Abstract

WLAN and wireless ATM are emerging technologies for wideband wireless local access. Two standards, IEEE 802.11 and ETSI HIPERLAN Type 1, are currently available for WLAN; the standardization process of wireless ATM, involving, among others, the Wireless ATM Group of the ATM Forum and the Broadband Radio Access Networks project of ETSI, is ongoing and related standards are expected to be released by mid-1999. This article focuses on MAC protocol aspects of wireless local access networks. It first investigates, from a traffic performance point of view, the MAC protocol of the IEEE 802.11 and ETSI HIPERLAN Type 1 standards, and then verifies to what extent these MAC protocols are suitable for wireless ATM. The analysis is then extended by considering a new MAC protocol, Dynamic Slot Assignment (DSA++), which has been designed to explicitly support ATM technology over the radio interface. DSA++ is a candidate for the ETSI HIPERLAN Type 2 standard, a developing ETSI standard for wireless ATM.

MAC Protocols for Wideband Wireless Local Access: Evolution Toward Wireless ATM

GIUSEPPE ANASTASI, LUCIANO LENZINI, AND ENZO MINGOZZI
UNIVERSITY OF PISA
ANDREAS HETTICH AND ANDREAS KRÄMLING
AACHEN UNIVERSITY OF TECHNOLOGY

Wireless local area network (WLAN) is an emerging technology that provides wideband wireless local access, thus offering intercommunication capabilities to mobile applications, and also an alternative, attractive way of setting up computer networks in environments where cable installation is expensive or infeasible. Currently, this technology is supported by two standards: the 802.11 standard [1], developed by the IEEE 802 LAN standards organization, and High Performance Radio LAN (HIPERLAN) Type 1 [2], defined by the European Telecommunications Standards Institute (ETSI) RES-10 Group. IEEE 802.11 considers data rates up to 2 Mb/s, and defines two network topologies: "infrastructure-based" (mobile terminals communicate with the backbone network through an access point) and "ad hoc" (mobile terminals communicate with each other without connectivity to the wired backbone network), while HIPERLAN considers a data rate of 23.5 Mb/s and the ad hoc topology only. Although both standards were mainly designed for conventional LAN data traffic, they nevertheless enable, to some extent, quality of service (QoS) guarantees. In particular, IEEE 802.11 considers an infrastructure topology and a priority scheme in the point coordination function (PCF), while HIPERLAN defines a channel access priority scheme based on the lifetime of packets [3].

The work performed by the Italian authors described in this article has been carried out under the financial support of the Italian Ministero dell'Università e della Ricerca Scientifica e Tecnologica (MURST) in the framework of the MOSAICO "Design Methodologies and Tools of High Performance Systems for Distributed Applications" Project.

[1] Specifically, wireless ATM will support all service categories defined by the ATM Forum and the International Telecommunication Union — Telecommunication Standardization Sector (ITU-T), which are currently CBR, rt-VBR, nrt-VBR, UBR, and ABR, as defined in the traffic management specification or the respective ITU-T specifications [8].

More recently, much attention has been focused on the extension of asynchronous transfer mode (ATM) to wireless communications, thus leading to wireless ATM technology [4]. The standardization process of wireless ATM is ongoing, and it involves, among others, the Wireless ATM Group (WAG) of the ATM Forum and the Broadband Radio Access Networks (BRAN) project of ETSI [5]. The main goal of such standardization efforts is to develop a technology for wideband wireless local access which will explicitly include ATM features in the radio interface, thus combining support of user mobility with statistical multiplexing and QoS guarantee provided by wired ATM networks.[1] This should enable to reduce the complexity of interworking between the wireless access network and the wired (ATM) backbone and, consequently, attain a higher level of integration [3, 6].

One of the key issues of wireless ATM is the choice of an appropriate MAC protocol which has to incorporate the wired ATM capability of integrating a variety of traffic types with different QoS requirements. This article is first intended to investigate the viability of operating wireless ATM with the (possibly enhanced) MAC protocols from the IEEE 802.11

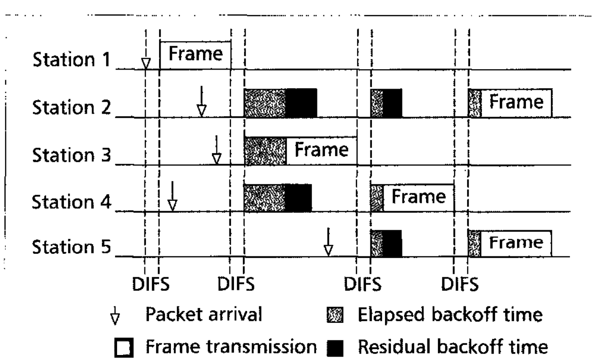


Figure 1. *Basic access mechanism.*

and ETSI HIPERLAN Type 1 standards, because of their popularity and large diffusion. More specifically, the article highlights, from a traffic performance point of view, their pros and cons as medium access control (MAC) protocols for WLAN and verifies to what extent they can provide different traffic types with different QoS guarantees as required with wireless ATM. The analysis is then extended by considering a new MAC protocol, Dynamic Slot Assignment (DSA++) [7], explicitly designed to support ATM technology over the radio interface. DSA++ is currently under investigation as a candidate for the ETSI HIPERLAN Type 2 standard, a developing ETSI standard for wireless ATM. The performance analysis of the MAC protocols involves the following steps:

- Defining a set of models for the traffic demand to analyze the protocol behavior in a realistic operations scenario. Specifically, an ON/OFF model, where the ON and OFF periods are distributed according to an exponential (MMPP) or a Weibull distribution, is used for Web traffic, while models based on Markov chains are used for voice and video (MPEG and low-bit-rate H.263) traffic models.
- Defining a set of performance measures for characterizing the QoS provided by the considered protocols for non-real-time and real-time traffic.
- Estimating, by means of ad hoc simulators implemented in C++, the performance of the protocols.

The article is organized as follows. The next section introduces the channel access schemes of all three protocols. In the section after that, a variety of traffic demand models, including non-real-time (data) as well as real-time (voice and video) traffic, are described, and we go on to define a set of performance measures to characterize the QoS provided by the protocols. The performance figures are derived via simulations, and results are given for IEEE 802.11, HIPERLAN, and DSA++, respectively. Finally, some conclusions are drawn.

The 802.11, HIPERLAN, and DSA++ MAC Protocols

This section presents an overview of the channel access mechanisms designed for the IEEE 802.11 and ETSI HIPERLAN (Type 1) standards. Furthermore, the DSA++ protocol, a candidate MAC protocol for wireless ATM currently under discussion in ETSI BRAN, is presented.

IEEE 802.11

The IEEE 802.11 MAC protocol provides two service types: *asynchronous* and *synchronous* (or, rather, *contention-free*). These types of service can be provided on top of a variety of physical layers and for different data rates. The asynchronous type of service is mandatory whereas the synchronous type of service is optional.

The asynchronous type of service is provided by the distributed coordination function (DCF), which implements the basic access method of the IEEE 802.11 MAC protocol and is also known as the carrier sense multiple access with collision avoidance (CSMA/CA) protocol.

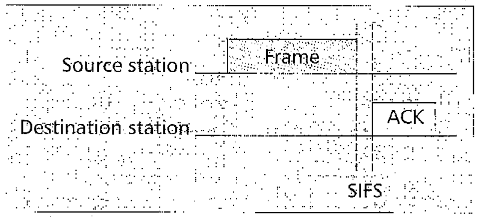


Figure 2. *Acknowledgment mechanism.*

Contention-free service is provided by the point coordination function (PCF) which basically implements a polling access method. The PCF uses a point coordinator which cyclically polls wireless stations,[2] giving them the opportunity to transmit. Unlike the DCF, the implementation of the PCF is not mandatory. Furthermore, the PCF itself relies on the asynchronous service provided by the DCF.

According to the DCF (Fig. 1), a station must sense the medium before initiating transmission of a packet. If the medium is sensed as being idle for a time interval greater than a distributed interframe space (DIFS), the station transmits the packet. Otherwise, the transmission is deferred and the backoff process is started. Specifically, the station computes a random time interval, the *backoff interval*, uniformly distributed between zero and a maximum called the *contention window* (CW). This backoff interval is then used to initialize the *backoff timer*. This timer is decremented only when the medium is idle, whereas it is frozen when another station is transmitting. Specifically, each time the medium becomes idle, the station waits for a DIFS and then periodically decrements the backoff timer. The decrementing period is referred to as the *slot time*, which corresponds to the maximum round-trip delay between two stations controlled by the same access point.

As soon as the backoff timer expires, the station is authorized to access the medium. If two or more stations start transmission simultaneously, a collision occurs.[3] Unlike wired networks (e.g., with CSMA/CD), in a wireless environment collision detection is not possible. Hence, as shown in Fig. 2, a positive acknowledgment (ACK) is used to notify the sending station that the transmitted frame has been successfully received. The transmission of the ACK is initiated at a time interval equal to the short interframe space (SIFS) after the end of reception of the previous frame. Since the SIFS is, by definition, shorter than the DIFS,[4] the receiving station does not need to sense the medium before transmitting the ACK.

If the ACK is not received, the station assumes that the transmitted frame was not successfully received, and hence schedules a retransmission and enters the backoff process again. However, to reduce the probability of collisions, after each unsuccessful transmission attempt the CW is doubled until a predefined maximum (CW_{max}) is reached. After a (successful or unsuccessful) frame transmission, if the station still has frames queued for transmission, it must execute a new backoff process.

In radio systems based on medium sensing, a phenomenon known as the *hidden station* problem may occur. This problem arises when a station is able to successfully receive frames from two different stations but the two stations cannot receive

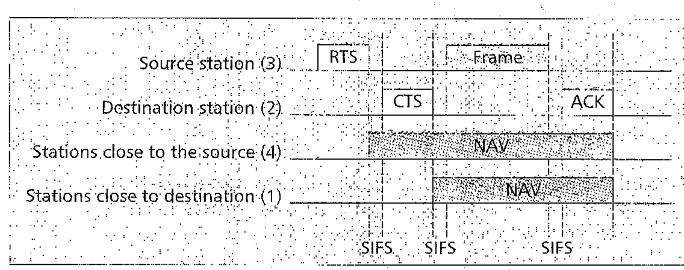
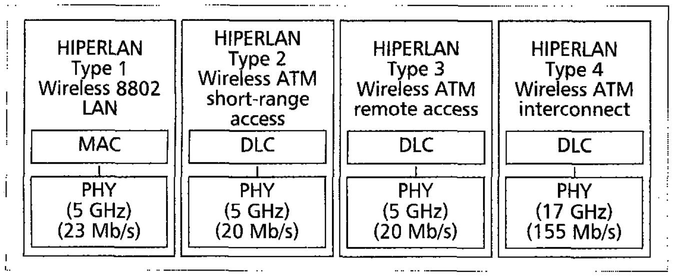


Figure 3. *RTS/CTS mechanism.*

[2] *In the article the terms wireless* stations, terminals, *and* nodes *are used interchangeably.*

[3] *The packet capture is not considered here.*

[4] *The DIFS is defined as DIFS = SIFS + 2 Slot-times.*



■ **Figure 4.** *Overview of HIPERLAN types.*

signals from each other. In this case a station may sense the medium as being idle even if the other one is transmitting. This results in a collision at the receiving station.

To deal with the hidden station problem the IEEE 802.11 MAC protocol includes an optional mechanism based on the exchange of two short control frames (Fig. 3): a Request To Send (RTS) frame, which is sent by a potential transmitter to the receiver, and a Clear To Send (CTS) frame, which is sent by the receiver in response to the received RTS frame. If the CTS frame is not received within a predefined time interval, the RTS frame is retransmitted by executing the backoff algorithm described above. After a successful exchange of RTS and CTS frames, the data frame can be sent by the transmitter after waiting for a SIFS.

The RTS and CTS frames include a *duration field* that specifies the time interval necessary to completely transmit the data frame and the related ACK. This information is used by stations which can hear either the transmitter or the receiver to update their *net allocation vector* (NAV), a timer which, unlike the backoff timer, is continuously decremented (i.e., irrespective of the status of the medium). Since stations that can hear either the transmitter or the receiver refrain from transmitting until their NAV has expired, the probability of a collision occurring due to a hidden station is reduced. Of course, the drawback of using the RTS/CTS mechanism is an increased overhead which may be significant for short data frames.

Furthermore, the RTS/CTS mechanism can be regarded as a way to improve the MAC protocol performance. In fact, when the mechanism is enabled, collisions can obviously occur only during the transmission of the RTS frame. Since the RTS frame is usually much shorter than the data frame, wastage in bandwidth and time due to the collision is reduced.

In both cases the effectiveness of the RTS/CTS mechanism depends on the length of the data frame to be protected. It is reasonable to think that the RTS/CTS mechanism improves the performances when data frame sizes are larger than the size of the RTS frame. Consequently, the RTS/CTS mechanism relies on a threshold, the *RTS threshold*: the mechanism is enabled for data frame sizes over the threshold and disabled for data frame sizes under the threshold. In order to support time-bounded services the IEEE 802.11 standard defines the PCF to permit a *single* station in each cell to have priority access to the medium. This is implemented through the use of the PCF interframe space (PIFS) and a beacon frame that notifies all the other stations in the cell not to initiate transmissions for the length of the contention-free period (CFP). Having silenced all the stations, the PCF station can then allow a given station to have contention-free access through the use of an (optional) polling frame sent by the PCF station. Note that the length of the CFP can vary within each CFP repetition interval depending on the system load.

ETSI HIPERLAN

ETSI has identified the need for a family of HIPERLAN standards, a collective reference to high-performance radio LANs, to meet the demand for better and faster wireless networking from all kinds of users. Figure 4 gives an overview of the HIPERLAN types with the operating frequencies and indicative data transfer rates on the radio interface.

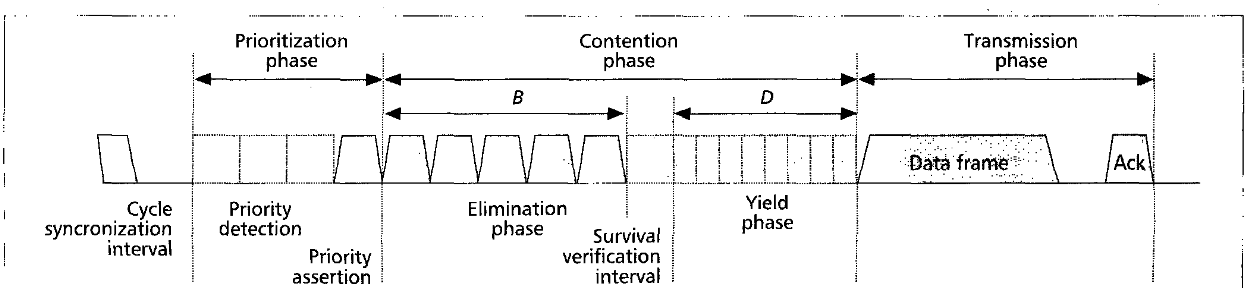
In the following the HIPERLAN Type 1 MAC protocol (hereafter HIPERLAN) and the DSA++ protocol, which is a candidate for HIPERLAN Type 2, will be addressed. HIPERLAN Types 3 and 4 are outside the scope of this article.

HIPERLAN Type 1, Wireless 8802 LANs – The HIPERLAN channel access mechanism [2] is based on channel sensing and a contention resolution scheme called *Elimination Yield — Non-Preemptive Priority Multiple Access* (EY-NPMA). According to the HIPERLAN channel access mechanism, channel status is sensed by each station[5] in the network. If the channel is sensed as being idle for at least 1700 bit periods, the channel is considered free, and the station is allowed to immediately start transmission of the data frame. Each data frame transmission must be explicitly acknowledged by an ACK from the destination station.

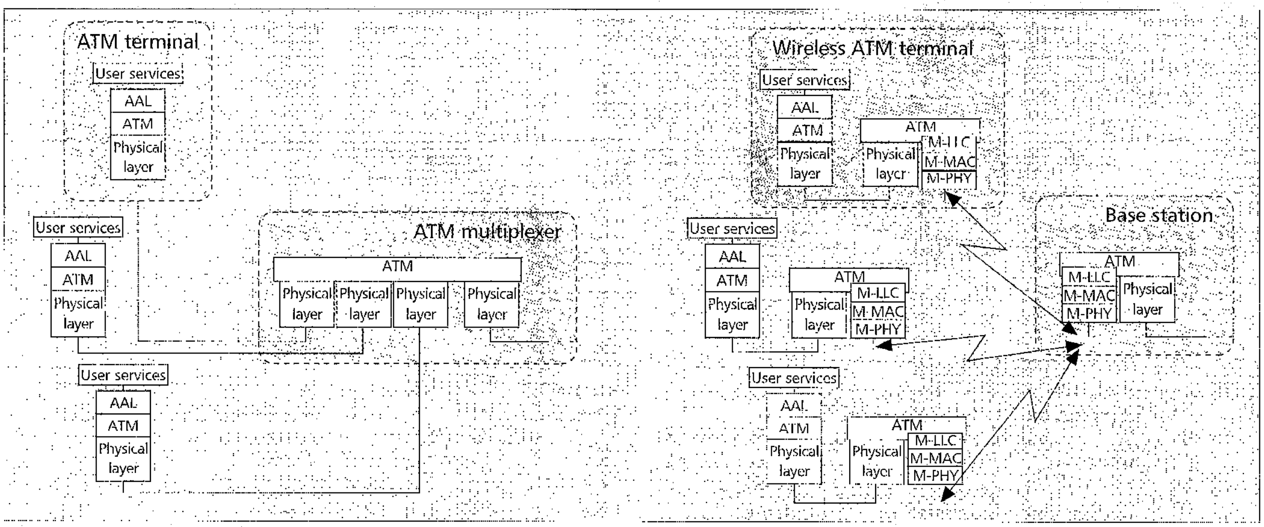
On the other hand, whenever a frame transmission is desired and the channel is considered not free, then, according to HIPERLAN terminology, a channel access with synchronization takes place. Synchronization is performed at the end of the previous transmission interval, and a channel access cycle then begins according to the EY-NPMA scheme. The channel access cycle consists of three phases: prioritization, contention, and transmission. Figure 5 shows an example of a channel access cycle with synchronization.

The aim of the prioritization phase is to allow only contending stations with the highest-priority frames to participate in the next phase. In HIPERLAN a channel access mechanism (CAM) priority level h is assigned to each frame. Priority levels are numbered from 0 to $H-1$, with 0 denoting the highest priority level. The prioritization phase consists of at

[5] *For the sake of uniformity we use the term* station *for HIPERLAN mobile users, although the standard uses the term* node.



■ **Figure 5.** *Channel access cycle with synchronization.*



■ **Figure 6.** *Correspondence between radio cell and ATM multiplexer.*

most H prioritization slots, each 256 bit periods long. Each station that has a frame with CAM priority level h senses the channel for the first h prioritization slots (priority detection). If the channel is idle during this interval, the station transmits a burst in the h + 1th slot (priority assertion), and it is admitted to the contention phase; otherwise, it stops contending and waits for the next channel access cycle. The contention phase starts immediately after transmission of the prioritization burst, and consists of two further phases: elimination and yield.

	High user priority	Low user priority
NMRL < 10 ms	0	1
10 ms < NMRL < 20 ms	1	2
20 ms < NMRL < 40 ms	2	3
40 ms < NMRL < 80 ms	3	4
80 ms < NMRL	4	4

■ **Table 1.** *Computation of the CAM priority.*

The elimination phase consists of at most n elimination slots, each 256 bit-periods long, followed by a 256-bit-period-long elimination survival verification slot. Starting from the first elimination slot, each station transmits a burst for a number B of elimination slots, according to the following truncated geometric probability distribution function:

$$\Pr\{B=b\}=\begin{cases}(1-q)q^b & 0\le b<n\\ q^n & b=n\end{cases}$$

After the end of the burst transmission, each station senses the channel for the duration of the elimination survival verification slot. If the channel is sensed as being idle, the station is admitted to the yield phase; otherwise, it drops itself from contention and waits for the next channel access cycle. The yield phase starts immediately after the end of the elimination survival verification interval and consists of at most m yield slots, each 64 bit periods long. Each station listens to the channel for a number D of yield slots before beginning transmission (if allowed). D is an r.v. with a truncated geometric distribution as follows:

$$\Pr\{D=d\}=\begin{cases}(1-p)p^d & 0\le d<m\\ p^m & d=m\end{cases}$$

If the channel is sensed idle during the yield listening interval, the station is allowed to begin the transmission phase; otherwise, the station loses contention and waits for the next channel access cycle.

The elimination and yield phases are complementary. As shown in [9], the elimination phase drastically reduces the number N of stations taking part in the channel access cycle; remarkably, this result is achieved almost independent of N. The yield phase, which performs well with a small number of contending stations, further reduces the number of stations allowed to transmit, possibly even to one. Furthermore, with EY-NPMA at least one station will always be allowed to transmit.

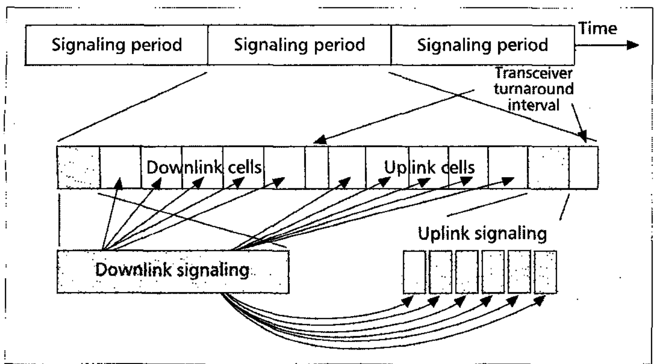
Real-time traffic transmission is supported in HIPERLAN by dynamically varying the CAM priority depending on the user priority and packet residual lifetime as reported in Table 1.

The user priority is an attribute assigned to each packet according to the type of traffic it carries; it determines the maximum CAM priority value the packet may eventually reach.

The residual packet lifetime is the time interval in which the transmission of the packet must occur before the packet must be discarded. Since multihop routing is supported within the standard, the residual packet lifetime is normalized (NMRL = normalized MPDU residual lifetime) to the number of hops the packet has to traverse to reach the final destination.

HIPERLAN Type 2, Short-Range Wireless Access to ATM Networks — HIPERLAN Type 2 is intended to provide local wireless access to ATM infrastructure networks by terminals that interact with access points which, in turn, are connected to an ATM switch or multiplexer. Such a wireless ATM access network must be able to provide the QoS, including the required data transfer rates, users expect from a wired ATM network. The specification of HIPERLAN Type 2 is currently carried out by ETSI BRAN, and the first functional standards will be available mid-1999.

Two MAC protocols are promising candidates for HIPERLAN Type 2. Both of these MAC protocols (DSA++ [7] and MASCARA [10]) use a combination of contention-based and contention-free access on the physical channel and are based on the same principle. In this article, therefore, only the DSA++ protocol will be explained.



■ **Figure 7.** *Structure of the signaling period.*

Dynamic Slot Assignment (DSA++)

The rationale behind the development of the Dynamic Slot Assignment protocol was to extend the ATM like statistical multiplexing to the radio interface in order to fulfill the requirements of wireless users.

As shown in Fig. 6, the radio cell with its central base station (BS) and wireless terminals (WTs) can be seen as a distributed, virtual ATM multiplexer with a radio interface inside. This leads to the idea of using a centralized master–slave type of MAC protocol, where the BS, as the master of a radio cell, schedules the contention-free transmission of ATM cells on the uplink as well as the downlink.

The virtual ATM multiplexer represents a distributed queuing system with queues inside the WTs (for uplink cells) and the BS (for downlink cells). As in fixed ATM networks with a relatively low data rate (e.g., 20 Mb/s), the QoS requirements of real-time-oriented services can only be supported if the transmission order of ATM cells is based on the waiting time inside the queues. As a consequence, the BS has to have current knowledge of the capacity requirements of the mobile WTs. This can be achieved by piggybacking onto uplink ATM cells the instantaneous requirements of each mobile WT. However, it may not be possible to piggyback the newest requirements (e.g., the mobile WT is idle). In this case, WTs are provided with special uplink signaling slots so that they can transmit their capacity requests to the BS according to a random access scheme.

The DSA++ protocol is implemented on top of a time-division multiple access (TDMA) channel. Time slots may carry either a signaling burst or one ATM cell along with the additional signaling overhead of the physical layer. A time-division duplexing system is implemented to build up the uplink and downlink channels.

Time slots are grouped together into signaling periods. The frame structure of a signaling period is reported in Fig. 7. The length of each signaling period, as well as the ratio between the uplink and downlink sections, is variable and assigned dynamically by the BS to cope with the current load of the system. Each signaling period consists of four phases.

Downlink Signaling – The downlink signaling burst is transmitted from the BS to the WTs and opens a signaling period of a specific length, giving information about the structure and slot assignments of the signaling period. More specifically, the downlink signaling informs the WTs about the number of slots in the other three phases, and contains at least:
- A *reservation message* for each uplink slot of the signaling period
- An *announcement message* for each downlink slot of the signaling period
- A control message to implement the collision resolution algorithm of the random access

Downlink Cells – In this phase the downlink cells are transmitted contention-free from the BS to the WTs.

Uplink Cells – Since each of these slots is assigned to specific WTs, in this phase uplink cells are transmitted contention-free from the WTs to the BS.

Uplink Signaling – During this phase, which is carried out via a sequence of short slots, the WTs have the possibility to access the channel in order to signal their capacity requests to the BS.

As stated above, random access is used for transmission of the capacity requests of the WTs. To be able to guarantee the QoS requirements of the connections, fast collision resolution with a deterministic delay is essential. Since all WTs (possible candidates to transmit via random access) are known by the BS, an identifier splitting algorithm can be used, which leads to short and deterministic delays to resolve any collision [11, 12]. A splitting algorithm groups the terminals into sets. All terminals of a set are allowed to transmit in a specific slot. A transmission will only be successful if exactly one terminal of a set transmits. After a collision, the set is divided into subsets according to the order of the splitting algorithm. In the case of an identifier splitting algorithm the follow-up subset is determined by the identifier of the terminal.

In Fig. 8 an example of a binary identifier splitting algorithm with an identifier space of dimension $n = 4$ is shown. τ_p is the duration of a period able to offer any number of random access slots.

According to the DSA++ protocol, at the beginning of each frame the identifier space (of size N) is divided into a variable number t of consecutive intervals, and a random access slot is assigned to each interval. The lth interval starts with terminal i_l and ends with terminal $i_{l+1} - 1$, with $i_1 = 0$ and $i_t = N - 1$. The downlink signaling burst signals the interval division to the WTs by transmitting the start identifier i_l of each interval. Due to the limited and known number of WTs served by the BS, the maximum time required to resolve a collision is limited. In [7] the authors show that the solution time of a collision can be reduced drastically if an estimation of the transmission probability of each terminal is used to determine the size of the subsets as well as the splitting order; see [7] for further details.

Since all ATM service classes will be supported, the coding of the capacity requests as well as the scheduling algorithm depends on the service class. For real-time-oriented service classes such as constant bit rate (CBR) and real-time variable bit rate (rt-VBR), an earliest due date strategy is used [13]; for service classes such as available/unspecified bit rate (ABR/UBR), Fair Weighted Queuing/FCFS can be used as a service strategy [14].

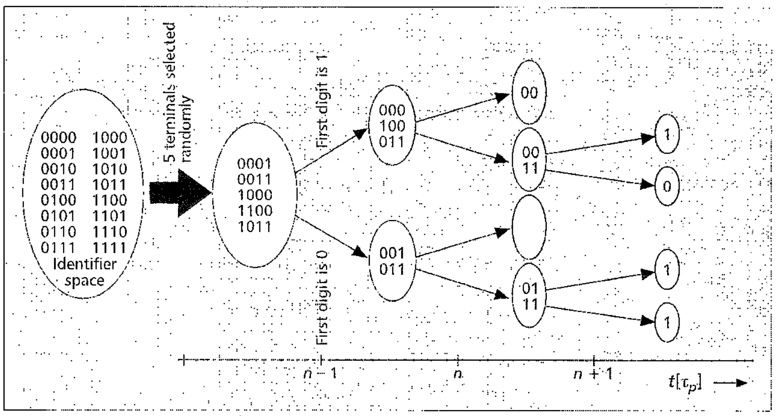
Traffic Models

The MAC protocols considered in this article are analyzed in the performance analysis with the traffic models introduced in the next subsections.

Data Traffic

When modeling data traffic, packet arrival processes are often assumed to be Poisson. However, recent studies based on experimental measures have shown that this model is not very realistic since it does not capture any correlation between consecutive packet arrival, which is, on the other hand, exhibited by experimental data [15, 16].

Consequently, to make the analysis of the MAC protocols as realistic as possible, in this article not only Poisson arrivals



■ **Figure 8.** *The principle of identifier splitting.*

are considered. Specifically, we assume that most of the traffic in future wireless and wired networks will resemble today's Web traffic. Hence we modeled data traffic by ON/OFF sources [17] where the ON and OFF durations are distributed according to the Weibull distribution. Furthermore, packet arrivals in the ON periods are generated according to an exponential distribution.

The Weibull distribution of a random variable X is

$$P\{X \le x\} = F(x) = 1 - e^{-\left(\frac{x}{\beta}\right)^{\alpha}}, \forall x \ge 0$$

where $\alpha > 0$ and $\beta > 0$ are real, and are called *shape* and *scale* parameters, respectively. As the α value decreases the Weibull distribution becomes more and more heavy tailed. When $\alpha=1$ the Weibull distribution is reduced to the exponential distribution. Since the packet arrivals during the ON periods are exponentially distributed this implies that when $\alpha=1$ the ON/OFF process is reduced to a two-state Markov modulated Poisson process (MMPP).[6]

Voice Traffic

A voice source has often been modeled in the literature via an ON/OFF process where the durations of the ON and OFF periods are exponentially distributed, and voice packets during the ON period are generated periodically [18]. This model can be used when the silence suppression is operated for bandwidth saving. However, when this is not the case, a CBR model is used. According to this model voice packets (of a fixed size) are generated periodically.

In this article we consider an ON/OFF model in the analysis of IEEE 802.11 and a CBR model in the analysis of ETSI HIPERLAN Type 1 and DSA++ protocols.

MPEG Video Traffic

The International Organization for Standardization (ISO) Moving Picture Experts Group (MPEG) algorithm is currently the most important compression algorithm used in all types of video applications. MPEG compression is done by reducing both the spatial and temporal redundancy of the video stream. This is achieved by using three different types of frame: I, P, and B frames, each coded in a different way. Typically, I frames require more bits than P frames, while B frames have the lowest bandwidth requirement. These frames are then arranged in a deterministic sequence, such as IBBPBBPBBPBB, which is called a group of pictures (GoP). The number of frames in a GoP is referred to as the *GoP size* [19].

To make the analysis as realistic as possible in this article we decided to make use of real traces of MPEG-1 traffic available at an FTP site [20, 21]. The trace used is coded with a GoP size equal to 12 and is made up of 40,000 frames.

H.263 Video Traffic

Bit rates generated by an MPEG video source are on the order of 1 Mb/s. Hence, transmission of MPEG video is feasible only with the HIPERLAN protocol. For IEEE 802.11 we considered low-bit-rate video conforming to the H.263 standard [22]. Although designed for fixed networks (one key application will be video telephony on normal analog telephone lines), H.263 will possibly be enhanced for use on third-generation mobile networks.

Since we did not have any H.263 video traces available long enough to allow a meaningful simulation experiment, in our analysis we represented the behavior of an H.263 video source by using the discrete auto-regressive DAR (1) model [23]. This is a discrete-time Markov chain with N states and transition probability matrix P given by $P = \rho I + (1 - \rho)Q$ where ρ is the auto-correlation coefficient, I is the identity matrix, and each row of Q consists of the negative-binomial probabilities $(f_0, f_1, \ldots, f_K, F_K^c)$ where

$$F_K^c = \sum_{k>K} f_k$$

and K is the peak rate in cells per video frame.

All the parameters of the DAR (1) model were estimated through a fitting procedure by using a real video sequence coded according to the H.263 standard, and available at a server [24].

Performance Measures

This section reports the performance measures commonly used to characterize the QoS provided by wireless MAC protocols. These performance measures are differentiated according to whether the type of traffic generated by wireless stations is non-real-time (data traffic) or real-time (voice and video).

Non-Real-Time Traffic

- *Average queuing delay:* average time elapsed from the time a packet joins the local queue[7] until it reaches the head of the local queue itself (i.e., starts contending for the channel)
- *Average MAC delay:* the average delay experienced by a packet from the time it reaches the head of the local queue until the beginning of its successful transmission
- *Average access delay:* the sum of the average MAC delay and average queuing delay
- *Aggregate throughput:* the average number of bits successfully transmitted by all stations per time unit

Real-Time Traffic

- *Cell loss rate (CLR):* the fraction of discarded cells
- *Percentiles* of the delay distribution

[6] *More precisely, the ON/OFF process reduces to an interrupt Poisson process (IPP).*

[7] *Packets are queued in a station local queue as soon as they arrive (i.e., they are generated). Therefore, except when the station is empty, a packet will experience some delay in the local queue before contending for the channel.*

IEEE 802.11 Performance Evaluation Results

Results related to data and video types of traffic are shown in the following. The former type of traffic is managed by the DCF, the latter by the PCF. The performance analysis of IEEE 802.11 is further extended toward the operating region envisaged for wireless ATM (i.e., at least 20 Mb/s) to verify whether or not this MAC protocol is a suitable candidate for wireless ATM.

System parameter	Parameter value (μs)
Slot time	50
SIFS	28
DIFS	128 (SIFS+2Slot-time)
Medium capacity	1 Mb/s

■ **Table 2.** *System parameter values.*

Traffic parameter	Parameter value
Station offered load	30 kb/s
Average ON duration	3.3 s
Average OFF duration	22.8 s

■ **Table 3.** *Traffic parameter values.*

Distributed Coordination Function

The simulation results outlined in this subsection are taken from [25]. System and traffic parameters are reported in Tables 2 and 3, respectively. The performance results are presented by loading IEEE 802.11 with several arrival processes by first assuming that the RTS/CTS is disabled and then enabled.

Influence of the Arrival Process with RTS/CTS Disabled – Figure 9 reports the aggregate throughput achievable by the DCF protocol vs. the number of stations when they are fed with Poisson and ON/OFF processes with several α values. The figure highlights that the throughput curves which refer to the various arrival processes are very close to each other. Since the medium capacity is equal to 1 Mb/s (Table 2), Fig. 9 shows that the protocol capacity achievable by IEEE 802.11 in the scenario defined by Table 3 is approximately 50 percent.

Furthermore, by varying the interarrival packet time distribution during ON periods (Weibull and constant distributions), the results obtained were practically the same as those reported in Fig. 9.

Figure 10 reports the average access delay vs. the number of data WTs, while Fig. 11 reports the average MAC delay and average queuing delay (i.e., the components of the average access delay) vs. the number of stations for three different values of α. Figure 10 shows that the more the ON and OFF duration distributions deviate from the exponential distribution, the worse the average access delay. Figure 11 highlights that the average MAC delay remains constant in the range of α values considered. This means that the collision avoidance mechanism is not influenced by the burstiness of the arrival process. In contrast, the average queuing delay increases significantly when the α value decreases. This behavior is justified in [25].

Influence of the Arrival Process with RTS/CTS Enabled – This protocol mechanism, designed to alleviate the problem of hidden stations, significantly influences the performance measures for any RTS threshold value. However, the improvement in performance is highly sensitive to the RTS/CTS threshold value and packet length distribution. For example, for the packet size distribution considered in [25],[8] results obtained suggest that an RTS threshold equal to zero should be used (i.e., to enable the RTS/CTS protocol mechanism for any packet length). With such a threshold, the utilization increases to approximately 75 percent.

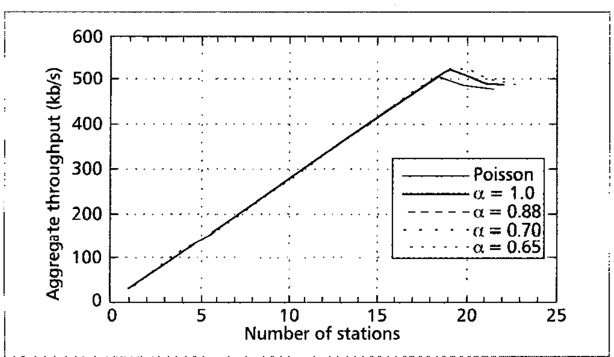
Suitability for Wireless ATM

In order to verify the suitability of IEEE 802.11 for wireless ATM, we first broadened our IEEE 802.11 MAC protocol analysis to verify whether or not this technology is suitable for efficiently managing the higher rates envisaged for wireless ATM. Figure 12 plots the IEEE 802.11 MAC protocol capacity (i.e., the maximum fraction of channel bandwidth used by successfully transmitted packets over all possible offered loads) for 1, 2, 5, and 10 Mb/s. As can be seen, the protocol capacity decreases when the channel speed increases.

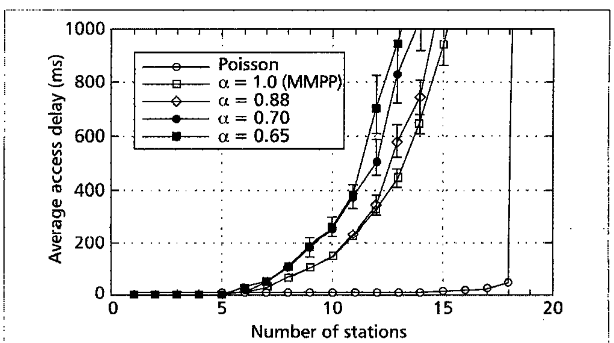
The inadequacy of the IEEE 802.11 MAC protocol to make effective use of the rates planned for wireless ATM might be a valid argument in itself to consider the protocol unsuitable for a wireless ATM interface. However, some other considerations can be made in order to support the result above.

As explained earlier, the PCF was introduced to manage real-time traffic. Since the overhead introduced by the protocol is very high (on the order of 75 bytes) this protocol is not adequate for managing packetized speech efficiently. In fact, to transmit 48 bytes of speech, PCF requires transmission of a packet of 48 (payload) plus 75 (overhead) bytes! Note that grouping several speech packets into the same PCF frame reduces the influence of the protocol overhead but introduces an additional delay. Hence, in practice, no more than two 48-

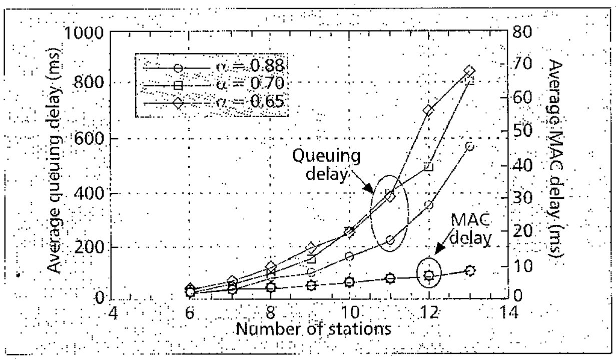
[8] *Bimodal distribution estimated via real traces taken from Bellcore.*



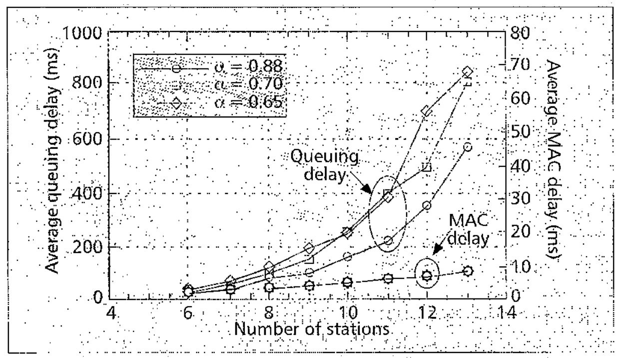
■ **Figure 9.** *Aggregate throughput vs. the number of data sources for different arrival processes.*



■ **Figure 10.** *Average access delay vs. the number of data sources for different arrival processes.*



■ **Figure 11.** *Average MAC delay and average queuing delay vs. the number of stations for several values of α.*



■ **Figure 12.** *Protocol capacity vs. different speed rates*

byte voice packets can be grouped into a PCF frame. Furthermore, since the protocol is able to manage a low number of voice connections at the QoS required, it cannot take full advantage of the statistical multiplexing of voice sources.

The situation is different for H.263 video frames since each frame is 420 bytes long. However, it has been shown [26] that the achievable QoS is very dependent on the bandwidth allocation scheme. In [26] a set of experiments is performed. Specifically, five video sources are considered which are all synchronized (frames arrives at the same instant) and transmit the same H.263 video sequence. To reduce the H.263 video model complexity it is assumed that each frame is made up of an integer number of 224-bit segments. Under this assumption the largest frame in the sequence used in the simulation experiments is 15 segments large.

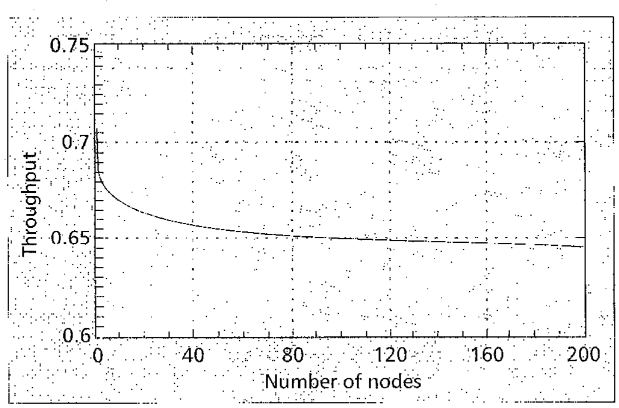
Initially each H.263 video source is allowed to transmit 15 segments for each CFP (peak allocation). Then the number of segments a video source can transmit during each CFP is reduced step by step.

The results obtained are summarized in Table 4 which reports the percentiles of the access delay distribution experienced by H.263 video frames for different bandwidth allocation schemes. As expected, the tail of the access delay distribution tends to increase as the allocated bandwidth decreases.

The PCF corresponds to a fixed allocation of bandwidth at least over some time. Therefore, PCF is not able to adjust the allocation dynamically to the capacity requirements of H.263. This results in a trade-off between bandwidth efficiency and access delay: the higher the efficiency, the higher the delay. It is not possible to get both, high efficiency and low access delay. In order to meet the delay requirements of H.263 a peak allocation is necessary which wastes a lot of available bandwidth.

Probability	15 segments	14 segments	13 segments	12 segments
0.9	2.6	2.6	2.6	2.6
0.99	3.7	3.7	43.3	123.3
0.999	4.0	122.8	281.5	481.9
0.9999	4.0	211.0	481.7	801.9

■ **Table 4.** *Percentiles in milliseconds of the access delay experienced by H.263 video frames*



■ **Figure 13.** *Throughput vs. the number of nodes.*

HIPERLAN Performance Evaluation Results

Some simulation results taken from [9] are outlined below. The operation parameter settings are reported in Table 5.

In [9] the authors prove that the probability values for the elimination (q) and yield (p) phases recommended by the standard are appropriate and that HIPERLAN is basically stable.

Furthermore, a closed formula for the HIPERLAN MAC capacity is derived. By means of this formula it is possible to realize that the aggregate user throughput (i.e., the throughput without the overhead introduced by control headers and control packets) which can be provided by the HIPERLAN MAC protocol is approximately 65 percent of the channel bandwidth, as shown in Fig. 13.

To evaluate the performance measures previously defined we carried out two sets of experiments: one with non-real-time traffic (data), the other with real-time traffic (video and voice).

Simulation Analysis for Non-Real-Time Data Traffic

The system parameters reported in Table 5 for the HIPERLAN simulation environment are those recommended by the standard [2]. Traffic-related parameters are reported in Table 6. The packet size was chosen to be constant and equal to 19,080 bits (i.e., the maximum length defined in the standard).

Parameter	Value
H	5
n	12
q	0.5
m	14
p	0.9

■ **Table 5.** *Operation parameter settings.*

Figure 14 reports the average MAC delay experienced by any packet in any station when the arrival process is Poisson and ON/OFF. In the latter case several α values are considered. Figure 14 shows that the average MAC delays of the various ON/OFF arrival processes almost overlap. On the other hand, the average MAC delay curve related to the Poissonian arrival process deviates considerably from the ON/OFF curves.

Traffic parameter	Parameter value
WT offered load	100 kb/s
Average ON duration	3.3 s
Average OFF duration	22.8 s
Packet lifetime	500 ms
User priority	Low priority

Table 6. *Traffic parameter values.*

The different behavior of the average access and MAC delays in the case of Poissonian arrivals with respect to the ON/OFF arrivals can be explained in terms of the burstiness of the arrival processes [9].

Figure 15 shows the average access delay versus the number of data stations for the arrival processes already considered in the analysis of the average MAC delay. The shape of the curves reported in Fig. 15 is similar to the shape of the curves related to the average MAC delay reported in Fig. 14. As expected, due to burstiness, the average access delay with ON/OFF processes is greater than with Poisson processes. Note that all the curves in Fig. 15 are below 500 ms, which is, in fact, the packet lifetime.

At first sight Fig. 15 might seem to show that, for the class of ON/OFF processes, the various forms of delay do not depend on the values of α (i.e., burstiness). However, in [9] the authors showed and justified that, by increasing the packet lifetime from 500 ms up to 10 s, the various curves reported in Fig. 15 increasingly diverge. They also proved that the average access delay is very much dependent on the channel access priority mechanisms.

To conclude, the HIPERLAN MAC protocol is a good protocol for data transmission. In fact, it is stable and its capacity is approximately 65 percent. However, performance is influenced by the packet lifetime and channel access priority mechanisms.

Suitability of HIPERLAN for Wireless ATM

Two scenarios have been defined in order to investigate the suitability of HIPERLAN for wireless ATM, one mixing low-bit-rate CBR (N-ISDN) with high-bit-rate ABR (Web) terminals, the other mixing high-bit-rate VBR (MPEG), again, with ABR (Web) terminals.

CBR (N-ISDN or Voice Traffic) – To check the suitability of HIPERLAN for managing different QoS requirements efficiently, we performed a simulation in which highly time-critical voice services (N-ISDN) and data services (Web browsing) were mixed. The N-ISDN service is modeled by a CBR source producing one ATM cell every 6 ms (packetizing delay for 64 kb/s). The data service is modeled by a traffic generator as described earlier with $\alpha = 1$.

Data services will always offer data to be transmitted, so with this scenario it is possible to check whether the QoS management of HIPERLAN is sufficient to prevent the time-sensitive services from delaying the data services with low QoS requirements.

The system parameters reported in Table 5 for the HIPERLAN simulation environment are those recommended by the standard [2]. Traffic-related parameters are reported in Table 7. The MSDU size was chosen to be one ATM cell (424 bits) for CBR, and between one and 45 ATM cells (424–19,080 bits, the maximum MSDU length defined in the standard) for ABR. Twenty Web stations offer the background load, and the number of CBR terminals varies from one to 60.

Figure 16 shows that with less than 14 N-ISDN terminals the cell loss ratio (CLR) is below 10^{-5}. With more than 14 N-ISDN terminals the CLR increases significantly. Assuming a tolerable CLR of 10^{-3}, only 20 N-ISDN terminals can be supported, although the sum data rate of these terminals is only 1.32 Mb/s (compared to the channel rate of 23 Mb/s).

In this scenario, since CBR traffic is high priority and ABR low priority, CBR traffic only is allowed to reach the highest channel access mechanism priority (i.e., CAM priority 0). Furthermore, since CBR traffic has an initial residual lifetime of 5 ms, it follows that each CBR MSDU starts contending immediately with CAM priority 0. As a consequence, in every channel access cycle any CBR MSDU will contend with the other CBR MSDUs only, since they all have a priority higher than that of any other ABR MSDU. In conclusion, the only interference produced by ABR traffic is a possible initial delay that CBR traffic has to experience in order to synchronize for the next channel access cycle when a previous ABR transmission is ongoing.

Real-Time VBR (MPEG-1 Video) – With this scenario the ability of HIPERLAN to handle rt-VBR traffic is checked. Here, MPEG-1 video stations are mixed with Web stations.

Because all MSDUs belonging to the same frame have the same initial lifetime of 40 ms, they start with CAM priority 2 and share this priority level with the Web stations. Thus, we examine whether it is possible to share CAM priority levels efficiently among different service categories.

The traffic-related parameters are reported in Table 8. Two levels for the VBR MSDU size have been investigated, 20 ATM cells (7680 bits) and 45 ATM cells (17,280 bits).

The number of VBR stations varies from 10 to 20, and the video sequences transmitted by each station are different and completely uncorrelated.

Figure 17 shows the CLR vs. the number of MPEG nodes and highlights that the performance of HIPERLAN Type 1

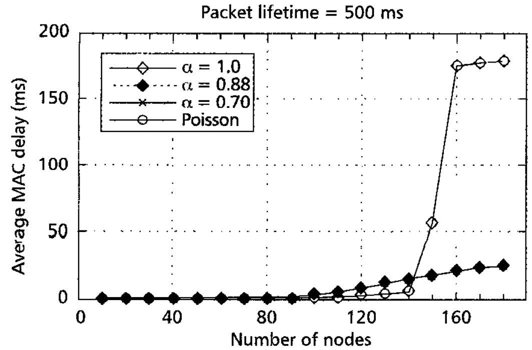


Figure 14. *Average MAC delay for several values of α and Poisson.*

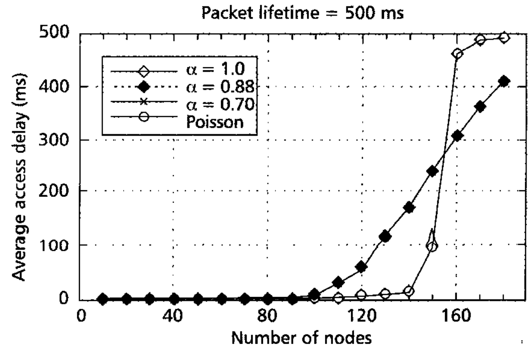
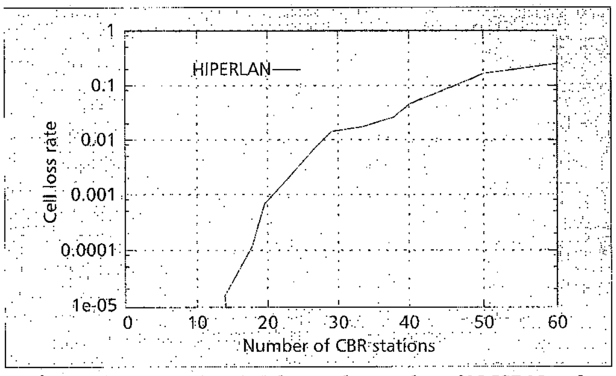
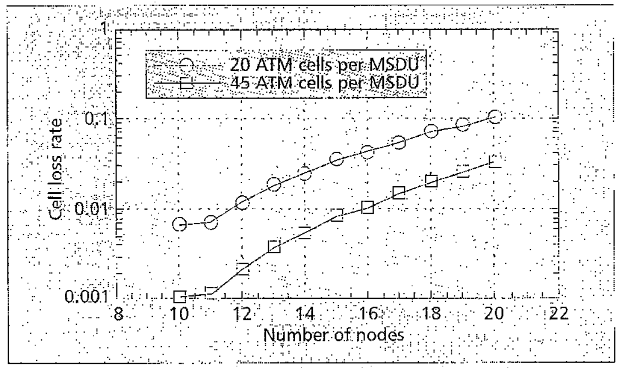


Figure 15. *Average access delay vs. the number of stations for several values of α and Piosson.*



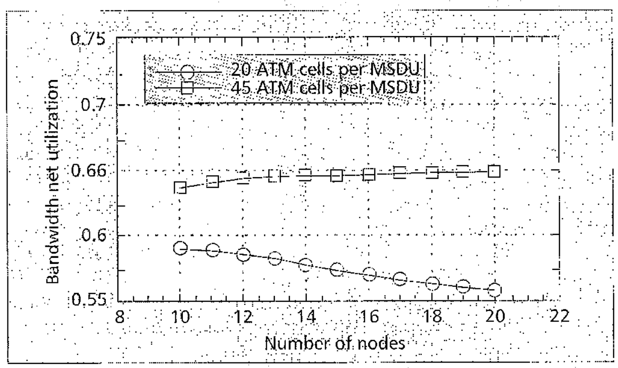
■ **Figure 16.** *Discard probability vs. the number of N-ISDN nodes.*



■ **Figure 17.** *Cell loss ratio vs. the number of MPEG-1 video nodes.*

Traffic parameter	N-ISDN station	Web station
WT offered load	64 kb s	1 Mb/s
Average ON/OFF duration	n/a	3.3 s/22.8 s
ATM service category	CBR	ABR
Packet lifetime	5 ms	500 ms
User priority	High priority	Low priority

■ **Table 7.** *Traffic parameter values for the CBR scenario.*



■ **Figure 18.** *Bandwidth utilization vs. the number of MPEG-1 video nodes.*

strongly depends on the size of the MSDU. Only with 45 ATM cells per MSDU and a tolerable CLR of around 10^{-3} can an MPEG-1 video station be handled in this scenario.

Figure 18, which shows the aggregate bandwidth utilization, confirms the analysis above. In fact, with 45 ATM cells/MSDU, the bandwidth utilization is nearly constant over the number of video terminals, while with 20 ATM cells/MSDU the utilization is lower, and decreases when the number of MPEG terminals increases. As a consequence, the CLR is higher in the latter case.

In summary, both scenarios show that HIPERLAN is able to handle different QoS requirements to some extent. The efficiency strongly depends on the length of the MSDUs: the longer the MSDUs, the better the performance.

DSA++ Protocol Performance Results

Since random access plays an important role in the DSA++ protocol, the time required to resolve a collision has a considerable influence on performance. The complementary distribution function of the delay to resolve a collision in the case of a binary identifier splitting algorithm is shown in Fig. 19. For comparison, the resulting delay in case of a coin-flip splitting algorithm (which uses an infinite number space) is also shown. The size of the identifier space specifies the number of terminals which have established a connection to the BS. Since collision resolution is performed from signaling period to signaling period, the delay of the collision resolution refers to the length of a period. For the analysis it was assumed that there is a binomial distributed number of terminals in a start set with a mean of 1.5.

It can be seen that the identifier splitting algorithm leads to a deterministic maximum delay to resolve a collision, and that a collision is resolved very fast even for a large number of terminals.

Traffic parameter	MPEG-1 video station	Web station
Station offered load	0.56 Mb/s	1 Mb/s
Average ON/OFF duration	n/a	3.3 s/22.8 s
ATM service category	rt-VBR	ABR
Packet lifetime	40 ms	500 ms
User priority	High priority	Low priority

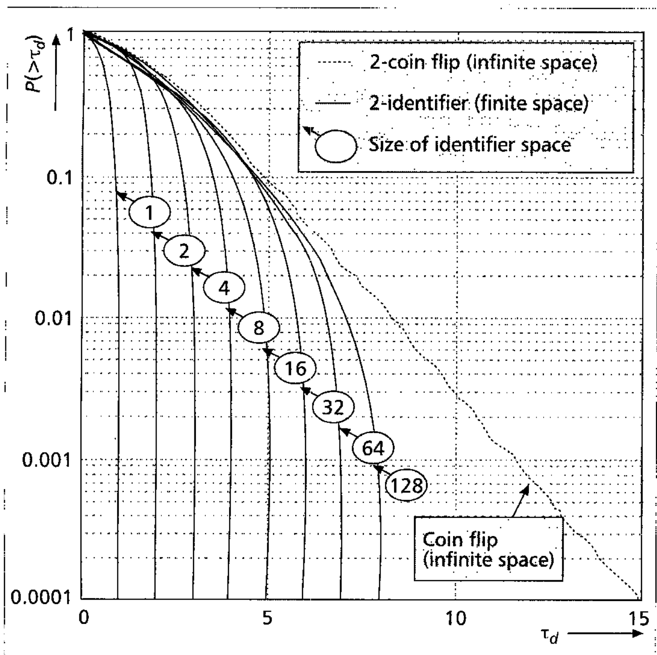
■ **Table 8.** *Traffic parameter values for the VBR scenario.*

Suitability for Wireless ATM

The same two scenarios as for HIPERLAN have been used to investigate the suitability of DSA++ for wireless ATM. ABR services are mixed with CBR and VBR services, respectively. The DSA++ specific simulation parameters are summarized in Table 9.

CBR (N-ISDN or Voice) Traffic – Simulations were carried out to evaluate the support of real-time-oriented CBR traffic under high system load. The simulation scenario consisted of one BS serving 20 ABR terminals and a varying number of CBR terminals. Each ABR terminal generated traffic with 1.25 Mb/s, so the available capacity of one physical channel (25 Mb/s) was completely used. Each CBR terminal obtained one CBR connection in each direction with 64 kb/s each. Table 10 summarizes the traffic parameters used in this scenario.

For the sake of simplicity, instead of exploiting the full DSA++ capabilities, in the analysis we used static priorities associated with each ATM service class (i.e., CBR > VBR > ABR > UBR) to support real-time services. The main result of our experiments is that where there was error-free transmission over the radio interface, none of the CBR ATM cells exceeded its maximum access delay, so no losses occurred.

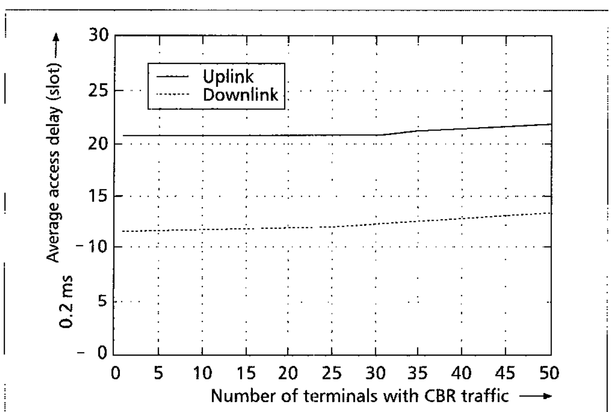


■ **Figure 19.** *Complementary distribution function of the time required to resolve a collision (second-order identifier splitting).*

In Fig. 20 the mean access delay experienced by any CBR terminals vs. the number of CBR terminals is shown. The figure highlights that the mean access delay is almost constant as it increases slightly with a higher number of CBR terminals, and the uplink delay is always higher than the downlink delay.

Therefore, the mean access delay of the CBR ATM cells is only influenced by the length of the signaling period and the other CBR services. Due to the overloaded system, the length of the signaling period is at its maximum and only limited by the available capacity inside the downlink signaling information. Since the interarrival time of the ATM cells of a CBR connection is deterministic, the BS is able to estimate the capacity requirements of the CBR terminals, and therefore is not required to transmit additional capacity requests via the uplink. This explains the negligible influence of the amount of CBR traffic on the access delay as long as the total capacity of all CBR connections is low compared to the capacity offered by one physical channel.

The difference between the uplink and downlink transmission results from the structure of the signaling period: the



■ **Figure 20.** *The influence of the number of CBR terminals on the mean access delay.*

Simulation parameter	DSA++ value
Channel speed	25 Mb/s
Maximum length of signaling period	20 slots = 400 μs
Normal slot length	20 μs
Short slot length	5 μs

■ **Table 9.** *Simulation parameters for DSA++.*

Traffic parameter	N-ISDN station	Web station
Station offered load	2*64 kb/s	1.25 Mb/s
Average ON/OFF duration	n/a	3.3 s/22.8 s
ATM service category	CBR	ABR
Packet lifetime	5 ms	500 ms

■ **Table 10.** *Traffic parameter values for the CBR scenario.*

uplink part is at the end of the signaling period, while the downlink part is at the beginning.

Real-Time VBR – With this scenario the ability of DSA++ to handle rt-VBR traffic is checked. Here, video stations are mixed with Web stations.

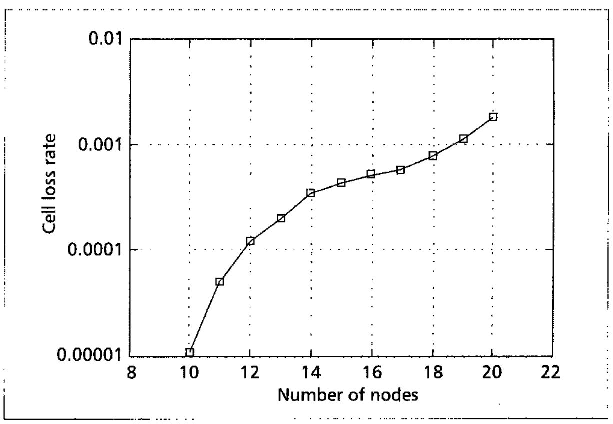
The traffic-related parameters are reported in Table 11.

Figure 21 shows the CLR of VBR ATM cells using the DSA++ protocol for a various number of video nodes. As for CBR services, the static priorities used by the scheduling algorithm lead to a small CLR for the VBR traffic even in the case of high overall loads. Assuming a tolerable CLR of around 10^{-3}, 18 video terminals can be handled, which is almost twice the maximum number of nodes HIPERLAN Type 1 can support.

Conclusions

This article analyzes, from a traffic performance point of view, three MAC protocols for wideband wireless local access: IEEE 802.11, ETSI HIPERLAN Type 1, and Dynamic Slot Assignment (DSA++).

The analysis shows that, although IEEE 802.11 manages asynchronous data traffic efficiently, it has serious (MAC) overhead problems in supporting voice, while handling of H.263 video is not recommended unless peak bandwidth allocation is performed during the contention-free periods. Fur-



■ **Figure 21.** *Cell loss rate vs. the number of video nodes.*

Traffic parameter	Video station	Web station
Station offered load	0.56 Mb/s	1 Mb/s
Average ON/OFF duration	n/a	3.3 s/22.8 s
ATM service category	rt-VBR	ABR
Packet lifetime	40 ms	500 ms

■ **Table 11.** *Traffic parameter values for the VBR scenario.*

thermore, increasing channel bit rate toward the operating region envisaged for wireless ATM (at least 20 Mb/s) showed that the performance of the IEEE 802.11 MAC protocol degrades to values which are not tolerable.

Regarding ETSI HIPERLAN Type 1, the analysis showed that, similar to IEEE 802.11, it is efficient in managing computer data traffic, while handling of voice is very inefficient due to the overhead introduced by the MAC protocol. Video can be handled efficiently by carefully selecting a scheduling discipline. However, performance results are sensitive to the distribution of packet lifetime. Unlike IEEE 802.11, the channel bit rate provided by the HIPERLAN standard is in the range envisaged for the forthcoming wireless ATM.

In both cases, the main reason for inefficiency in the transmission of voice traffic is the heavy overhead introduced by the MAC protocols. This overhead is due in part to the addressing scheme which, for compatibility with existing wired corporate LANs, spans over 48 bits.

This problem does not seem to affect the DSA++ protocol, which was shown to manage several service categories, even voice, in an efficient manner. DSA++ was explicitly designed to support ATM cell transfer over the radio interface in such a way that protocols of the ATM adaptation layer are not involved. As a consequence, the MAC protocol provides seamless interworking with the wired counterpart of the ATM network.

In conclusion, the analysis suggests that, although both IEEE 802.11 and ETSI HIPERLAN MAC protocols have some interesting features in supporting the integration of real-time and non-real-time traffic, their performance does not come up to that envisaged for the wireless ATM MAC protocol. On the other hand, an integrated approach such as that provided by DSA++, based on transparent transmission of packets between the wireless and wired networks, enables us to fully meet the wireless ATM requirements.

References

[1] IEEE 802.11D3, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," Jan. 1996.

[2] ETSI TC-RES, "Radio Equipment and Systems (RES); HIgh PERformance Radio Local Area Networks (HIPERLAN); Type 1; Functional Specification, Technical Correction," Dec. 1996.

[3] K. Pahlavan, A. Zahedi, and P. Krishnamurty, "Wideband Local Access: Wireless LAN and Wireless ATM," *IEEE Commun. Mag.*, Nov. 1997, pp. 34–40.

[4] *IEEE Pers. Commun.*, Special Issue on Wireless ATM, vol. 3, no. 4, Aug. 1996.

[5] ETSI, "Requirements and Architectures for Wireless ATM Access and Interconnection," TR 101 031, Apr. 1997.

[6] O. Kubbar, and H. T. Mouftah, "Multiple Access Control Protocols for Wireless ATM: Problems Definition and Design Objectives," *IEEE Commun. Mag.*, Nov. 1997, pp. 93–99.

[7] D. Petras and A. Krämling, "Wireless ATM: Performance Evaluation of a DSA++ MAC Protocol with Fast Collision Resolution by a Probing Algorithm," *Int'l. J. Wireless Info. Networks*, vol. 4, no. 4, 1997.

[8] ATM Forum, "Traffic Management Specification," v. 4.0, af-tm-0056.000, Apr 1996.

[9] G. Anastasi, L. Lenzini, and E. Mingozzi, "Stability and Performance Analysis of HIPERLAN," *Proc. IEEE INFOCOM '98*, San Francisco, CA, Mar.–Apr. 1998.

[10] F. Bauchot *et al.*, "MASCARA, a MAC Protocol for Wireless ATM," *ACTS Mobile Summit*, Granada, Spain, Nov. 1996, pp. 647–51.

[11] D. Bertsekas and R. Gallager, *Data Networks*, Prentice Hall, 1992.

[12] D. Petras and A. Krämling, "Fast Collision Resolution in Wireless ATM Networks," *Proc. 2nd MATHMOD '97*, Vienna, Austria, Feb. 1997, pp. 517–24.

[13] H. Kist and D. Petras: "Service Strategy for VBR Services at an ATM Air Interface," *2nd Euro. Pers. Mobile Commun. Conf.*, Bonn, Germany, Oct. 1997, pp. 181–88.

[14] H. Zhang, "Service Disciplines for Guaranteed Performance Service in Packet-Switching Networks," *Proc. IEEE*, vol. 83, no. 10, Oct. 1995, pp. 1374–96.

[15] R. Gusella, "Characterizing the Variability of Arrival Processes with Indexes of Dispersion," *IEEE JSAC*, vol. 9, no. 2, Feb. 1991, pp. 203–11.

[16] W. E. Leland, M. S. Taqqu, W. Willinger, D. V. Wilson, "On the Self-Similar Nature of Ethernet Traffic," *IEEE/ACM Trans. Networking*, vol. 2, no 1, Feb. 1994, pp. 1–15.

[17] S. Deng, "Empirical Model of WWW Document Arrivals at Access Link," *Proc. ICC '96*.

[18] P. T. Brady, "A Model for Generating ON-OFF Speech Patterns in Two-way Conversation," *Bell. Sys. Tech. J.*, Sept. 1969, pp. 2445–72.

[19] D. Le Gall, "MPEG: A Video Compression Standard for Multimedia Applications," *Commun. ACM*, vol. 34, no. 4, Apr. 1991, pp. 46–58.

[20] ftp://ftp-info3.informatik.uni-wuerzburg.de/pub/MPEG/traces

[21] O. Rose, "Statistical properties of MPEG video traffic and their impact on traffic modelling in ATM systems," *Proc. 20th Annual Conf. Local Comp. Networks*, Minneapolis, CA, Oct. 1995.

[22] ITU-T Draft Rec. H.263, "Video Coding for Low Bitrate Communication," Dec. 5, 1995.

[23] D. P. Heyman, A. Tabatabai, and T. V. Lakshman, "Statistical Analysis and Simulation Study of Video Teleconference Traffic in ATM Networks," *IEEE Trans. Circuits and Sys. for Video Tech.*, vol. 2, no. 1, Mar. 1992.

[24] ftp://bonde.nta.no/pub/tmn/qcifsource/

[25] G. Anastasi, E. De Stefano, and L. Lenzini, "QoS Provided by the IEEE 802.11 Wireless LAN to Advanced Data Applications: a Simulation Analysis," Wksp. Nomadic Comp., Geneva, Switzerland, Apr. 5, 1997.

[26] L. Bonuccelli, "Service Integration in a Wireless LAN IEEE 802.11," Laurea thesis, Univ. of Pisa, 1997 (in Italian).

Biographies

GIUSEPPE ANASTASI (anastasi@iet.unipi.it) received a Laurea degree in electronic engineering and a Ph.D. degree in computer engineering, both from the University of Pisa, Italy, in 1990 and 1995, respectively. In 1991 he joined the Department of Information Engineering of the University of Pisa where he is presently assistant professor of computer engineering. His main fields of interest include mobile computing and networking, distributed computing systems, and modeling and performance evaluation of computer networks. He is a member of the IEEE Computer Society.

LUCIANO LENZINI (lenzini@iet.unipi.it) holds a degree in physics from the University of Pisa, Italy. He joined CNUCE, an institute of the Italian National Research Council (CNR) in 1970. Starting in 1973, he spent a year and a half at the IBM Scientific Center, Cambridge, Massachusetts, working on computer networks. He has since directed several national and international projects, including RPCNET, the first Italian packet switching network; STELLA, the first European broadcasting satellite network; and OSIRIDE, the first Italian OSI network. His current research interests include integrated service networks, and the design and performance evaluation of both metropolitan area network MAC protocols and packet-based radio access mechanisms for third-generation mobile systems.

ENZO MINGOZZI (mingozzi@iet.unipi.it) is a Ph.D. student of computer engineering at the University of Pisa, Italy. He received a Laurea degree in computer systems engineering from the University of Pisa, magna cum laude, carrying out a thesis about the design and performance evaluation of a MAC protocol for third-generation mobile systems. His current research interests focus on the design and analysis of MAC protocols and service integration in wireless LANs.

ANDREAS HETTICH (ahe@comnets.rwth-aachen.de) received a diploma degree in electrical engineering in 1996 from Aachen University of Technology, Germany. Currently he is working toward a Ph.D. degree at Aachen University of Technology. In 1996 he joined the Chair for Communication Networks as a research assistant. His current research interests include broadband radio networks, multiple access techniques, and error control mechanisms. He is involved in the standardization of Broadband Radio Access Networks (BRAN) at ETSI.

ANDREAS KRÄMLING (akr@comnets.rwth-aachen.de) received his diploma degree in electrical engineering in 1996 from Aachen University of Technology, Germany. During his study he has been a student worker at the Institute of Technical Electronics, where he worked in the area of multispectral color analysis. In 1996 he joined the Department of Communication Networks, Aachen University of Technology, as a research assistant. His research interests are in the field of mobile communication protocols, especially wireless ATM, fast collision resolution algorithm, and dynamic channel allocation for wireless ATM networks.