/OSI reference model. The mechanism for transfer of symbols between HIPERLAN nodes.

Protocol Data Unit (PDU): Data unit exchanged between entities at the same ISO layer.

priority phase: First phase of channel access mechanism corresponding to priority assertion and contention.

priority overhead: Duration of the priority phase in channel access mechanism.

selection phase: The phase which terminates the channel access mechanism just after elimination phase.

selection overhead: Duration of the selection phase in the channel access mechanism when packet transmission follows in the same channel cycle.

source HPhSAP field: The field in the information data in the packet which indicates the physical address of the node which transmits (one hop) the packet.

training sequence overhead: Length of the phase in the packet transmission used by the receiver to adapt to radio distortion.

3.2 Abbreviations

For the purposes of this ETR, the following abbreviations apply:

CAM	Channel Access Mechanism
CD	Critical Delay
DAM	Don't Acknowledge Me QoS parameter
DLC	Data Link Control
DP	Discard Parameter
DTD	Desired Transfer Delay
EDF	Earliest Deadline First
EY-NPMA	Elimination-Yield Non-pre-emptive Priority Multiple Access
HID	HIPERLAN IDentifier
HIPERLAN	HIgh PErformance Radio Local Area Network
HMPDU	HIPERLAN MAC Protocol Data Unit
HPhSAP	HIPERLAN Physical Service Access Point
LAN	Local Area Network
LBH	Low Bit rate Header
LLC	Logical Link Control
MAC	Medium Access Control
MSDU	MAC Service Data Unit
MP	MSDU Priority QoS parameter
MPDU	MAC Protocol Data Unit
MPEG	Moving Pictures Expert Group
MSDU	MAC Service Data Unit
NTB	Not Transmit Before
OSI	Open Systems Interconnection
PDU	Protocol Data Unit
PHY	Physical Layer
QoS	Quality of Service
RTD	Residual Transit Delay
TD	Transit Delay
TB	Time Bounded
TBS	Time Bounded Services
UP	User Priority

3.3 Symbols

For the purposes of this ETR, the following symbols apply:

i_E	duration of the elimination slot interval
i_{EVS}	duration of the elimination survival verification interval
i_{PA}	duration of the priority insertion interval
i_{PS}	duration of the prioritisation slot interval
i_{YS}	duration of the yield slot interval
xi	delay threshold for the i^{th} CAM priority assignment

4 HIPERLAN Architecture

4.1 General overview

The following description of the HIPERLAN systems architecture assumes knowledge of the OSI Reference Model and HIPERLAN Services and Facilities. See ISO 7498 [3] for a description of the OSI Reference Model and the conceptual division of communications functions into their respective layers. See also ETR 133 [1] and prETS 300 652 [2] for details of HIPERLAN services and facilities and the general reference model for HIPERLAN. In addition, reference is made to support of the ISO MAC service in ISO/IEC 10039 [4] and ISO 8802-2 [5]. The implications of support of the ISO MAC service are assumed to be understood and are not repeated here unnecessarily.

Figure 1 below presents the overall HIPERLAN architecture described in the System Definition Document ETR 133 [1]. This architecture design shows that this ETR will develop the data transfer services with respect to the support of Time Bounded Services (TBS). Integration of TBS is the subject of numerous publications, see Clark *et al.* [9] and related references.

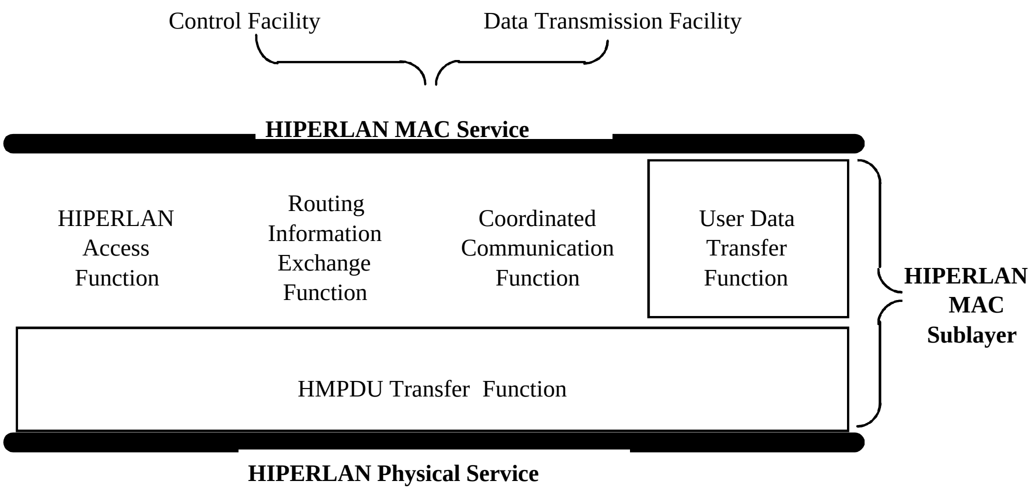


Figure 1: Overall HIPERLAN architecture

4.2 LLC and MAC operations

4.2.1 LLC operations

Connection establishment is beyond the scope of HIPERLAN. It is up to the MAC service to determine *a priori* knowledge of the environment from the MAC service provider and to decide whether a connection can be supported based on such knowledge.

4.2.2 MAC operations

According to the ISO/IEC 10039 [4] the MAC service makes a best effort attempt to satisfy each request. The translation of the Quality of Service (QoS) of a packet in Channel Access Mechanism (CAM) priority levels helps to meet this request.

The architectural choices at the MAC level should be such that effective control is possible by the DLC or by any other end system.

5 General requirements for Time Bounded Services in HIPERLAN

5.1 Review of potential Time Bounded Services supported by HIPERLAN with their requirements

5.1.1 Internal services

Internal services are all of the services that HIPERLAN uses to manage the network and to ensure its connectivity. Generally internal services produce internally generated PDUs. Internal services include the following: membership protocols, allocation (or reallocation) of channels and relaying. The following subclause is devoted to relaying.

5.1.2 Relaying

The duty of this service is to maintain the connectivity of the network. It must react in real-time against any topology changes in the network. TBS, like other data transfer services, uses forwarding services. All of these considerations imply that relaying operations are to be considered as Time Bounded (TB) internal services and receive a high priority in the MAC operations.

It can be noted that the traffic caused by forwarding management is completely MAC dependent and therefore its load can be controlled by normalisation (for example not to exceed 10 % of the total channel capacity).

5.1.3 External services

5.1.3.1 Voice

Upper bounds on packet delays range between 10 ms to 50 ms. The throughput requested by voice ranges from 1 kbit/s to 64 kbit/s per connection, depending on the expected quality and possible utilisation of compression algorithms.

5.1.3.2 Video

The throughput requested by a video transmission is in the region of a few hundred kbit/s. Upper bounds on the packet delays are within 100 ms.

5.1.3.3 Real-time traffic

Asynchronous traffic may be also time constrained. For example alarms and more generally all the packets involved in real-time operations are TBS. Examples of such processes exist in control command systems.

5.1.3.4 Purely asynchronous traffic

Some traffic is usually considered as not time constrained, for example electronic mail or automatic file management. Nevertheless, for standardization convenience such traffic can be considered as TBS with default parameters.

The parameters given above can be found in DTR/SMG-050201 (see annex B) and are summarised in table 1.

Table 1

	Throughput	Delay	Bit error rate
Voice	2 kbits	40 ms	10^-3
Voice high quality	32 kbits	40 ms	10^-3
Video telephony	64 kbits	100 ms	10^-6
Video conference	384-768 kbits	100 ms	10^-6
File transfer		500 ms	10^-6

6 Review of the existing tools to ensure time constraint in a distributed system

6.1 Supporting Time Bounded applications as a scheduling problem

HIPERLAN is a radio network. The HIPERLAN MAC can be considered as a server. A packet delivery can be considered as a task that the server HIPERLAN MAC has to execute, see figure 2.

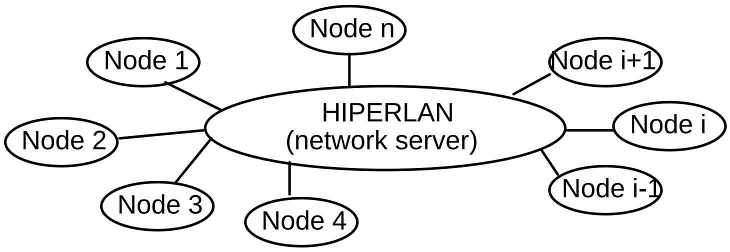


Figure 2: HIPERLAN MAC modelled as a server

Notice that figure 2 is only an abstraction. Indeed HIPERLAN is not a centralised network and moreover the server attached to HIPERLAN is capable of performing several parallel services when several simultaneous transmissions are possible (spatial reuse). This detail is just a straightforward sophistication of the usual model where the multiple access LAN is considered as a single server which provides services, in our case transmission to nodes.

TBS are characterised by the delivery of packets according to certain timing constraints. Therefore this is a scheduling problem and the issues rely on the fact that the instantaneous resource that the server can offer to clients is finite. The resource is actually the multiple access channel: a 10 Mbit/s cable in Ethernet, a 23,529 4 Mbit/s radio link with possible spatial reuse in HIPERLAN.

The intuitive idea is that a LAN should transmit a TB packet first before a packet which is not time constrained and should transmit a packet with a short deadline before a packet with a longer deadline. In this scenario, the likelihood of meeting the deadline of the TB packet is higher than in a scenario with no control. In the following, two well-identified techniques which are known to offer potential solutions to the scheduling under time constraints are described.

The description focuses on packet timing expressed in terms of deadlines. Timing constraints expressed in terms of time windows are also possible but are out of the scope of HIPERLAN MAC modus operandi.

6.2 Scheduling transmission in a network

This subclause gives a short survey of scheduling tools.

6.2.1 Earliest deadline first algorithm

The Earliest Deadline First (EDF) algorithm [6] is one of the most familiar scheduling techniques. This scheduling algorithm simply favours the packet holding the shortest deadline. Many optimality results concerning this algorithm have been shown especially in centralised systems IEEE Std 802.5 [13]. Obviously such an algorithm is likely to help to keep packet delays within a given window.

6.2.2 Priority functions technique

6.2.2.1 General presentation

This technique requires a special tool in the Channel Access Mechanism (CAM), called CAM priorities. CAM priorities are non pre-emptive priorities: any node automatically defers before any other node about to transmit a packet with a higher priority packet. In other words, the CAM automatically favours the packet of higher priority in any access conflict with several nodes (see figure 3). More precisely the performance of packets of a given traffic must not be affected by traffic of a lower priority [7][8].

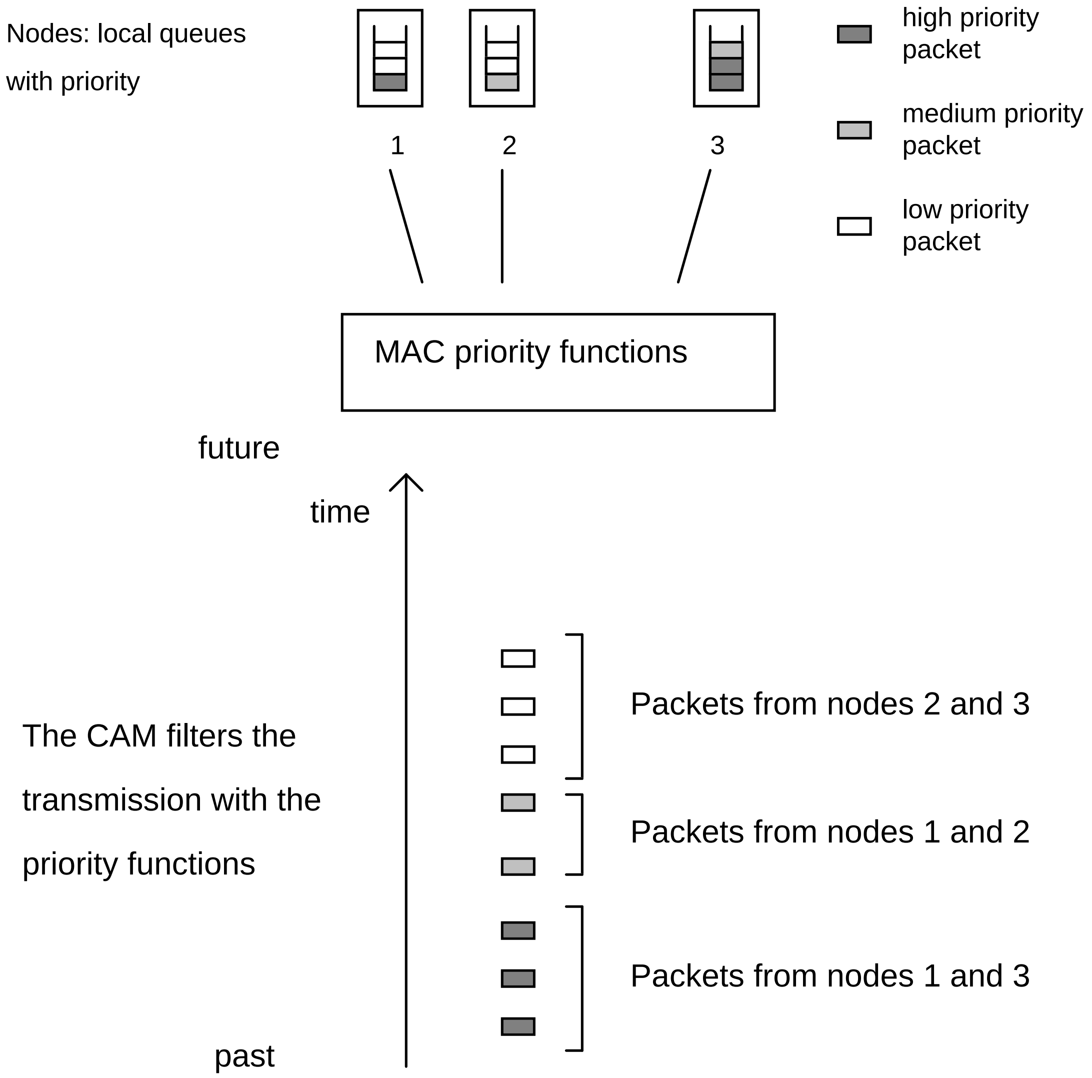


Figure 3: Priority functions scheduling technique

It may be noticed that this model always gives better performance to high priority packets than to low priority packets. Therefore such a tool may help to achieve packet timing even with little information on the global state of the system.

Performance of a system with priority levels is studied in annex A, subclause A.2.2.

6.2.3 Declaration technique

This technique is in a sense the opposite of the previous technique. It strives to build the most precise view of the global state of the system as the basis for the scheduling algorithm [10], [11]. It is based on the exchange of information between all the connected nodes about their local waiting queues in order to update a distributed knowledge of the global state of the system. To achieve that goal, the transmission resource itself must be used to exchange such information. On this basis, any consensus algorithm can be used to organise the transmission.

An example is given below of a declaration technique mechanism. It is assumed that each node broadcasts to the other nodes the timing parameters (deadlines and lengths) of their packets in the waiting queue. With the knowledge of the characteristics of all the packets in the system, each node knows which packet must be transmitted first and consistently schedules its transmissions, see figure 4.

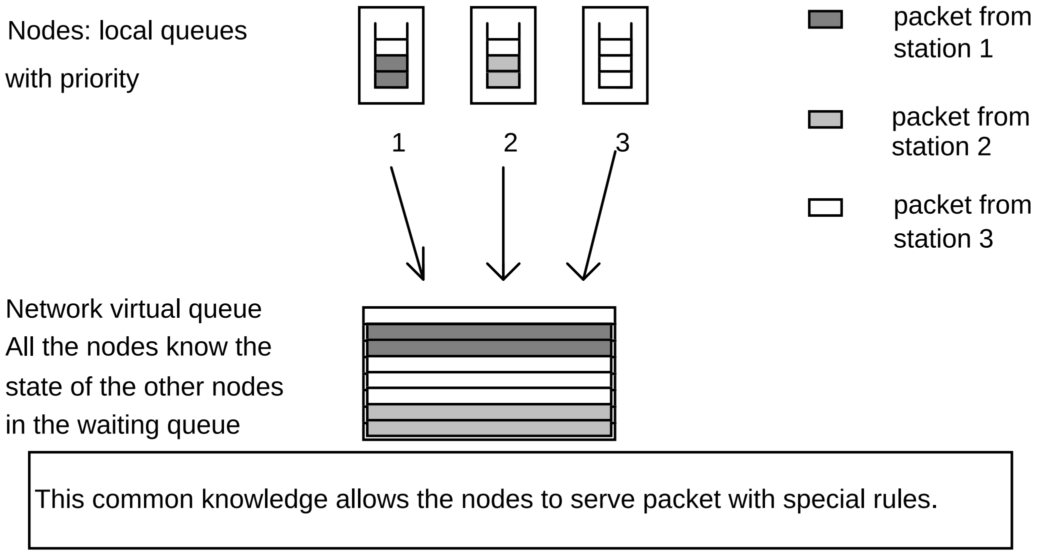


Figure 4: Declaration scheduling technique

7 HIPERLAN Architecture overview for Time Bounded Services

7.1 General HIPERLAN architecture and basic principles

7.1.1 MAC Modus Operandi

MAC operates on a per packet basis and provides the best effort to achieve packet delivery to the user destination(s). The only possible actions of MAC on a packet are: defer, transmit or reject. The user indicates the Quality of Service attached to the packet via QoS parameters.

MAC transmits a packet as soon as possible, and may defer transmission only for internal MAC considerations. It is the user's responsibility to sequence and re-sequence packets before and after MAC submission.

7.1.2 Unified data transfer service

A generic approach is used which assumes that all packets submitted to the MAC hold time constraints which may be parametrised with default values. Therefore TBS are merged into a unified data transfer service, which treats all requests to the MAC in a unified way. Figure 5 shows the general HIPERLAN architecture related to the data transfer service.

The QoS parameters contain a deadline parameter expressing the validity limit of the packet in the MAC. Packets which missed their deadlines are automatically discarded. An indication to the MAC user of QoS failure is generated in such a case. The packets are delivered to the MAC user via an indication.

The QoS parameters contain a priority expressing the relative importance of the packet against other packets for the access to the shared resource when the demand exceeds the supply. The impact of such a parameter is expected at high load periods.

7.1.3 QoS algorithm

The HIPERLAN architecture dedicated to Time Bounded / data transfer services addresses in a unified way the two main QoS parameters: deadline and MAC Service Data Unit (MSDU) priority. The tools used by the data transfer service are a CAM priority mapping and a scheduling in the local queue.

The scheduling algorithm is based on an EDF-like algorithm. The determination of the CAM priority depends on the deadline and the MSDU priority. Although the impact of the deadline and MSDU priority parameters are expected to occur in different scenarios, it is possible to mix these two parameters in the scheduling and the CAM priority mapping.

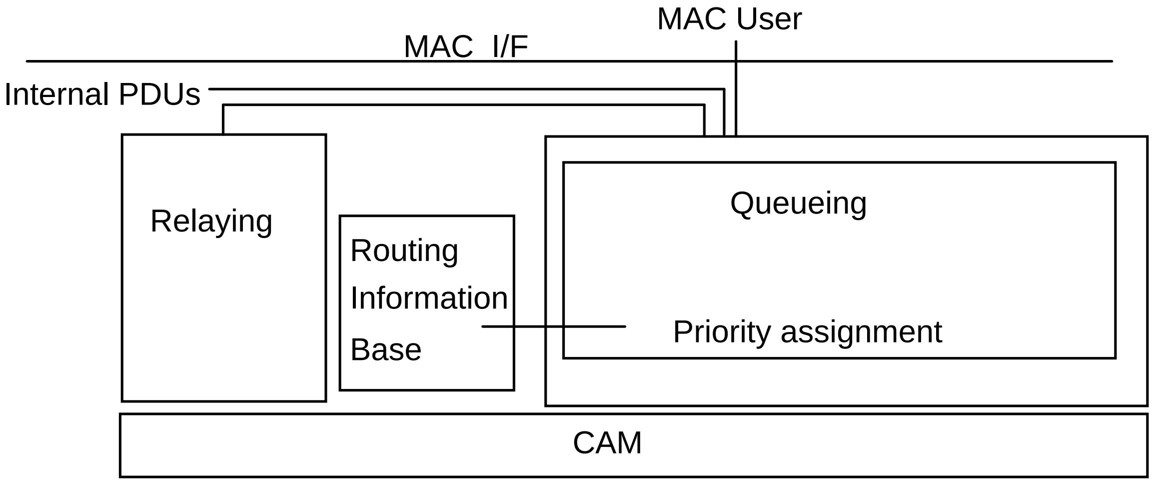


Figure 5: The unified data transfer service

7.2 Presentation of the architecture

The following four points are presented:

a) the interfaces of the data transfer service with its high and low contact layer: MAC user, relaying service and CAM layer;

b) the set of QoS parameters which are the basic input of the data transfer algorithm. Two main parameters are defined: the deadline (Desired Transit Delay) parameter and the MSDU priority parameter;

c) the CAM priority assignment. A mapping function is defined between the QoS parameters and the CAM priority levels;

d) the packet selection to CAM. The packet selected by the MAC to be the candidate to the CAM is among the packets holding the highest CAM priority.

The presentation focuses on the main points which are merely adopted as a working assumption by the committee. Comments and outlines are given. Alternative techniques are also discussed throughout.

7.3 Data transfer interface

7.3.1 CAM interface, data transfer packet header and other packet parameters

The CAM indicates ends and beginnings of channel cycles.

The submission of a packet to CAM is done in terms of a request with the packet CAM priority as a parameter. Other parameters such as length or acknowledgement request are also sent. Only one packet may be a candidate for the CAM.

The CAM confirms the request by the four possible alternatives in the CAM: successful transmission, unsuccessful transmission, elimination in contention, elimination in CAM priority. In the following it is assumed that the confirmation addresses the request without ambiguity (through implicit common references).

The updated and initial QoS parameters are inserted in the packet header. In particular the remaining time to live, the Residual Transit Delay (RTD), is put in the header so that timing of packet can be maintained in a consistent way in a multi hop context without the need for a common time reference.

A successfully transmitted packet is destroyed in the data transfer queue.

7.3.2 Additional or alternative features for the CAM interface

A Retry parameter can be attached to each packet and is incremented at each transmission attempt.

Broadcast or multicast packet could be acknowledged and a lack of energy after packet transmission could be interpreted as an unsuccessful transmission and the packet confirmed by CAM. Alternatively broadcast packets could be negatively acknowledged. In this case a presence of energy after the packet transmission could be interpreted as an unsuccessful transmission.

Non-acknowledged broadcasts could be transmitted several times in order to increase reliability. The initiative of such duplication should be left to the data transfer user.

7.3.3 Data transfer interface: packet insertion

Packets are inserted in the data transfer queue as long there is room available. Packets come either directly from the MAC user or indirectly from the forwarding function when the host node is a forwarder. Packets internally generated by MAC (for example PDU for relaying information exchange and management) will be treated differently.

The data transfer user submits packets in the form of a request. The request carries the QoS parameters. If none, the default QoS parameter will be assigned.

The data transfer service confirms the request by one of the following alternatives: packet transmitted or packet rejected. If the packet is rejected an additional parameter specifies the reason for the rejection: rejection because of resource, rejection because of QoS parameters, rejection because of congestion. For this last cause of rejection an optional filter will operate. The filter uses a basic heuristic to tell whether a request is feasible or not and to reject it in consequence.

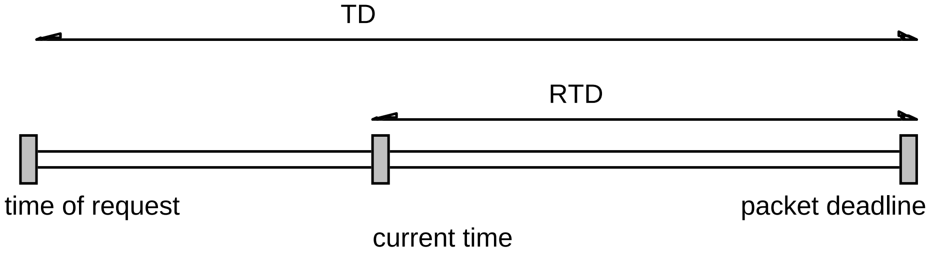
7.4 QoS parameters

In ETR 133 [1] the QoS parameters include: maximum transfer delay, delay variance and a discard parameter. In ISO 8802-2 [5] the LLC user provides the MAC, in addition the QoS parameters already described, a user priority. From this panel the committee has selected two basic QoS parameters: the desired transfer delay and the MSDU priority.

According to the subclause 7.3, the maximum transfer delay will be updated during the life time of the packet as a Residual Transfer Delay indication. Together with the MSDU priority, these parameters will be the basic input of the scheduling algorithm.

7.4.1 The transit delay

The first parameter which will be desired is the maximum Transit Delay (TD). This parameter will be updated during the lifetime of the packet in MAC and the update of the TD is the residual transit delay (RTD). A packet is discarded when its RTD expires. The default TD value is 500 ms, see figure 6.



Packet timing parameters initial and updated

Figure 6: Transit delay QoS parameter

7.4.2 The MSDU priority

The MSDU Priority (MP) indicates the relative importance of the packets. This parameter may be used to select the packet to be submitted to the CAM and to determine its CAM priority. There are two values: 0 = normal; 1 = high. The default value is 0.

7.4.3 Set of Quality of Service parameters

The following set of QoS parameters will be specified at each packet request made by the MAC user: (TD, MP). During the life time of the packet there is the following set: (TD, RTD, MP).

7.4.4 Notes on QoS parameters

The above QoS parameters correspond to the basic expected actions of MAC over a packet: transmit, defer or discard. The TD parameters means Transmit before TD or discard after TD; The User Priority means Defer or Discard before higher UP.

Note that the two main QoS parameters correspond to two different scenarios. When the demand on the channel is well below the channel capacity the data transfer operations are merely based on the TD parameter to get all packets within their deadline. On the other hand, when the demand on the channel exceeds the channel capacity the scheduling algorithm uses the MP parameter to detect which packet will be transmitted and which one will be jettisoned.

7.4.5 Additional or alternative QoS parameters

7.4.5.1 The Discard Parameter (DP)

The discard parameter is related to the maximum statistical loss on a given packet traffic. Such a parameter is difficult to manage on a per packet basis because one should maintain statistics on sequence of several packets.

A natural form of this parameter would be a loss probability but it is known that this is not manageable on a per packet basis. MAC should map this parameter into a Critical Delay (CD). This critical delay should be subtracted from the RTD before the application of the scheduling algorithm. Therefore a packet holding a higher DP delay would have a greater chance of being selected and a higher chance of being transmitted before the expiration of its real RTD.

7.4.5.2 The Don't Acknowledge Me (DAM) parameter

The MAC user can require in the QoS parameters that the packet is not MAC acknowledged. This option may avoid a useless duplication of services between MAC and LLC with too many embedded acknowledgements. This may substantially improve the channel utilisation if the reliability is suitable. Such an option is also suitable for future adaptive LLCs. The default value of the DAM parameter is negative (acknowledgement required) for point to point packets.

7.4.6 Refinement on QoS parameters

The MSDU priority parameter may have more than two potential values. For example: 1 for high, 0 for normal, -1 for low1, -2 for low2, etc. The alternative normal / low provides the user with the possibility to handle traffic with mixed priority packets. An interesting example is with MPEG where intermediate pictures (I pictures) are computed by reference to key pictures (K pictures). In this perspective the user should send K pictures under priority 0, and I pictures under priority -1. High priority video transfer should shift priority by one level: priority 1 for K picture and priority 0 for I pictures.

The obvious alternative is to set several levels of high priority: high1, high2, etc. However, this strategy is not preferable to the previous one because of CAM priority assignment (which will be discussed in next clause).

An interesting feature is to set the highest priority as the default value. The advantage is that it prevents abusive use of the highest priority level in case of overlapping but non co-operating HIPERLANs, and still offers the operating system the flexibility of priority use. The disadvantage is that it does not offer a real high priority level with restricted access.

7.5 CAM priority assignment

7.5.1 Basic principles

A mapping function will be defined between the QoS parameters and the CAM priority levels. This function takes as input variable the number of CAM priority levels, the set of QoS parameters and other MAC information, and returns a CAM priority level. The MAC information will be the number of hops between the nodes to the packet destination(s). When this number is not known, a default value of 1 hop is assumed.

The packet selected by the MAC to be the candidate for the CAM will be among the packets holding the highest CAM priority.

7.5.2 CAM priority assignment

Each packet receives a CAM priority level which is a function of the ratio R of the RTD over the number of hops to reach its destination. A figure of four CAM priority levels has been considered as a reasonable assumption for HIPERLAN. The mapping consists in defining an increasing function F(R).

The CAM priority will be F(R) + MSDU priority with the convention that negative values map to zero. Other CAM priority mappings will be described as alternatives in a separate subclause, see figure 7.

The description of a suitable function F(R) on five priority levels consists of determining three decreasing time thresholds x1, x2, x3 and x4. The following rules apply:

if R<x4 then F(R)=3;

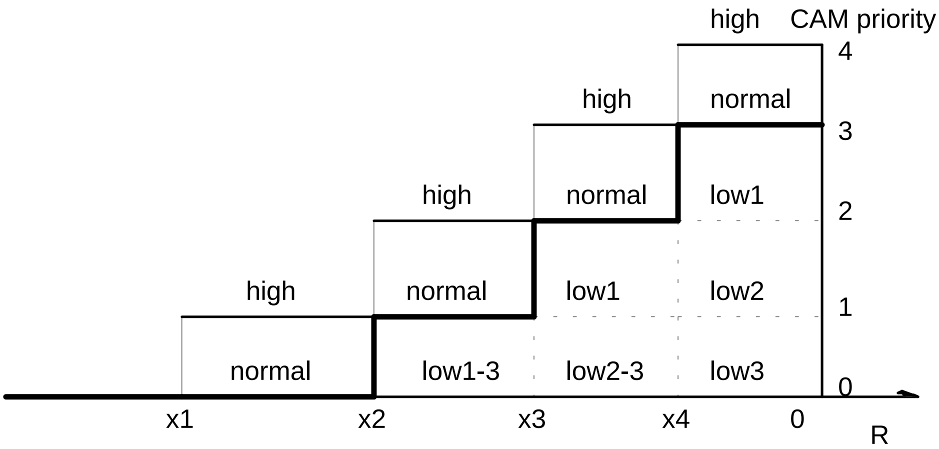
if x4<R<x3 then F(R)=2;

if x3<R<x2, then F(R)=1;

if x2<R<x1, then F(R)=0;

if R>x1, then F(R)=-1.

Tentative values for these thresholds have been set at x4=10 ms, x3=20 ms, x2=40 ms and x1=80 ms.



CAM priority mapping with normal and high MSDU Priorities
optional low1, low2 and low3 MSDU priorities are indicated

Figure 7: CAM priority assignment

7.5.3 Note on CAM priority assignment

7.5.3.1 Indications of the impact of TD and MP QoS parameters

In low load conditions the packets will be transmitted almost immediately without any significant update of their RTD.

In heavy load conditions the packets will see significant updates of their RTD but the CAM priority assignment will be as follows: packets with low RTD will receive high CAM priority so that the EDF scheduling algorithm will operate efficiently.

In overload conditions some packets will be delayed until expiration of their RTD and therefore will be discarded. If the EDF algorithm is appropriately used, then the discard of packets will tend to be independent of their initial TD and services. Nevertheless the user may want to tell the MAC which packet he would like to discard and which one he wants to favour. This is the role of the MSDU Priority parameter. The MSDU Priority parameter tells which packet can be sacrificed in case of the global channel overflow. When the overflow is of a significant duration, packets are driven to their terminal CAM priority. Therefore an efficient use of the MSDU priority parameter requires determining of the value of the terminal packet CAM priority.

7.5.3.2 Indications of the number of CAM priority levels and to the time threshold values

The more CAM priority levels there are the closer the scheduling algorithm will be to an ideal EDF algorithm. In any case, adding new xi will be efficient only if it is done towards low values. A decreasing geometric is a good starting point. Note that there are always less thresholds with lower MSDU priority levels.

Additional principles in order to guide the xi determination: each xi can be considered as a defence line of the TB traffic against external or internal perturbations. However, it should not be forgotten that the system cannot behave better than pure EDF scheduling within the same MSDU priority.

Briefly, each threshold xi will defend the packets with current RTD lower than xi from the temporary bursts in the flow of packets with current RTD higher than xi. If the burst is sporadic enough, the proportion of packets crossing the threshold xi over those transmitted before this deadline should be significantly small. Of course this property will not hold if the burst has too long a duration.

For instance, x1 will protect the isochronous traffic (voice and video) from the possible bursts of asynchronous traffic holding default TD values. Note that this first threshold only operates on high MSDU priority traffic.

x2 protects medium TD time bounded traffic from higher TD time bounded traffic, for example: video, with higher throughput or perturbed by asynchronous bursts.

x3 protects low TD time bounded traffic with low throughput (voice) from medium and higher TD time bounded traffic, for example: video, with higher throughput or perturbed by asynchronous bursts.

The last one, x4, will protect all time bounded traffics against their own bursts or against any other external perturbation. Since only five CAM priorities are assumed, x4 plays the role of last line of defence of TB traffic. Therefore x3 should be lower than any expected initial TD.

Numerical determination of those xi can follow from these considerations and the parametrisations of the TB services listed in clause 5.

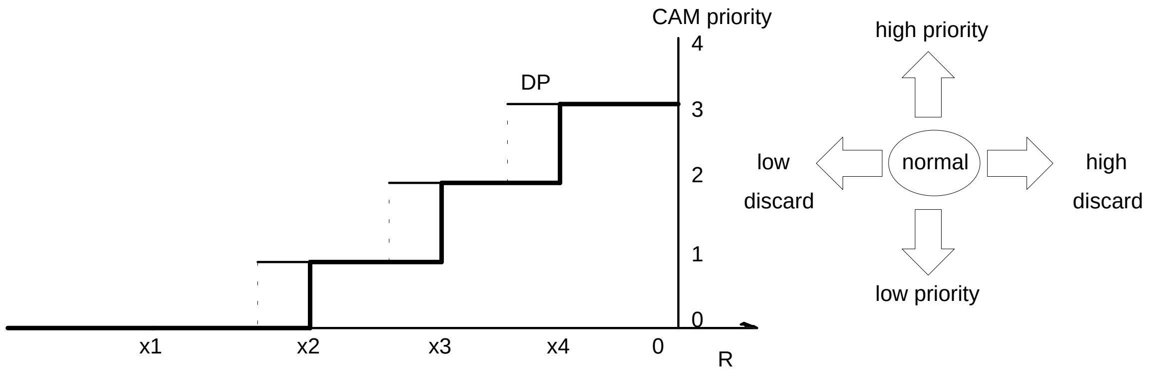
7.5.4 Alternative or refined CAM priority assignment

It has been seen that the effect of the MSDU priority must be on the terminal CAM priority of the packet. On the mapping function it is equivalent to a vertical translation of the plot.

It is mandatory that higher priority MSDUs receive a higher terminal priority than lower priority packets in order to keep the effect of the MSDU priority parameter at overload conditions. The following alternative mapping has been envisioned: CAM priority = min(1+F(R), F(0)+MSDU priority). This refinement makes it possible to moderate the impact of the MSDU Priority when the packet is still far from its deadline and keeps the packets indistinguishable before they reach their terminal CAM priority. The actual mapping accentuates the impact of the MSDU priority by introducing discrepancy between packets even when they are far from their respective deadlines.

The values x1 x2 x3 and x4 can be shrunk in order to take into account a larger number of hops or some other external condition. For example the last line of defence x3, could be brought down to 4 ms in order to cope with all possible TD per hop (TD divided by the expected number of hops).

If a discard parameter DP were expressed in terms of critical time, the impact would be on the computation of R, by subtracting the critical delay to R. In the mapping function, it is equivalent to a horizontal translation of the plot, see figure 8.



CAM priority mapping with a discard parameter in normal priority, the four degrees of freedom

Figure 8: Alternative CAM priority assignment

A mapping must be defined between Discard Parameter and Critical Delay. For example, a logarithmic scale with a normal discard statistic, Dnormal= **-** 30 dB and a 10 dB loss delay, y=4x/3 could be used. The formula CD=-(DP+Dnormal)*y should be used with DP expressed in dB, see figure 9.

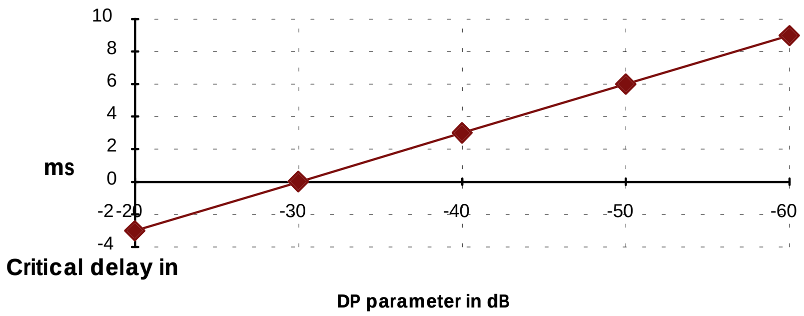


Figure 9: Translation of a DP parameter into a critical delay

7.6 Packet queuing and selection to CAM

7.6.1 Basic principles

The data transfer service must select and submit to CAM a packet candidate among those holding the highest CAM priority. The CAM is non pre-emptive except at the end or beginning of channel cycles. At the end of each (unsuccessful) CAM channel cycle, the candidate packet is returned so that a new candidate can be submitted for the next channel cycle.

7.6.2 Detailed packet selection

If the highest CAM priority of the packet in data transfer queue contains more than one packet, then the selection among those packets must be made according to the lowest RTD. If more than one packet holds the same lowest RTD, the selection is made according to the (local) MPDU Priority. If an ambiguity still remains, then the data transfer picks up any one of the remaining packets.

7.6.3 Refinement or alternative on packet selection

The event that two packets hold the same RTD with different MP is not expected to occur frequently. Thus, if one wants an efficient impact of MP parameter at this stage, then one might adopt an enlarged selection rule such that the candidates are all packets with an RTD within Delta of the lowest RTD. Delta must be fixed.

One could imagine to reverse the selection parameters by first using the MP parameter and, second, the RTD. But in this case one would greatly disadvantage the packets with low MP which are close to their deadline in favour of packets holding a high MP but which are still far from their deadline.

An easy simplification would be not to use the MP parameter at this stage, since it already impacts the access to the highest CAM priority.

For various reasons MAC may force some packets not to be transmitted before a certain time. An example of such a case is when power saving features are activated: MAC must wait for the end of the sleep period of the destination to transmit its packet. In this case a Not Transmit Before (NTB) parameter might be attached to each packet (default value 0). Consequently the selection of CAM candidates can only occur on packets with expired NTB. Warning: the NTB parameter is a MAC parameter and is not a user QoS parameter, because of the MAC Modus Operandi.

There is the possibility that a selected packet cannot succeed in any of its retransmissions because its destination is temporarily out of range (successive unsuccessful retries). In this case the data transfer service could select another candidate. The Retry and NTB parameters could be used to temporarily defer this same packet.

7.6.4 Packet queuing and feasibility

The packet or references to packet are stored in the MAC up to the limit of the local memory space. If a packet comes when there is buffer overflow its request is rejected and the confirm carries the parameter: rejected because of resource.

An incoming packet can also be rejected in real-time because its QoS parameters are not feasible. The element of the architecture devoted to this operation is provisionally called the filter. The filter takes as input the QoS of the incoming packet and the QoS parameters of the other already stored packets, it determines that the new QoS is not feasible according to simple and efficient rules. Due to the difficulties encountered in determining such rules the filter is considered as optional and will be discussed in the refinement clause.

7.6.5 Comment on packet queuing

Note the following implementation detail: the packets coming from the relaying function are stored in full in the MAC; packets directly pending from MAC user request can be stored as a simple pointer.

7.6.6 Alternatives and refinements

7.6.6.1 The buffer overflow

When a packet arrives in MAC when there is current buffer overflow it can nevertheless be accepted by taking the place of a packet holding, for example, a lower MP. The packet with the lower MP will be rejected and its confirm would possibly carry the parameter "rejected because of resource". This raises the issue of which packet would be selected for destruction: the one with the lower MP and/or with the highest RTD? A Discard parameter would be useful in this case. Several destructions might be needed in order to accept a long packet if the latter has to be stored in full.

7.6.6.2 The filter

The filter handles incoming packets and accepts or refuses them according to their feasibility. The feasibility rule must be such that it does not discard a packet when it still has a chance of being transmitted. It is a difficult problem to fulfil this requirement with a simple heuristic without restricting it to trivial pathological cases.

Two kinds of rules are considered: the sub-optimal heuristics and the super-optimal heuristics. The sub-optimal rules do not kill enough packets to make the filter efficient. The super-optimal rules kill too many packets and endanger the communication process.

Three possible rules are listed below with comments.

A trivial rule: to add the lengths of all the already stored packets whose RTD and MP make them eligible for transmission before the new incoming packet. If the result is larger than the TD of the new packet, then the latter is discarded. Note that this rule does not consider the random delay induced by the multiple access scheme, therefore the rule is sub-optimal. Nevertheless, the rule is not absolute because there is never certainty about the future of any already stored packet. Indeed there is always the possibility that a more urgent new packet will take the place of an old one and force the latter to be discarded. In this case the feasibility rules that forced some packets to be discarded in the past were probably not correct.

A queuing rule: to count the already stored packets whose RTD and MP make them eligible for transmission before the new incoming packet. If the result multiplied by a constant Delta1 is larger than the TD of the new packet, then the latter is discarded. Note that this rule generalises the previous one (the length of the packet can be added in the estimate). Delta1 plays the role of the mean access delay. If the estimate is too small, then the rule is sub-optimal; if the estimate is too large, the rule is super-optimal. The estimate of the average access delay is not easy because it is difficult to forecast the CAM priorities a packet will carry at the time of its transmission.

A congestion rule: to average the RTD of all the stored packets. If the result is smaller than a given constant Delta2, then a congestion is assumed and all the new packets are rejected. New packets will again be accepted when the ratio is above Delta2. Delta2 plays a similar role to Delta1 in the previous rule: if Delta1 is too small the rule is sub-optimal, otherwise it is super-optimal. This has the merit of simplicity but its all or nothing principle may lead to chaotic behaviour. There is an issue about determining the appropriate value of Delta2.

In case no filter is implemented, the buffer overflow control and the discard policy can do a similar job.

In buffer overload conditions the packet can only be accepted only when a previously stored packet has been transmitted. Consequently the packets are accepted in the queue per period exactly equal to the average channel access, simulating a filter based on queuing rule but with the ideal Delta evaluation.

In general it is expected that overload conditions lead to buffer overflow. However, if the buffer is too large then the discard policy will take control of the number of stored packets. Packets will be uniformly discarded between each channel access and MAC will transmit only packets with RTD or input times separated by an actual channel access delay.

7.7 Exception in data transfer: the internal service PDU

The forwarding management PDU are examples of internal MAC PDU whose delivery is critical for the network maintenance and the achievement of data transfer. Each time such a packet is generated to the MAC it will be allocated a default TD (500 ms) and systematically be selected as the CAM candidate with the highest CAM priority. Note that in existing systems, management packets are usually offered the highest CAM priority [12], [13]. The packet will be discarded when RTD reaches zero.

7.7.1 Alternatives and refinements

The critical MAC PDU could be treated as a regular packet with the more stringent QoS parameters. The disadvantage is that the most stringent QoS parameters imply shorter TD and a uniform probability of being jettisoned in local overload conditions.

The present rule protects the MAC PDU against local overload conditions but it does not protect the packet against global overload conditions. In global overload conditions the channel access delay might be too large because too many nodes attempt to transmit packets at the highest CAM priority. A remedy is to set critical MAC PDU to a specific highest CAM priority distinct from the other CAM priorities offered for usual data transfer services. If this is done, critical MAC PDU will only contend between themselves. On the other hand, the internal MAC PDU traffic must be tuned in such a way that it does not swamp all the channel capacity.

8 LLC guidelines for the utilisation of Time Bounded Services

To manage TBS the MAC user has a set of QoS parameters: Desired Transfer Delay (DTD), the MSDU Priority (MP). An additional Discard Parameter (DP) is not included in the current HIPERLAN specifications but will be discussed here also.

8.1 Use of the DTD parameter by the time bounded MAC user

The DTD parameter is the most important parameter. DTD is upper bounded by the upper limit of validity of the MSDU, i.e. the largest delay that the packet can experience without an adverse effect on the time bounded service. For voice and video, this quantity ranges between 20 ms to 100 ms. For voice, it is the sampling period; for video, it is the picture refreshing period taking into account that one picture takes several packets.

The DTD is only the maximum delivery delay between MAC users. Therefore, it does not include the extra processing delay that the packet will experience above MAC (formatting, re-sequencing, buffering, etc.). Therefore, DTD is at most the limit of validity of the packet minus the processing delays.

Although TBS such as voice and video require accurate timing in packet processing (to reconstitute the sample of voice or moving video), it would be useful to define lower limits of packet validity. However, the DTD will not be directly impacted by such a parameter because MAC delivers packets as soon as possible regardless of eventual lower limits of packet validity. The timing in packet delivery and processing will be provided above MAC, for example via buffer and re-sequencing. In this perspective DTD can also be limited by the size of the re-sequencing buffers. The size of the buffer is DTD plus extra processing delay per packet divided by the time bounded period, see figure 10.

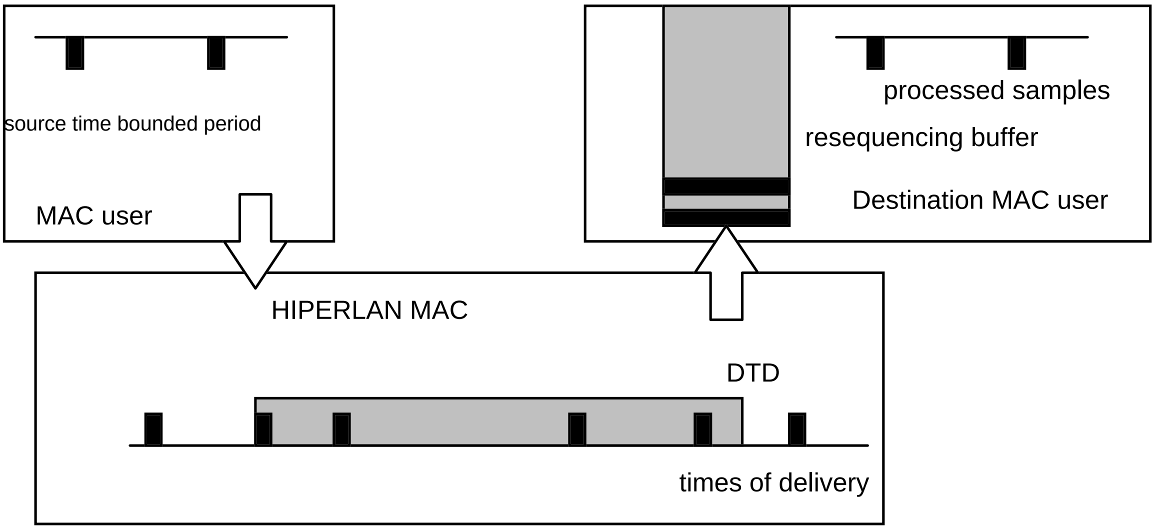


Figure 10: Use of DTD by the time-bounded MAC user

8.2 Use of the MP parameter by the time bounded MAC user

The MSDU priority parameter is more a system oriented parameter than a user oriented parameter. The system manager tells which TBS has more priority than another one, and this feature has nothing to do with the specific requirements of the service.

Therefore there are few cases where the direct MAC user (the LLC) will tune this parameter. One case has already been mentioned for MPEG video compression. There are other examples: the call packet to establish a telephone connection may have higher priority than the sampling packets, see figure 11.

In fact these cases show that the MAC user will use the priority parameter like a discard parameter. Indeed the alternative to using a discard parameter is interesting because the number of MSDU priorities is limited by the number of CAM priorities.

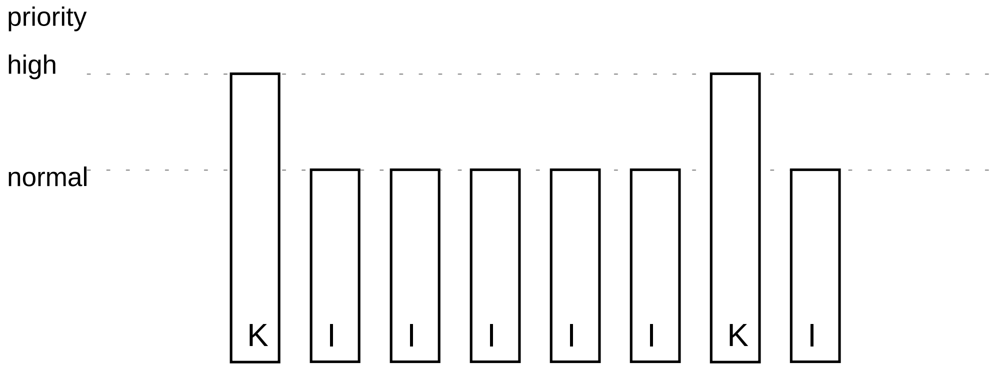


Figure 11: MPEG transmission procedure with K pictures and I pictures using priorities

8.3 Use of the DP parameter by the time bounded MAC user

The DP parameter should be expressed in dB, without the need for real accuracy. The example given above with an MP parameter can be translated with the use of a DP parameter, see figure 12.

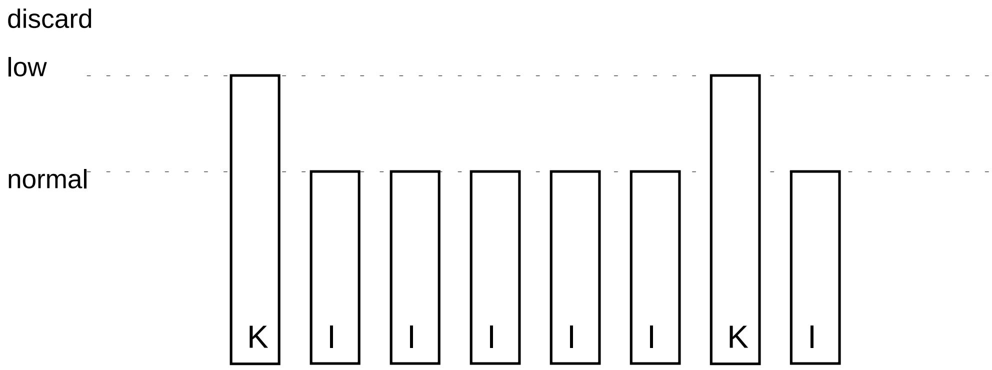


Figure 12: MPEG transmission procedure with K pictures and I pictures using discard parameters

8.4 Use of the DAM parameter by the time bounded MAC user

The Don't Acknowledge Me parameter can be used by the MAC user in order to limit the overhead attached to each packet. This alternative would be useful if a high bandwidth with medium reliability is required. This parameter would also be useful for enhanced LLC which tune the size of their window frame according to the actual performance of the MAC below.

9 Conclusion

This ETR presents how TBS together with the other HIPERLAN transfer services can be managed in a unified HIPERLAN data transfer service. This unification uses the concept of QoS parameters. The service requirements over HIPERLAN MAC are expressed in terms of these parameters. Guidelines to QoS parameter settings are provided in clause 8.

The HIPERLAN architecture for data transfer is based on two fundamental tools: the EDF algorithm and the hierarchically independent Channel Access (CAM) priority levels. See annex A, clause A.2 for an analysis of the effect of these two schemes.

Results from analytical models or from simulations show that the architecture described for the HIPERLAN data transfer service is able to support different and simultaneous services. With such an architecture, services like TBS are possible in a distributed and multi-hop context. See annex A, clause A.1 for simulation results and analytical models to assess this.

However the proper operation of the data transfer requires that all the connected nodes use consistently the QoS parameters. Therefore the standard must provide tests to verify if a node does properly use the QoS parameters to translate them into CAM priority.

Annex A: Analytical models and simulation results

This annex presents analytical models and simulation results related to the data transfer architecture of HIPERLAN.

Clause A.1 presents a simulation model to validate the architecture of HIPERLAN to handle TBS. The idea is to provide evidence of the correct support of time bounded services by testing real scenarios. Most of the simulation results given concern the precise design of HIPERLAN as described in DTR/SMG-050201 (see Bibliography, annex B). However results are also presented concerning different architectures which mostly are derived from the HIPERLAN architecture by changing given parameters. Refinements of the HIPERLAN architecture are also simulated. It is clearly indicated when the parameters or the architecture differ from the HIPERLAN choices as defined in DTR/SMG-050201.

Clause A.2 presents other results obtained by analytical models concerning the architecture of HIPERLAN. The idea is to capture the effect of the hierarchically independent CAM priority levels and the effect of the EDF algorithm. It is shown that these two tools greatly help to provide TB services.

A.1 A simulation model for the HIPERLAN architecture

A.1.1 Main goals of a simulation model

Since the envisioned design for HIPERLAN includes the use of both hierarchically independent priority levels and an EDF-like algorithm, there is no existing analytical model to predict the exact behaviour of such an algorithm. Although the Cobham formulas provide mean access delay for an access scheme with various priority levels, this approach can capture neither the EDF effect nor the distribution of the delays. Very recent papers address the performance behaviour of the EDF with priorities but the results still only concern mean access delays.

This annex develops a model which allows us to test the TBS architectural choices. Briefly a queuing approach will be used to model the life of packets in the network. The transmission of a packet will be considered as a service of a distributed server and will remove packets from the station queues. An event driven simulation is used with a kernel which will consist mainly of a scheduler and waiting queues. This event driven simulation drives a global time and is therefore a proper tool to implement EDF-like algorithms. The input to a given simulation will consist of input traffic (given loads and time constraints), the output of a simulation will mainly be the ratio of packets not receiving their required services.

The figures used in the simulations below are taken from DTR/SMG-050201 and the simulations will therefore provide real performance of the HIPERLAN architecture. However, slight variations in the architecture described in DTR/SMG-050201 are tested with simulation. Unless otherwise indicated the simulation results refer to figures and precise architecture described in DTR/SMG-050201.

A.1.2 The simulation model

A.1.2.1 Model of nodes

A node will be represented by a waiting queue. The number of buffers available in a node is a parameter of the simulation model. A request arriving at a node with full buffers is rejected. The ratio of rejected requests will be an output of the simulations (see figure A.1). This queue will hold the QoS of generated packets in this node. In these waiting queues, the updates due to arrivals or departures of packets and the updates of QoS with the time are performed.

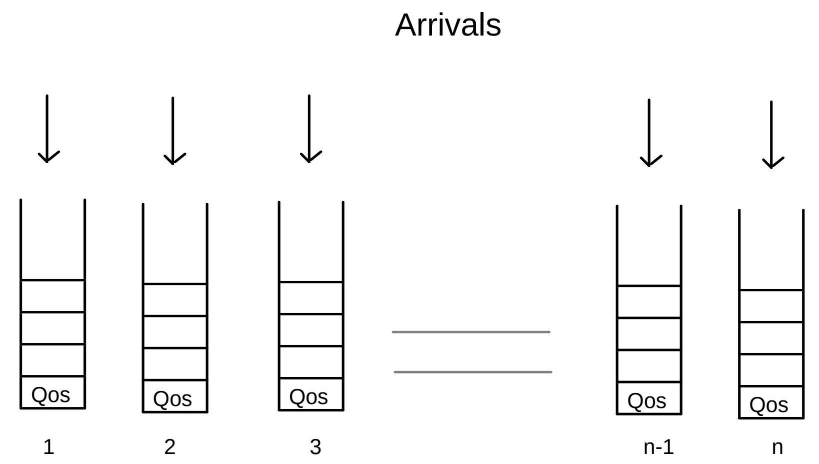


Figure A.1: Models of packets arrival and storing on different nodes

The algorithm which handles the data transfer service computes the CAM priority for every queued packet (see figure A.2). This computation is mandatory, since the candidate packet for the CAM must be the packet holding the highest priority. Other criteria to select a CAM candidate are likely to be used.

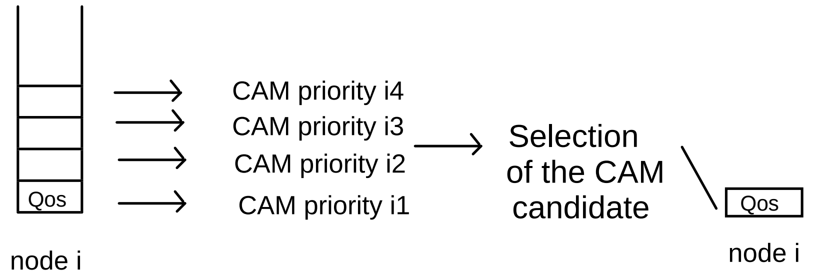


Figure A.2: Selection of CAM candidate in node i

A.1.2.2 Transmission of a packet

A.1.2.2.1 Generality

It is assumed that the CAM operates like a random selection among packets with the highest priority (see figure A.3). The time is advanced from a duration corresponding to the duration of the packet. At the real duration of the packet, an overhead may be added representing the collision and the acknowledgement.

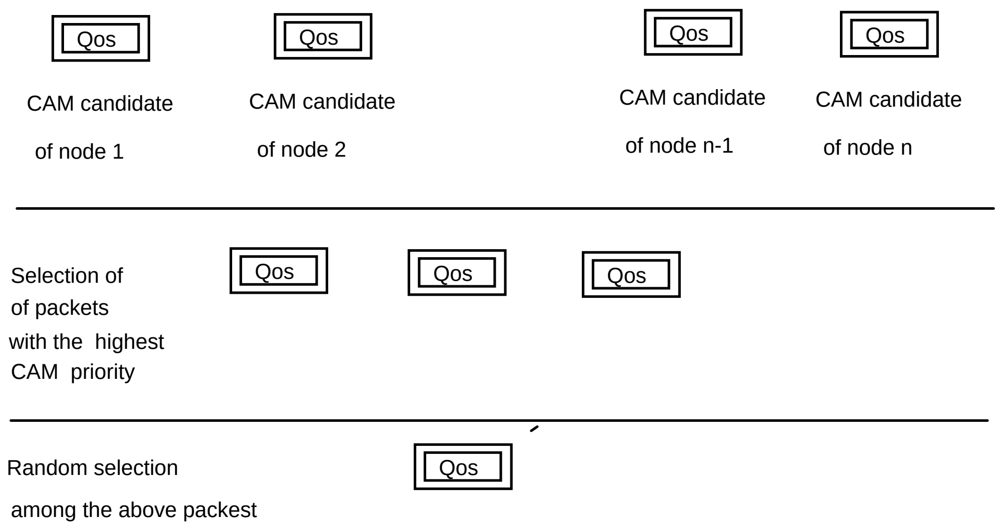


Figure A.3: Selection of global CAM candidates over waiting nodes

A.1.2.2.2 Overheads

A.1.2.2.2.1 Physical overheads

Four different overheads linked to the transmission of the packet may be distinguished. These overheads are now detailed and presented as they appear on the communication channel during the transmission of a packet:

a) Inter frame separation: 256 bits;

b) priority overhead: $cam\_priority \times p_{ES} + P_{PA}$;

c) elimination overhead, mean value: $(\frac{\log(n)}{\log(2)} + \frac{g}{\log(2)} - 0,5)i_{ES} + i_{EVS}$, where n denotes the number of contending nodes. See RES 10/94 (see Bibliography) and related references for the formula;

d) selection overhead, mean value: $5p_Y$;

e) Low Bit rate Header (LBH) overhead: 35 low rate bits which correspond to a duration of 560 bits (high bit rate);

f) training sequence overhead: 450 bits;

g) gap between packet and acknowledgement: 512 bits;

h) acknowledgement overhead: 23 low rate bits which correspond to a duration of 368 bits.

For unicast transmission the constant overheads are: inter-frame separation, LBH, time separation between the packet and its acknowledgement, duration of the acknowledgement. The sum of all these overhead corresponds to 2 146 bits.

For multicast transmission the constant overheads are: inter frame separation, LBH. The sum of all these overhead corresponds to 1 266 bits.

In the following this overhead is called the constant transmission overhead.

The overhead consisting of the prioritisation phase, the elimination phase and the yield phase is called in the following, the variable transmission overhead.

A.1.2.2.2.2 MAC overheads

Two types of MAC overheads may be distinguished. The first type of MAC overhead consists of all fields in an HMSPDU. The second type of MAC overhead consists of the expansion due to the coding.

The 31/26 expansion: the data blocks are 416 bits which result after the coding in a block of 496 bits.

Table A.1 lists all the overheads in the case of data transfer PDU when no alias address is used.

Table A.1

Block length indicator field	1 octet
Padding length indicator field	1 octet
Identifier field	4 octets
Destination HPhSAP address field	6 octets
Source HPhSAP address field	6 octets
HMPDU length indicator field	2 octets
HMPDU type indicator field	1 octet
HMPDU residual MPDU life time	2 octets
HMPDU source address	6 or 12 octets
HMPDU source address	6 or 12 octets
HMPDU sequence number field	2 octets
HMPDU user priority and lifetime	2 octets
HMPDU key identifier and initialisation vector	1 or 4 octets (depending on whether or not the PDU is encrypted)
HMPDU sanity check	0 or 4 octets (depending on whether or not the PDU is encrypted)
Checksum field	4 octets

In the following simulations, an unencrypted HMPDU with source and destination address without alias is considered. In this case the MAC overhead is 43 octets = 344 bits.

Let us consider the MAC overhead for packets of 260 bits, 320 bits and 10 000 bits. For the packet of 260 bits or 320 bits the 344 bits overhead in MAC fields implies the use of two blocks and therefore it results in $2 \times 496 = 992$ bits on the air. For the 10 000 bit packets 25 blocks of 416 bits are needed which produce 12 800 bits on the air.

A.1.2.2.3 Collision

According to the design of the Elimination-Yield Non-pre-emptive Priority Multiple Access (EY-NPMA) there is a maximum collision rate of 3,5 % with nodes within range. To take into account this phenomenon, a collision is emulated in the simulation software and the percentage of collision is set to its maximum value of 3,5 %. This collision lasts for the transmission duration of a packet and the packet is in this case not removed from the node. The packet will be sent again.

A.1.2.2.4 Collision with hidden nodes

Although the model does not take into account a precise model for physical transmission the software does, however, allow hidden nodes to be taken into account via a parameter which gives the percentage of collisions caused by hidden nodes. This parameter denotes, for a given transmission, the probability that this transmission will collide with one from a hidden node.

The assumption in the case of collision is the following: this packet is not allowed to be submitted again before a time interval corresponding to the transmission duration of a packet of maximum length: 19 522 bits. Note that this is restriction is not called for by the standard.

Notice that without special mention, the results presented in this ETR concern transmissions without 5 % of hidden nodes.

A.1.2.2.5 Transmission in multi-hop

Since the model does not into account a precise model for transmission, it is impossible to use it in spatial reuse. However the following simplification to validate time bounded multi-hop is used. A generated packet receives the number of hops to reach its destination: x and this packet will be transmitted x times before quitting the system. This simplification is a worst case for multi-hop transmission since it introduces strong correlation between transmission times in the different cells.

Notice that without special mention the results presented in this document concern single hop transmissions.

A.1.2.3 Arrivals of packets

Two types of arrivals, synchronous traffic and asynchronous traffic, will be considered. For all types of traffic, the length of packets is a main parameter.

For synchronous traffic, the other main parameter will be the period of packet arrivals. Moreover corresponding to usual applications, voice and video, one can easily find out QoS parameters. Another simulation parameter will be the number of synchronous traffics. Notice that any synchronous traffic is attached to a given node.

For asynchronous traffic, the main parameters will be the inter arrival of the packets and the number of arriving packets in a batch. Generally this traffic corresponds to file transfers. There is a special case of asynchronous traffic where batches contain only one packet. This traffic consists of alarm packets. These packets can be considered asynchronous but unlike the previous traffic there is a very clear urgency for the delivery of such packets. For this type of traffic, the main parameter is the inter arrival of packets.

A.1.3 Organisation of the software

A.1.3.1 An event driven simulator

The simulations mainly rely on a global scheduler. This scheduler mainly manages two types of events: arrivals of new packets and departure of packets after transmission. The software increments time between event processing and stores the forthcoming events. That leads to the updates of QoS parameters of packets and to the updates of the mapping of CAM priorities. The software is organised around a kernel which handles events and provides the basic management operation in a waiting queue. Another C program holds the architectural design concerning the TBS. Therefore it is possible to compare the behaviour of many different architectures and their performance.

A.1.3.2 Types of events

The events can be of two types:

a) arrival of a new packet in a queue. This arrival may occur either during a transmission or when the channel is idle. It may be noticed that in the case of an arrival during a transmission the priority update is not mandatory if pre-emption is prevented during a transmission;

b) departure of a packet occurs at the end of the transmission. Notice that the end of a successful packet transmission does necessarily lead to the removal of the packet from its host node. In the case of collision the packet remains in its host node. Otherwise, it is stored in the queue of the destination node if a next hop is scheduled. The packet is definitely discarded from the simulation if its last hop is achieved.

A.1.4 Results of simulations

A.1.4.1 Configuration of the simulation

The following parameters are used consistently in all the following simulations:

- 23 529 400 bits/s radio channel;
- transmission overhead = 2 146 bits for unicast transmission and 1 266 bits for multicast transmission;
- $\mathbf{i_E}$ = 256 bits, $\mathbf{i_{EVS}}$ = 256 bits;
- $\mathbf{i_{PA}}$ = 256 bits, $\mathbf{i_{PS}}$= 256 bits;
- $\mathbf{i_{YS}}$ = 64 bits.

The configuration of the simulation is the following:

- physical parameters:
  - simulation duration = 2s;
  - fully connected stations;
  - number of stations: 30 nodes;
- traffic types:
  - synchronous traffic;
    - 32 kbit/s links composed of 320 bit packets (useful data, overhead excluded). Thus within a link a packet is generated every 10 ms. The number of such links is a parameter of the simulations;
    - 13 kbit/s links composed of 260 bit packets. Thus within a link a packet is generated every 20 ms. The number of such links is a parameter of the simulations;
    - 2 Mbit/s links composed of 10 000 bit packets. The number of such links is a parameter of the simulations.

The links are randomly chosen among the nodes at the beginning of the simulation;

  - asynchronous traffic: file transfer;
    - 10 000 bit packet. The load of this traffic is an input parameter of the simulation;

NOTE: On the air the above mentioned packets of 320 bits, 260 bits and 10 000 bits result in packets of respective length: 982 bits, 982 bits and 12 400 bits.

The source or host node of the packet are randomly chosen among the connected stations.

- QoS parameters:
  - the QoS parameters used by the synchronous traffic are the following: for the 32 kbit/s link DTD = 20 ms, for the 13 kbit/s link DTD = 20 ms, for the 2 Mbit/s link DTD = 50 ms. The MP is normal for all the links;
  - the QoS parameters used by the asynchronous traffic are DTD = 500 ms, MP normal.
- threshold for the CAM priority mapping:
  - x4=10 ms, x3=20 ms, x2=40 ms and x1=80 ms.

A.1.4.2 Results of the simulation

A.1.4.2.1 Various synchronous links

The aim of the above simulation results is to show that HIPERLAN can support a significant quantity of simultaneous synchronous traffic mixed with asynchronous transfer. Results with various numbers of synchronous links are presented. The results also emphasise the use of the CAM priority levels. In HIPERLAN there are actually 4 priority levels plus one which is reserved for high priority packets. Results with fewer priority levels are presented to compare with the HIPERLAN design. The percentage of hidden collisions is 5 %.

The following scenarios were tested:

- 10 links at 32 kbit/s, 10 links at 13 kbit/s, 1 link at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s (Poisson traffic, mean inter arrival 60 000 bits). The total load of the channel is therefore 13,4 Mbit/s including all of the overheads for 6,380 Mbit/s of user data load;

- 15 links at 32 kbit/s, 15 links at 13 kbit/s, 1 link at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore 15,8 Mbit/s including all of the overheads for 6,605 Mbit/s of user data load;

- 20 links at 32 kbit/s, 20 links at 13 kbit/s, 1 link at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 18,1 Mbit/s including all of the overheads for 6,830 Mbit/s of user data load;

- 25 links at 32 kbit/s, 25 links at 13 kbit/s, 1 link at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 20,5 Mbit/s including all of the overheads for 7,055 Mbit/s of user data load;

- 30 links at 32 kbit/s, 30 links at 13 kbit/s, 1 link at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 22,8 Mbit/s including all of the overheads for 7,280 Mbit/s of user data load.

All of the traffic in this simulation uses a normal priority. The ratio of discarded packets with the HIPERLAN architecture using 4 priority levels is compared with using less priority levels (2 and 1). In this case, the same function to map the priority is used but we only keeps the highest priority levels. Depending on the actual number of CAM priority we do not consider the thresholds: x4, x3, x2.

In the above scenarios, the total channel load and user data load for the specific traffic mix are indicated. Figures A.4 to A.7 below, give the ratio of discarded packets for the various traffics, as a function of the total generated channel load by traffic mix. The importance of calculating the total channel load generated for each traffic mix is to show its relationship to the total channel capacity of 23,529 4 Mbit/s.

It can be seen that the system can successfully support up to 20 links at 32 kbit/s, 20 links at 13 kbit/s, 1 link at 2 Mbit/s with a simultaneous file transfer of an average 3,93 Mbit/s. It appears that, with four priority levels, the system seems to operate correctly in the three first situations whereas the two last situations lead to problems. The effect of the number of the CAM priority levels is very clear. Without CAM priority levels, the system does provide the requested quality of service. The difference between the use of two priority levels and four priority levels is not significant in these scenarios.

Figure A.4: Discard probability for the 32 kbps traffics

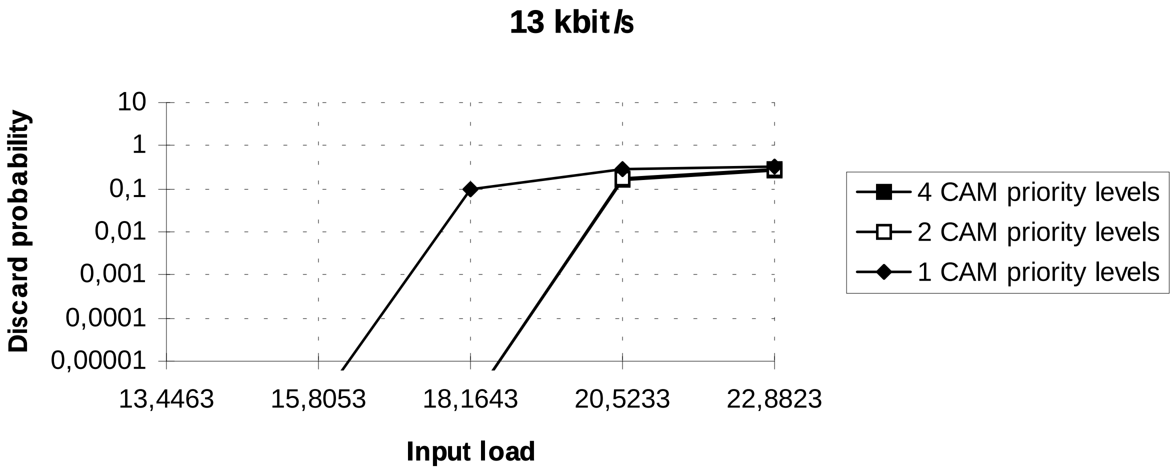


Figure A.5: Discard probability for the 13 kbps traffics

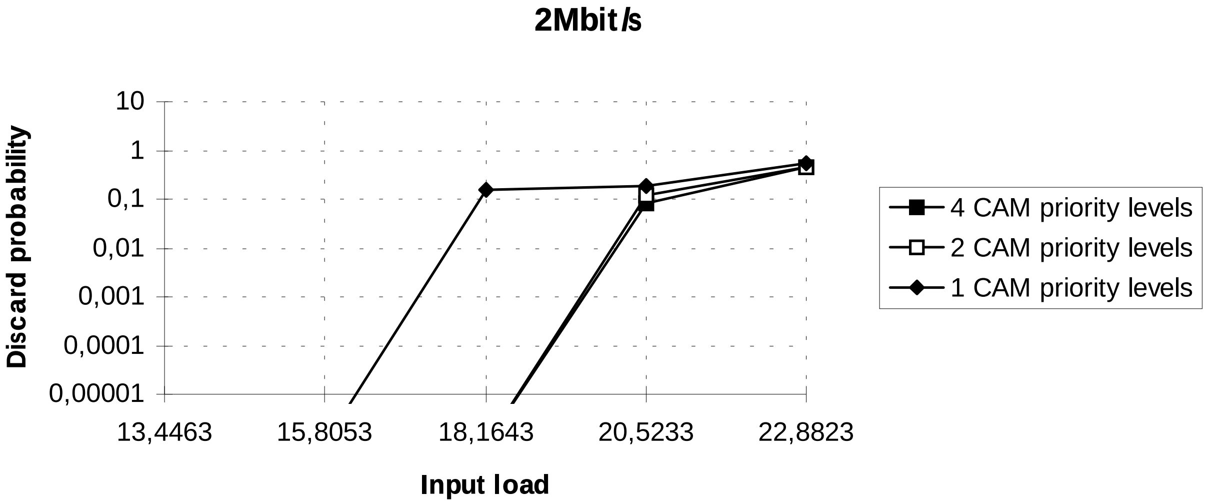


Figure A.6: Discard probability for the 2 Mbps traffics

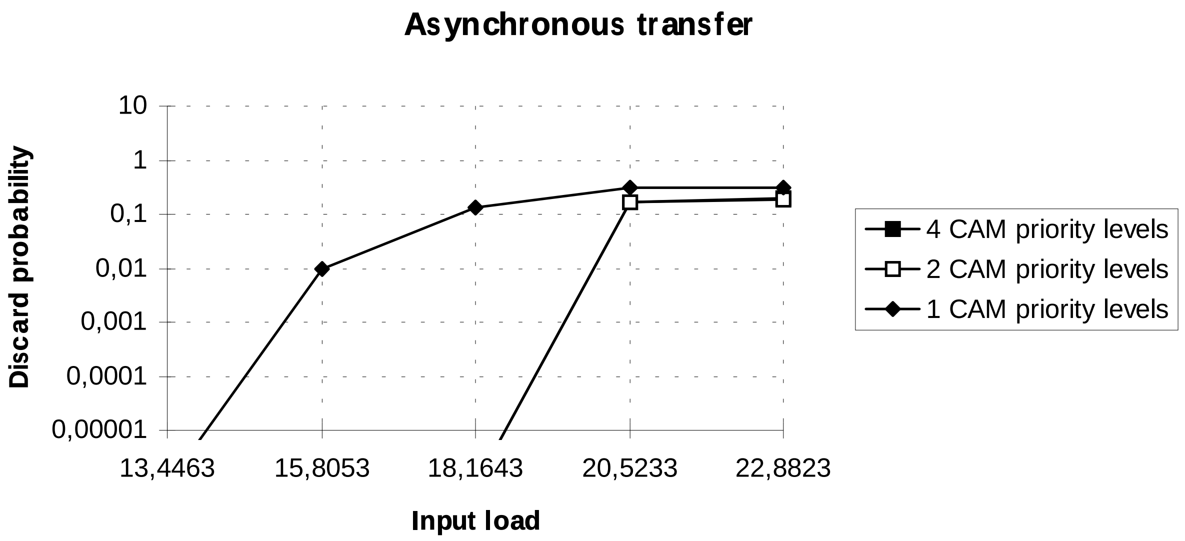


Figure A.7: Discard probability for the asynchronous (file transfers) traffic

The following scenarios were tested:

- 10 links at 32 kbit/s, 10 links at 13 kbit/s, 1 link at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 13,44 Mbit/s including all of the overheads for 6,32 Mbit/s of user data load;

- 10 links at 32 kbit/s, 10 links at 13 kbit/s, 2 links at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel (including all the overheads) is therefore: 16,27 Mbit/s including all of the overheads for 8,32 Mbit/s of user data load;

- 10 links at 32 kbit/s, 10 links at 13 kbit/s, 3 links at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 19,10 Mbit/s including all of the overheads for 10,32 Mbit/s of user data load;

- 10 links at 32 kbit/s, 10 links at 13 kbit/s, 4 links at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 22,17 Mbit/s including all of the overheads for 12,32 Mbit/s of user data load;

- 10 links at 32 kbit/s, 10 links at 13 kbit/s, 5 links at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 25 Mbit/s including all of the overheads for 14,32 Mbit/s of user data load.

All of the traffic in this simulation uses normal priority. We compare the ratio of discarded packets with the HIPERLAN architecture 4 priority levels and with fewer priority levels: 2 and 1. In this case, the same function to map the priority is used but we only keep the highest priority levels. Depending on the actual number of CAM priority the thresholds: x4, x3, x2 are not considered.

In the above scenarios, the total channel load and user data load for the specific traffic mix are indicated. Figures A.8 to A.11 below give the ratio of discarded packets for the various traffics as a function of the total generated channel load by traffic mix. The importance of calculating the total channel load generated for each traffic mix is to show its relationship to the total channel capacity of 23,5294 Mbit/s.

It can be seen that the system can successfully support up to 10 links at 32 kbit/s, 10 links at 13 kbit/s, 3 links at 2 Mbit/s with a simultaneous file transfer of an average 3,93 Mbit/s. It appears that, with four priority levels, the system seems to operate correctly in the three first situations whereas the last situation leads to problems. The effect of the number of the CAM priority levels is very clear. Without CAM priority levels, the system does not provide the requested quality of service.

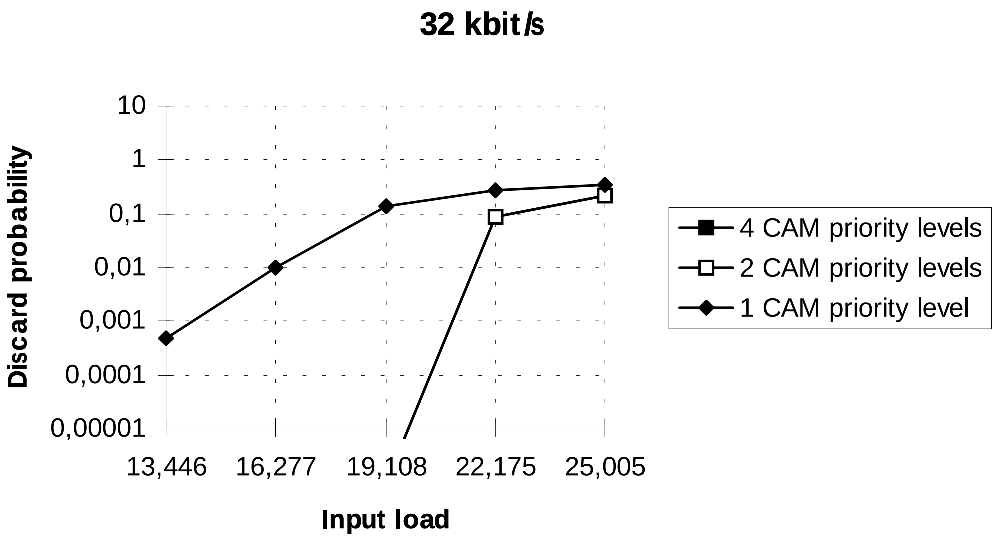


Figure A.8: Discard probability for the 32 kbps traffics

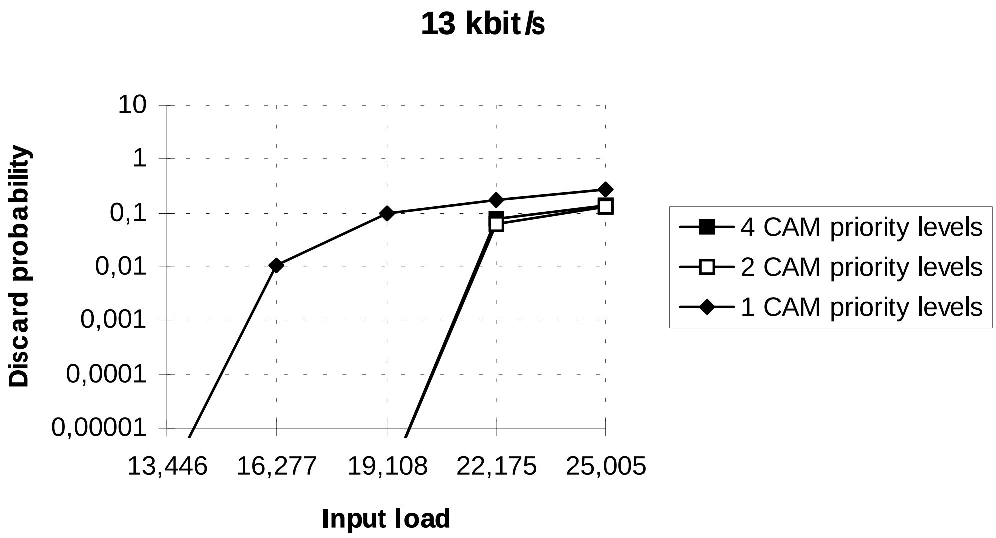


Figure A.9: Discard probability for the 13 kbps traffics

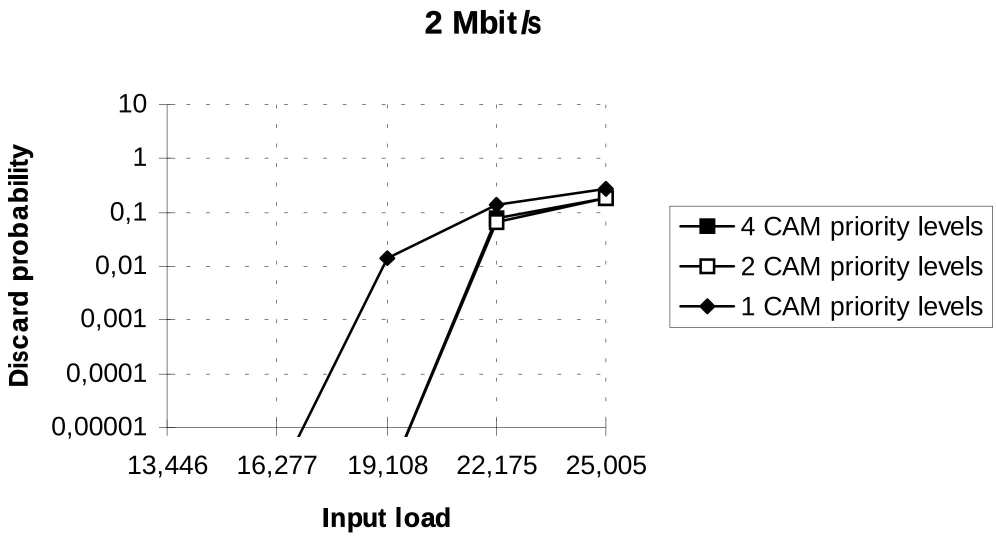


Figure A.10: Discard probability for the 2 Mbps traffics

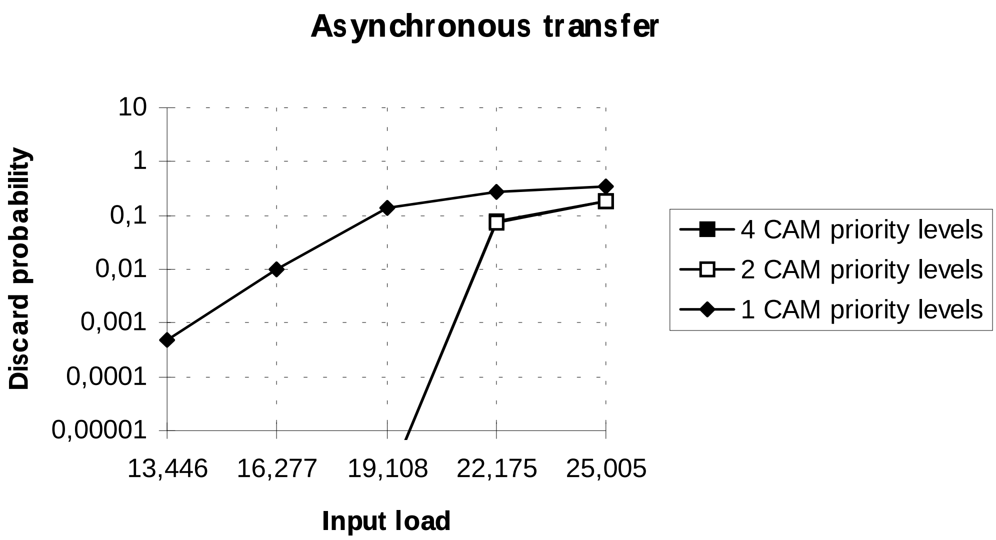


Figure A.11: Discard probability for the asynchronous (file transfer) traffic

A.1.4.2.2 Effect of hidden node collisions

The effect of collisions caused by hidden nodes has been simulated and the results are presented below. The simulation was based on the assumption that each packet would experience a collision probability of 20 %.

The following scenarios were tested:

- 10 links at 32 kbit/s, 10 links at 13 kbit/s, 1 link at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 13,44 Mbit/s including all of the overheads for 6,32 Mbit/s of user data load;

- 10 links at 32 kbit/s, 10 links at 13 kbit/s, 2 links at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 16,27 Mbit/s including all of the overheads for 8,32 Mbit/s of user data load;

- 10 links at 32 kbit/s, 10 links at 13 kbit/s, 3 links at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 19,10 Mbit/s including all of the overheads for 10,32 Mbit/s of user data load;

- 10 links at 32 kbit/s, 10 links at 13 kbit/s, 4 links at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 22,17 Mbit/s including all of the overheads for 12,32 Mbit/s of user data load;

- 10 links at 32 kbit/s, 10 links at 13 kbit/s, 5 links at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 25 Mbit/s including all of the overheads for 14,32 Mbit/s of user data load.

All traffic in this simulation uses normal priority. We compare the ratio of discarded packets with the HIPERLAN architecture 4 priority levels and with fewer priority levels: 2 and 1. In this case, we use the same function to map the priority but we only keep the highest priority levels. Depending on the actual number of CAM priority the thresholds: x4, x3, x2 are not considered.

In the above scenarios, the total channel load and user data load are indicated. Figures A.12 to A.15 below give the ratio of discarded packets for the various traffics as a function of the total generated channel load by traffic mix. The simulation results show that the system can still provide a very small ratio of discarded packets but the degradation with an increasing load is quicker. It can be seen that the system can successfully support up to 15 links at 32 kbit/s, 15 links at 13 kbit/s, 1 link at 2 Mbit/s with a simultaneous file transfer of an average 3,93 Mbit/s. The effect of the number of the CAM priority levels is very clear. The difference between 2 and 4 priority levels is obvious. That difference was not very clear with only 5 % of hidden nodes (see subclause A.1.4.2.1). This result is confirmed by the simulation results of subclause A.2.1.

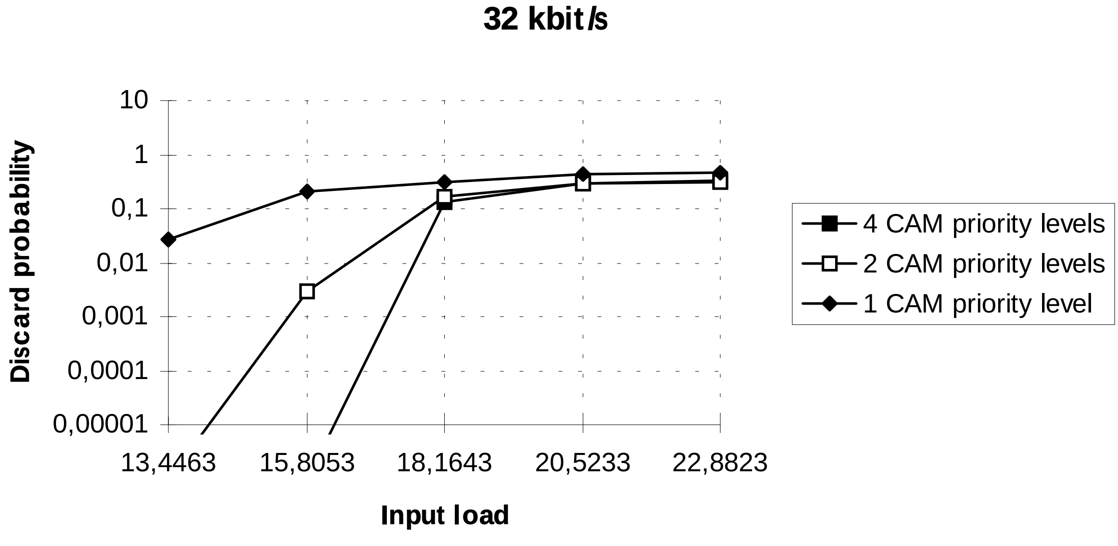


Figure A.12: Discard probability for the 32 kbps traffics with hidden nodes

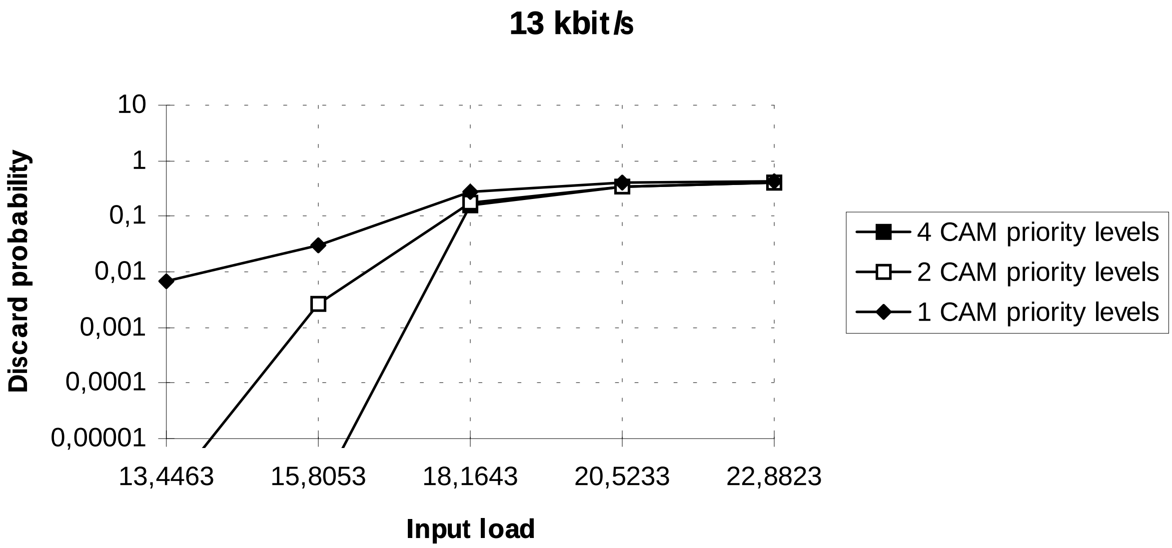


Figure A.13: Discard probability for the 13 kbps traffics with hidden nodes

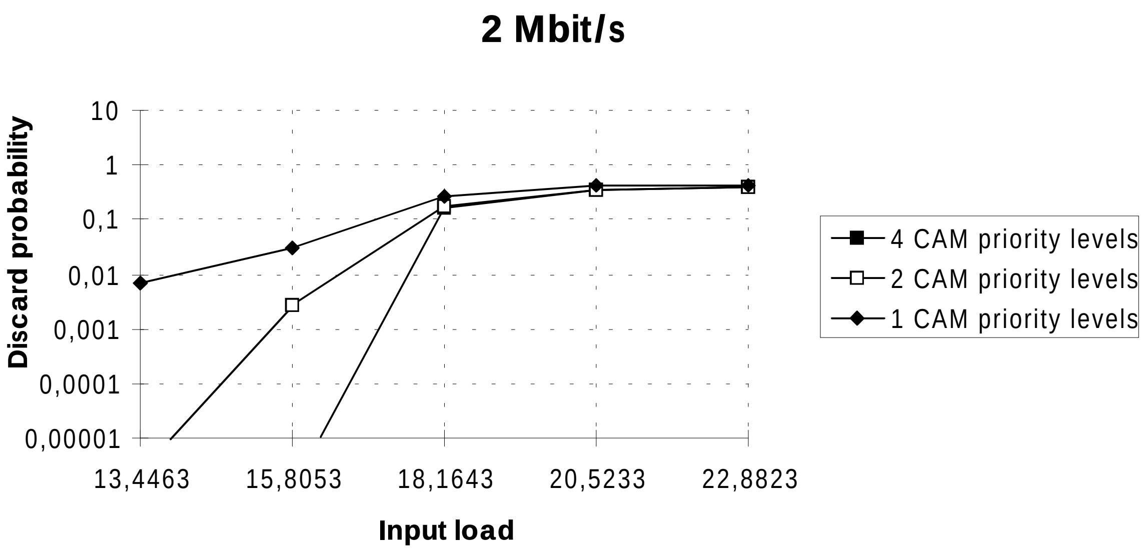


Figure A.14: Discard probability for the 2 Mbps traffics with hidden nodes

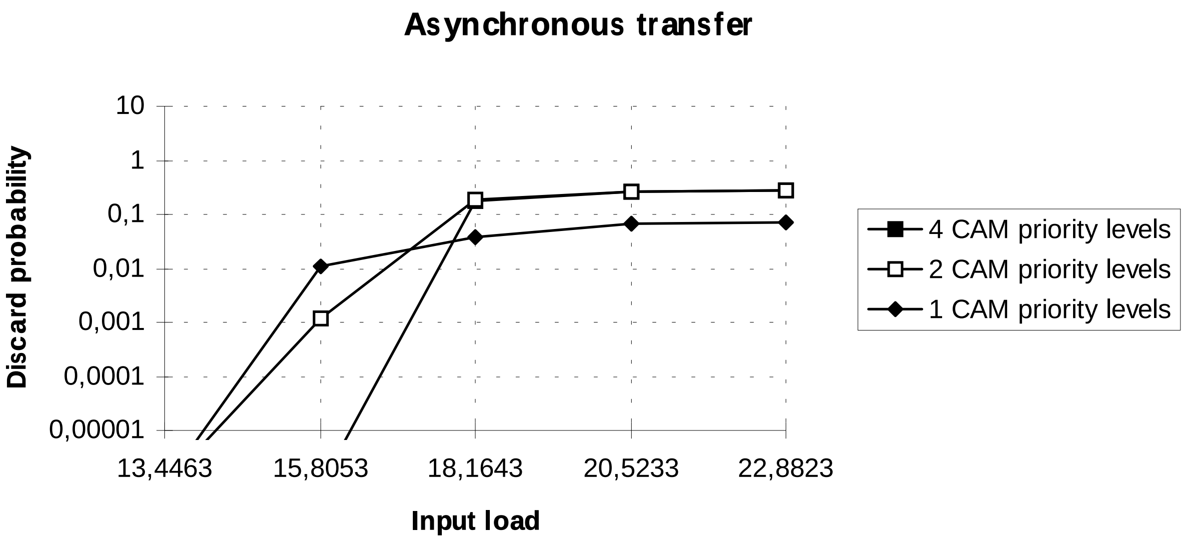


Figure A.15: Discard probability for the 2 Mbps traffics with hidden nodes

A.1.4.2.3 Multi-hop communication

The following investigates the effect of different numbers levels of priority in multiple hop transmissions, here transmissions over two hops have been used as examples.

In the figures below the four different situations presented are denoted by the corresponding total load.

The following scenarios were tested:

- 5 links at 32 kbit/s with two hops for a packet to reach its destination, 5 links at 13 kbit/s with two hops for a packet to reach its destination, 1 link at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 11,08 Mbit/s including all of the overheads for 6,155 Mbit/s of user data load;

- 10 links at 32 kbit/s with two hops for a packet to reach its destination, 10 links at 13 kbit/s with two hops for a packet to reach its destination, 1 link at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 13,44 Mbit/s including all of the overheads for 6,38 Mbit/s of user data load;

- 15 links at 32 kbit/s with two hops for a packet to reach its destination, 15 links at 13 kbit/s with two hops for a packet to reach its destination, 1 link at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 15,8 Mbit/s including all of the overheads for 6,605 Mbit/s of user data load;

- 20 links at 32 kbit/s with two hops for a packet to reach its destination, 20 links at 13 kbit/s with two hops for a packet to reach its destination, 1 link at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 18,16 Mbit/s including all of the overheads for 6,83 Mbit/s of user data load.

All traffic in this simulation uses normal priority. We compare the ratio of discarded packets with the HIPERLAN architecture 4 priority levels and with less priority level: 2 and 1. In this case, we use the same function to map the priority but we only keep the highest priority levels. Depending on the actual number of CAM priority the thresholds: x4, x3, x2 are not considered.

Figures A.16 to A.19, below, give the ratio of discarded packets for the various types of traffic.

It can be seen that the system can successfully support up to 10 links at 32 kbit/s, 10 links at 13 kbit/s, 1 link at 2 Mbit/s with a simultaneous file transfer of an average 3,93 Mbit/s. It appears that, with four priority levels, the system seems to operate correctly in the three first situations whereas the last situation leads to problems. The effect of the number of the CAM priority levels is very clear. Without CAM priority levels, the system does provide the requested quality of service.

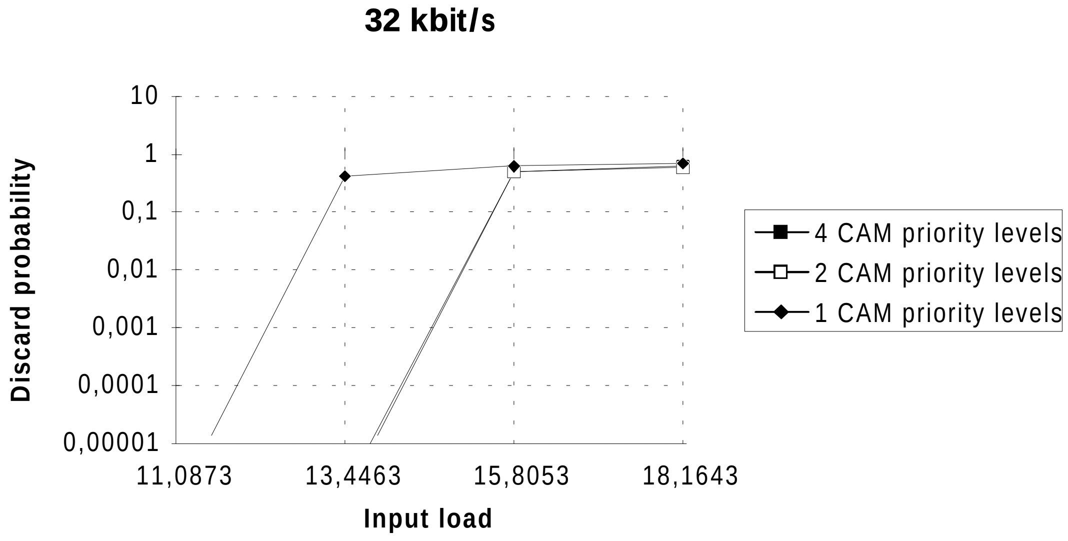


Figure A.16: Discard probability for the 32 kbps traffics with multi-hop

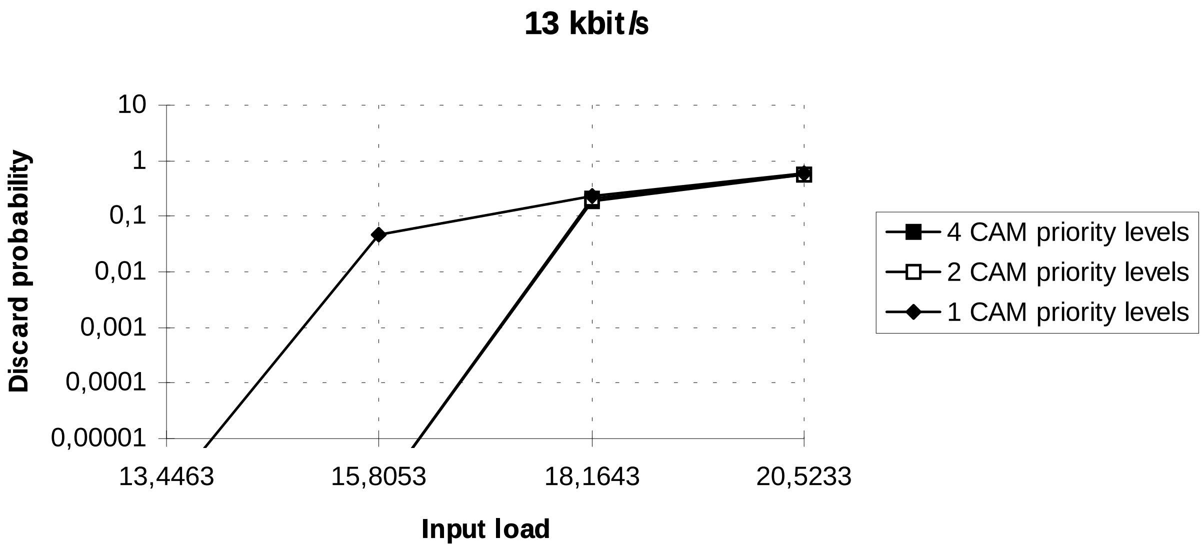


Figure A.17: Discard probability for the 13 kbps traffics with multi-hop

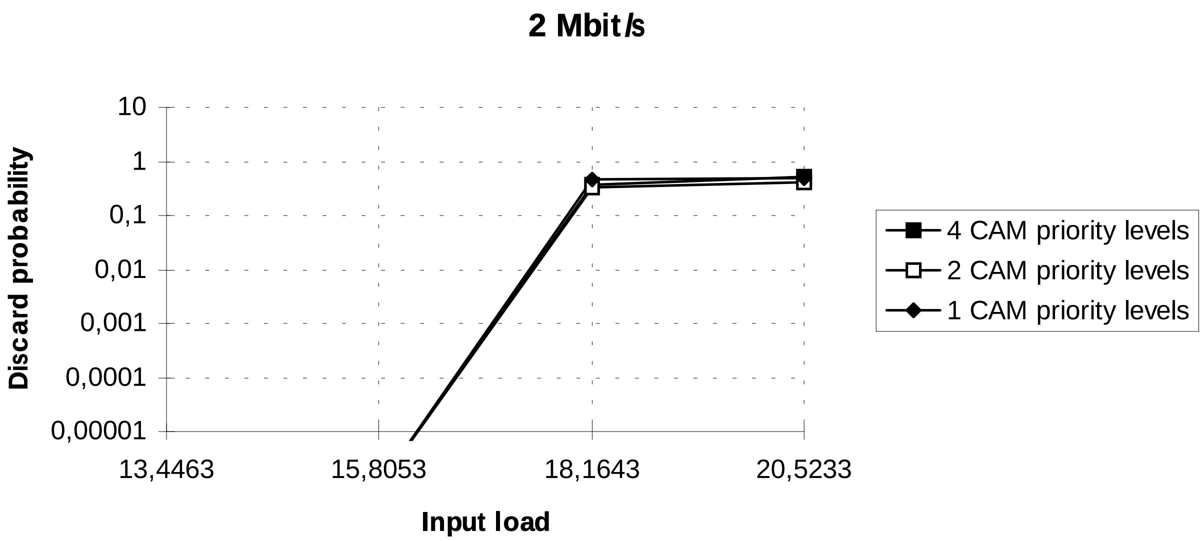


Figure A.18: Discard probability for the 2 Mbps traffics with multi-hop

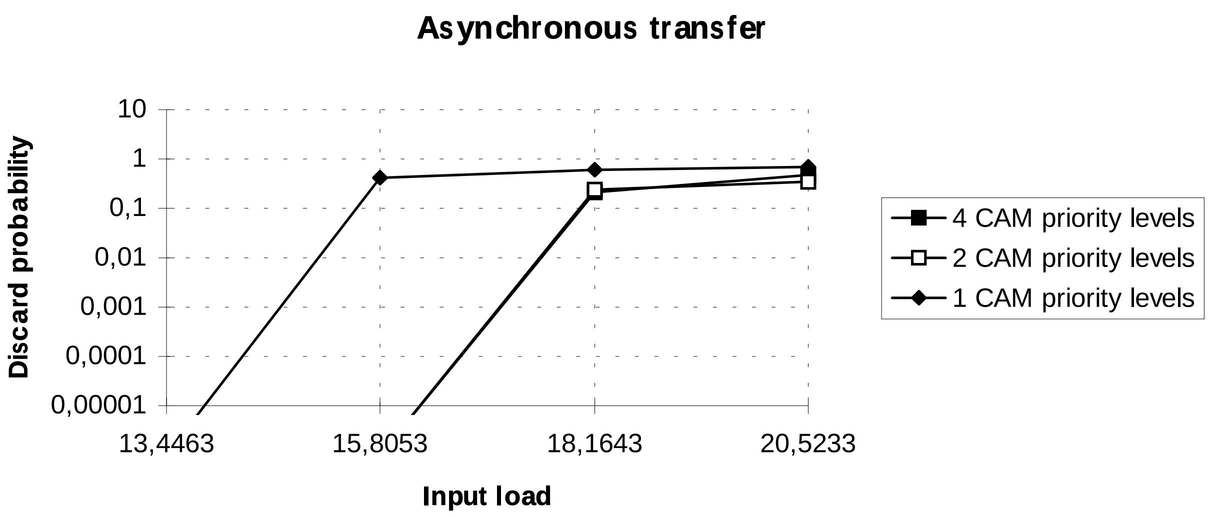


Figure A.19: Discard probability for the asynchronous traffics with multi-hop

A.1.4.2.4 Utilisation of priority

The effect of the MP QoS parameter is studied. In the simulations above three strategies are compared:

a) all the traffics use the MP: normal priority;

b) all the traffics use the normal priority except the 32 kbit/s links which uses the MP: high priority;

c) all the traffics use the normal priority except the 32 kbit/s links which uses the high priority and the asynchronous transfer which uses priority MP low1. Notice that this last strategy is not specified in prETS 300 652 [2] however it is a straightforward change, see subclause 7.5.2.

The following scenarios were tested:

- 10 links at 32 kbit/s, 10 links at 13 kbit/s, 1 link at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s (Poisson traffic, mean inter arrival 60,000 bits). The total load of the channel is therefore: 13,4 Mbit/s including all of the overheads for 6,38 Mbit/s of user data load;

- 15 links at 32 kbit/s, 15 links at 13 kbit/s, 1 link at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 15,8 Mbit/s including all of the overheads for 6,605 Mbit/s of user data load;

- 20 links at 32 kbit/s, 20 links at 13 kbit/s, 1 link at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 18,1 Mbit/s including all of the overheads for 6,83 Mbit/s of user data load;

- 25 links at 32 kbit/s, 25 links at 13 kbit/s, 1 link at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 20,5 Mbit/s including all of the overheads for 7,055 Mbit/s of user data load;

- 30 links at 32 kbit/s, 30 links at 13 kbit/s, 1 link at 2 Mbit/s and 3,93 Mbit/s of channel load for file transfer of 10 000 kbit/s. The total load of the channel is therefore: 22,8 Mbit/s including all of the overheads for 7,28 Mbit/s of user data load.

In the above figures the four different situations presented are denoted by the corresponding total load.

In figures A.20, A.21, A.22 and A.23 below, the ratio of discarded packets for the different traffics is presented. It can be seen that strategy 2 makes it possible to favour the 32 kbit/s links. Strategy 3 makes it possible to favour all the synchronous links. This can be easily explained by the fact that in this case the asynchronous packets can not reach the CAM priority 1.

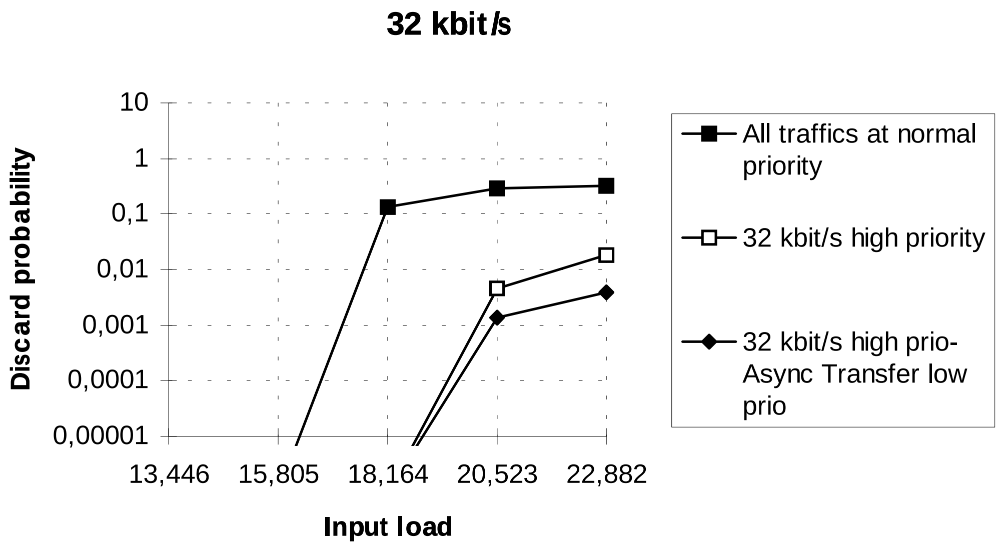


Figure A.20: Discard probability for the 32 kbps traffics with different user priorities

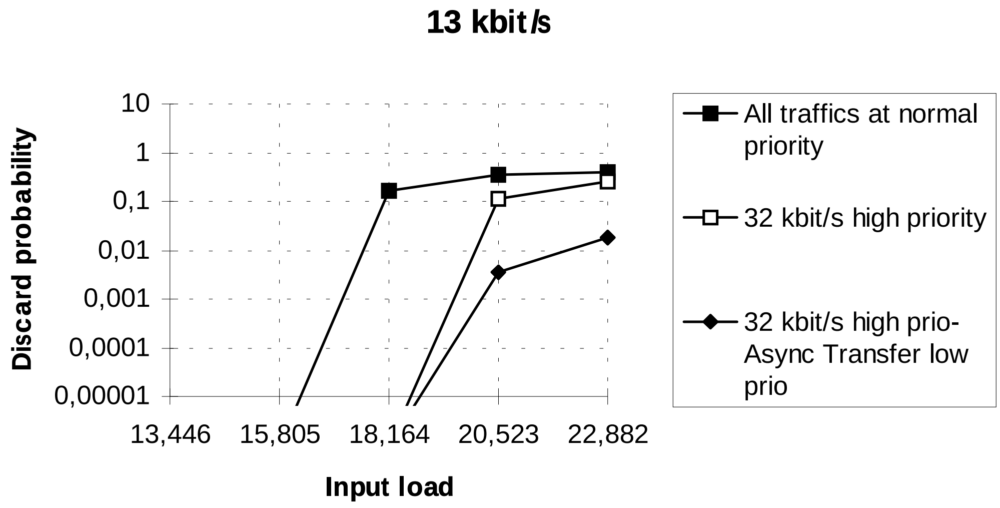


Figure A.21: Discard probability for the 13 kbps traffics with different user priorities

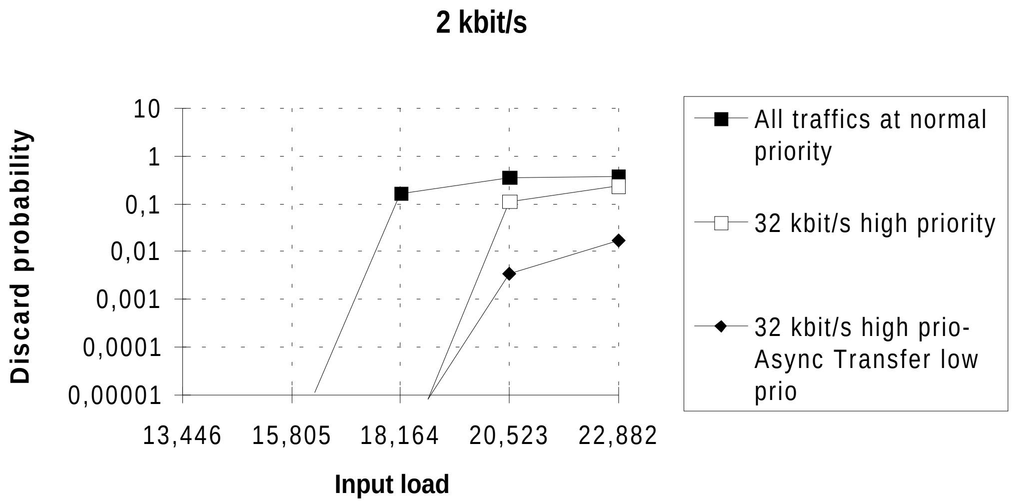


Figure A.22: Discard probability for the 2 Mbps traffics with different user priorities

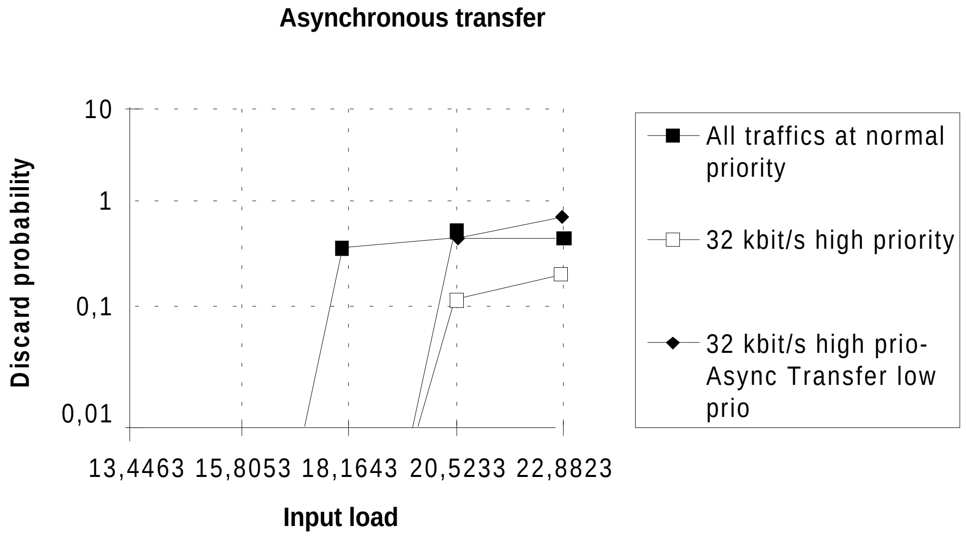


Figure A.23: Discard probability for the asynchronous traffics with different user priorities

A.1.4.2.5 Utilisation of the DP

The above simulations concerned the reference scenario.. In the following, we wish to control the discard of the 13 kbit/s link. The usual policy is compared with two techniques:

- we map this traffic to the smallest DTD. In the following simulations, 10 ms are used instead of the 20 ms of the reference scenario;

- the discard parameter is used. In the following simulations, DP = - 10 ms is used. Results are given in figures A.24, A.25, A.26 and A.27.

In these simulations it can be seen that the DP allows the discard ratio to be reduced of traffic for which this discard parameter is applied. Here, the discard parameter is applied to the 23 kbit/s traffic; not to the other traffic types. For the other traffic types the discard ratio is left roughly unchanged.

For convenience of reading, the discard parameter is expressed in time units, equating the amount of time which must be subtracted from RTD in CAM mapping.

NOTE: As QoS parameters the DP parameter could be expressed in dB, qualifying the maximum discard rate admissible for the traffic. The default discard rate is then assumed to be - 30 dB and leads to zero discrepancy on RTD. A discrepancy of 10 ms should correspond to 30 dB, therefore subtracting 10 ms from RTD should correspond to a theoretical - 60 dB discard rate.

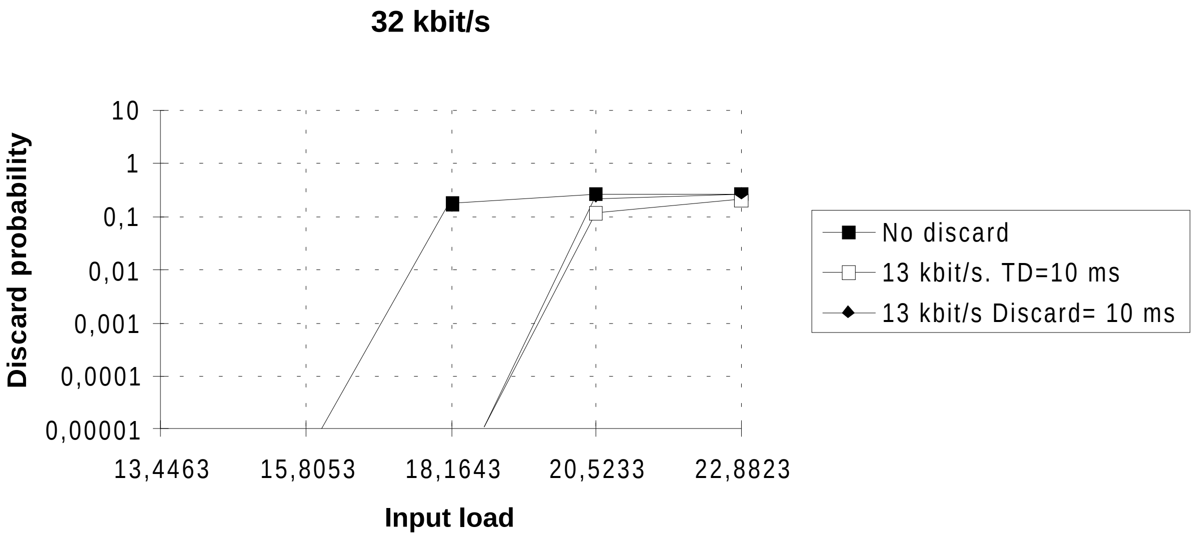


Figure A.24: Discard probability for the 32 kbps traffics with different discard parameters

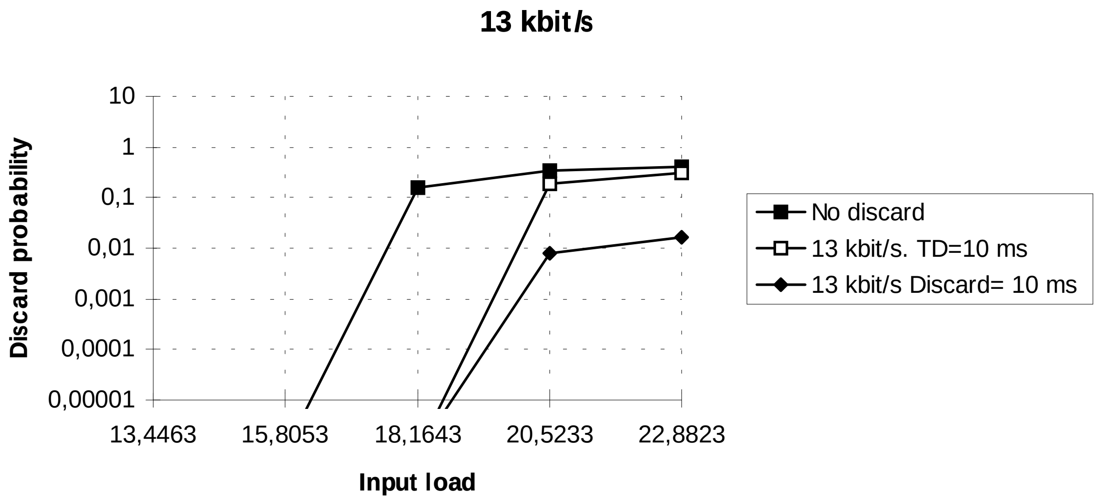


Figure A.25: Discard probability for the 13 kbps traffics with different discard parameters

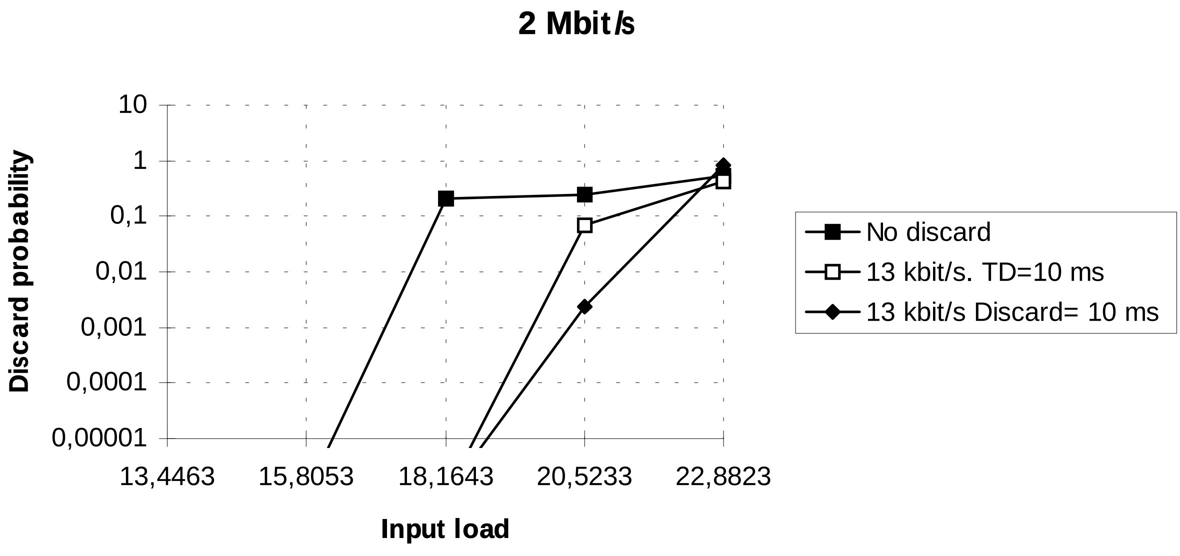


Figure A.26: Discard probability for the 2 Mbps traffics with different discard parameters

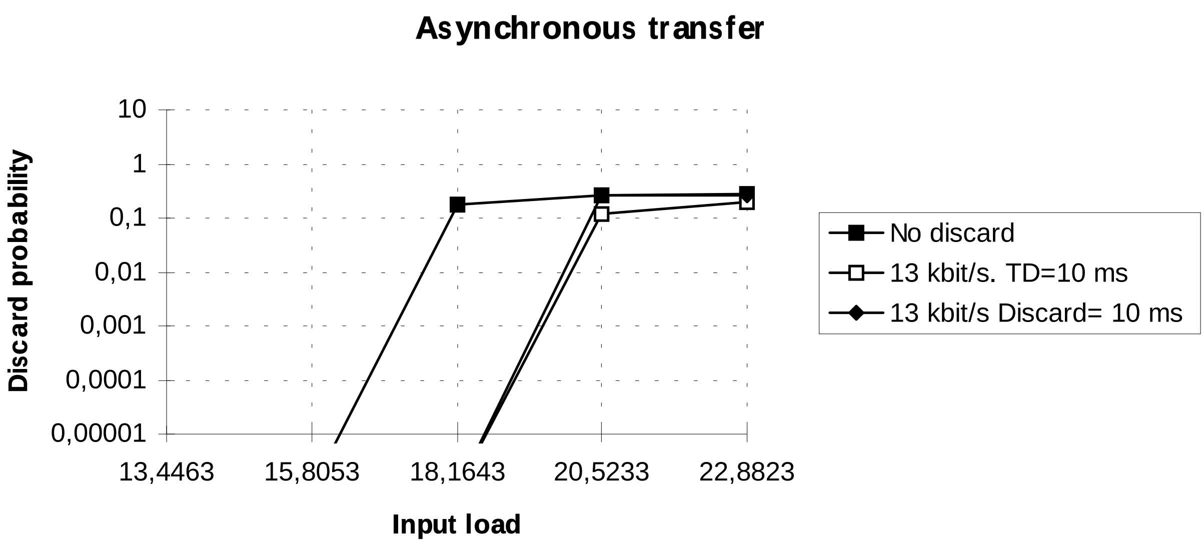


Figure A.27: Discard probability for the asynchronous traffic with different discard parameters

A.2 Other results

This subclause summarises other results obtained by an alternative simulator and from analytical models. The results agree well with those obtained via the main simulation tool.

A.2.1 Results obtained from an event driven medium access and radio channel simulator

A simulation tool has been developed for evaluating channel access mechanisms using realistic radio channel models (fading). The simulation tool involves priority access. Therefore the impact of CAM priorities can also be measured using this tool. The output result is the proportion of packets of a given priority to meet a given deadline.

The table below shows the impact of hidden nodes over short packet transmission. The packets are supposed 2 000 bits long (including all overheads).

NOTE: This is not consistent with A.2.2.3, note 1 which assumes that the overhead > 2 000 bits.

The deadline is 20 ms. One traffic type per priority level is considered. The lowest priority level is overloaded. The statistics address the highest priority since this level is attainable by every packet close to its deadline. There is no update of QoS parameters: the packets remain on the same priority level during their lifetime, and the packets are not discarded when they exceed their deadline.

In the absence of hidden nodes, the traffic on the highest level is not influenced by the traffic on the lower levels. In the presence of hidden nodes, there are mutual interferences which can be great since the lowest level is overloaded. The number of hidden nodes increases the order of magnitude of the probability of exceeding the deadline. In a different perspective, the impact of hidden nodes consists of decreasing the admissible load on the highest priority level in order to achieve the required quality of service.

The impact of the number of priority level is that for the same traffic and QoS profile, it reduces the load on the highest priority level and therefore reduces the probability of exceeding the deadline even in a hidden nodes configuration. Indeed a higher number of priority levels makes the traffic more resistant to hidden nodes. Of course the traffic load must remain within a certain limit as must the proportion of hidden nodes.

Table A.2 below, takes into account a proportion of hidden nodes given by a fading parameter. Fading is modelled as jitters of power transmission and is expressed in dB.

Table A.2: Impact of hidden nodes due to fading

Load of the low priority	Probability of access delay > 20 ms. Fading in dB = 0
up to 0,57	0
0,61	0,0476
0,62	0,2764
0,63	0,4

Load of the low priority	Probability of access delay > 20 ms. Fading in dB = 10
up to 0,4	0
0,43	0,000186
0,46	0,000348
0,49	0,000374
0,52	0,00361
0,55	0,004764
0,59	0,03
0,61	0,084
0,62	0,28
0,63	0,34

Load of the low priority	Probability of access delay > 20 ms. Fading in dB = 10
up to 0,03	0
0,04	0,001062
0,17	0,0286
0,22	0,1218
0,26	0,229
0,29	0,2904
0,31	0,2906
0,33	0,29415

A.2.2 Analytic model for evaluating effect of priority level

The aim of this analysis is to provide some ideas about the global performance of the system. We define simultaneous traffic types for various services and assume each of these services to satisfy its requirements. Here our aim is to specify compatible conditions on load and timing requirement.

The following analysis is based on very simple assumptions.

First, for simplicity a single hop system is considered.

Second, it is assumed that this system is underloaded and therefore that the priority with which a packet is transmitted is the priority at the arrival of the packet. This assumption is compatible with the evaluation of situations where timing constraints are satisfied.

A.2.2.1 Performance evaluation

Let us assume that we have n priority levels. Priority 1 will be the highest priority and priority n will be the lowest. Let us suppose that the utilisation factor k is ρ_k and L_k is the time duration of a packet of priority k. The mean waiting time for priority i: W_i is given by the first Cobham formula [8]:

$$W_i = \frac{\sum_{k=1}^{n} \rho_k L_k}{2\left(1 - \sum_{k=1}^{i-1} \rho_k\right)\left(1 - \sum_{k=1}^{i} \rho_k\right)}$$

The M/GI/1 corresponds to a perfect CAM. Therefore the previous formula gives an upper bound to the delays.

A.2.2.2 How does the number of priority levels influence the received service in a prioritised system?

The priority technique is inherently a hierarchical model. The higher priority packet always receives priority over packets of a lower priority traffic. As a result, packets of a higher priority have lower delays than packets of a lower priority.

There is a very general law which states that the performance in terms of access delay of a system depends on the cumulative load of packets of traffic of higher or equal priority. This behaviour which is very intuitive can be seen in the Cobham formula, since the average delay of the traffic i is a function of

$$\sum_{k=1}^{i} \rho_k .$$

Actually the mean access delay depends on the cumulative load of the traffic of higher or equal priority. It is, of course, the same for the whole distribution of the delays. Therefore if we wish to discriminate between different requirements we must provide different priority levels.

This property is the key to understanding what another priority level adds to the system. Although the explanation can be generalised, the following example is presented with only two priority levels. Let us suppose that the distribution of the delay of an application A requires that the network is used up to a given maximum load ρ_A. We note that this only makes sense if the bandwidth required by application A produces a load less than ρ_A. It is very likely that another service B, with a lesser demand on the delay distribution, can be offered up to a total load of ρ_B. In such a case ρ_A of the bandwidth can be used for application A, actually more than one single application of type A may be used if the total load stays below ρ_A while ρ_B-ρ_A of the channel will be used for the second application. That holds if two priority levels can be offered otherwise the two applications must be merged together but only being able to use ρ_A of channel for both of the applications since the quality of service can not be differentiated. This reasoning can obviously be repeated recursively.

A.2.2.3 Numerical computations

In the following, figures and parameters described in subclause A.1.4.2.1 are used. Table A.3 summarises the results.

Traffic types:

- synchronous traffic:
    - 32 kbits/s links composed of 320 bit packets. Thus within a link a packet is generated every 10 ms. For the computations below 20 such links are considered;
    - 13 kbits/s links composed of 260 bits packets. Thus within a link a packet is generated every 20 ms. For the computations below 20 such links are considered;
    - 2 Mbits/s links composed of 10 000 bit packets. For the computations below 1 such a link is considered;
- asynchronous traffic: file transfer:
    - 10 000 bit packet. For the computations below a Poisson arrival with mean inter arrival 60 000 bits is considered.

NOTE 1: The following overheads are assumed: the overhead due to the MAC fields and the coding scheme, the constant transmission overhead (2 146 bits), the overhead due to the prioritisation phase. The length of the packets used below take into account these overheads.

NOTE 2: It is assumed that the CAP is selected according to the initial time to live. The 32 kbits links receive priority 1, the 13 kbits/s links receive priority 2, the 2 Mbit/s links and the asynchronous packet priority receive priority 4.

Table A.3: Average waiting times computed via Cobham formulas

ρ_i	ρ_i	$\Sigma_1^i \rho_i$	L_i (bits)	W_i (ms)	20 W_i
20 links at 32 kbits	0,31	0,31	3 650	0,24	4,97
20 links at 13 kbits	0,17	0,48	3 906	0,47	9,47
1 links at 2 Mbits	0,13	0,61	15 826	0,84	16,75
Asynchronous transfer	0,26	0,87	15 826	3,47	69,38

The previous computations show that if it is assumed that the distribution of the delay does not exceed 20 times the average delay, the requirements of the previous services are always met. These computations however suppose that the EDF scheme does not modify the initial channel access priority of packets.

A.2.3 Analytical analysis of an EDF scheme driving different CAM priority levels

This subclause provides some insight into an EDF algorithm driving different CAM priorities compared to no scheduling at all. The comparison concerns the probability of exceeding a given deadline.

The scenario analysed encompasses a potential infinite number of stations and a Poisson, traffic of rate less than 1. All the packets hold the same relative deadline T. It is assumed that each node holds at most one packet per time which its submits to channel contention. To this end the packets are virtually stored in a global queue. Two cases are compared:

The contention algorithm uses perfect CAM priorities: packets are served in a FIFO basis (same deadline);

The contention algorithm does not use CAM priorities: packets are served in a random order.

To achieve the determination of the model, the service of packets is exponential with rate 1.

A.2.3.1 The perfect scheduler algorithm

A FIFO M/M/1 queue is analysed. We define the waiting time W_F of a random packet as a random variable. The Laplace-Stieljes transform $W_F(s) = E\left(e^{-sW_F}\right)$ of this random variable is known to have the following expression:

$$W_F(s) = \frac{1-\lambda}{1-\lambda+s}$$

From the above expression, it results that $W_F(s)$ is singular when $s = \lambda - 1$, therefore the distribution of W_F has an exponential tail of rate $1-\lambda$. In other words: $-\log \Pr\{W_F > T\} \approx (1-\lambda)T$. A more careful look shows that W is in fact exactly exponential of rate $1-\lambda$.

A.2.3.2 Random order packet service

The Laplace-Stieljes transform of the waiting time in the random selection is $W_R(s) = w(\lambda, s)$ where $w(\lambda, s)$ is the solution of the differential equation:

$$\lambda \frac{w(\lambda, s)}{z} + \left[(s+1+\lambda)z - z^2 - \lambda\right] w_z(z,s) = \frac{z}{1-z}$$

The notation w_z is reduced to $\frac{\partial w}{\partial z}$. The resolution of such an equation is based on the determination of the roots in z of the polynomial $(s+1+\lambda)z - z^2 - \lambda$. The roots are $\frac{1}{2}\left(1+\lambda \pm \sqrt{\Delta(s)}\right)$ with $\Delta(s) = (1-\lambda)^2 + s^2 + 2(1+\lambda)s$. Thanks to this perspective, $W_R(s)$ is singular for $s = -\left(1-\sqrt{\lambda}\right)^2$ (the first non-negative root of $\Delta(s)$. It gives the asymptotic expansion: $-\log \Pr\{W_F > T\} \approx (1-\lambda)^2 T$.

A.2.3.3 Results of comparison

The figures below illustrate the probability of delay excess versus the load. The delay threshold is either 100 average packet transmission times (figure A.28), 40 packet transmission times (figure A.29), or 10 packet transmission times (figure A.30). The probability is given in dB; the load is given in percentage of channel capacity.

The very large gap between the FIFO service and the random service demonstrates the usefulness of the EDF scheme.

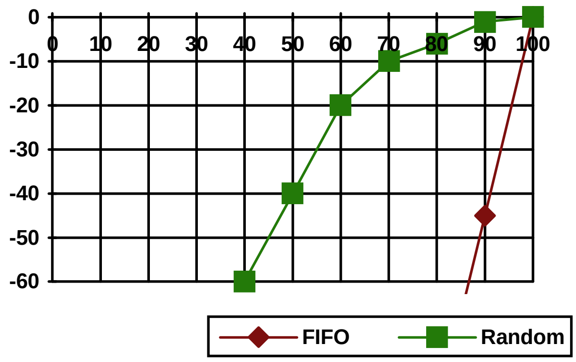


Figure A.28: Probability of exceeding delay threshold equal to 100 transmission times

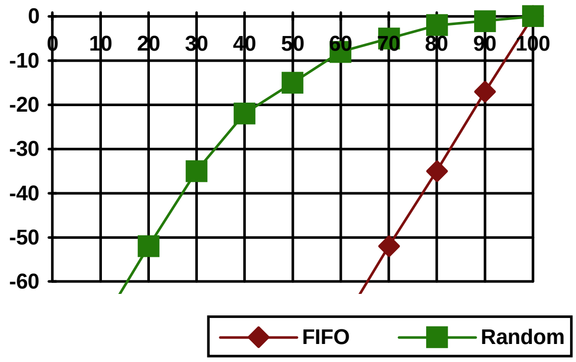


Figure A.29: Probability of exceeding delay threshold equal to 40 transmission times

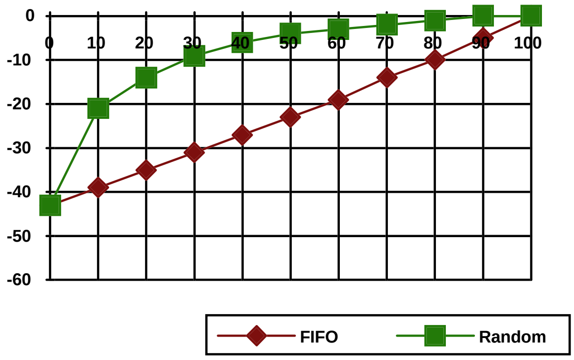


Figure A.30: probability of exceeding delay threshold equal to 10 transmission times

Annex B: Bibliography

The work in this ETR was based on the following unpublished documents:

- DTR/SMG-050201: "Framework for services to be supported by the Universal Mobile Telecommunications System (UMTS)".

- RES10/94 CAM Performance Evaluation of CAM protocols after a stretched pulse. Philippe Jacquet, Paul Mühlethaler. October 22, 1994.

History

Document history	
October 1995	First Edition
February 1996	Converted into Adobe Acrobat Portable Document Format (PDF)

ISBN 2-7437-0279-6
Dépôt légal : Octobre 1995