Assured and Expedited Forwarding Extensions for IEEE 802.11 Wireless LAN

Albert Banchs[a], Markus Radimirsch[b], Xavier Pérez[a]
[a] NEC Europe Ltd., Network Laboratories Heidelberg, Germany
[b] Institut für Allgemeine Nachrichtentechnik, University of Hannover, Germany

***Abstract*—In this paper, we propose DIME (DIffServ MAC Extension), an extension of the IEEE 802.11 MAC protocol to support Differentiated Services. The proposed extension consists of two optional modules: the Expedited Forwarding (EF) and the Assured Forwarding (AF). The Expedited Forwarding extension (DIME-EF) reuses the Interframe space of the Point Coordination Function (PCF) of the IEEE 802.11 standard in a distributed manner, while the Assured Forwarding extension (DIME-AF) relies on the Distributed Coordination Function (DCF) with a modified algorithm for the computation of the Contention Window (CW). Best Effort is supported by the functionality of the current 802.11 standard in such a way that legacy IEEE 802.11 terminals behave as Best Effort terminals in the DIME architecture. While the performance of the Assured Forwarding extension has been thoroughly evaluated by the authors elsewhere [1], this paper concentrates on the overall architecture and the performance of the Expedited Forwarding extension.**

***Index Terms*— Wireless LAN, Differentiated Services, Assured Forwarding, Expedited Forwarding, Quality of Service, MAC, IEEE 802.11**

I. INTRODUCTION

One of the biggest challenges in today's computer networks is to provide the Quality of Service (QoS) appropriate for the constantly growing demand from the side of applications. Over the last ten years, considerable effort has been made to provide QoS to the Internet, with proposals such as Integrated Services [2] and Differentiated Services [3]. Both of these architectures use queuing mechanisms which schedule and drop packets according to their delay priority and bandwidth assurance.

QoS mechanisms are of particular relevance in the case of Wireless LAN, where the bandwidth is scarce and the efficient use of it is of special importance. Frequency is a scarce resource and, due to the propagation characteristics of the radio channel, is a shared medium for those using it.

Since Wireless LANs may be considered as just another technology in the communications path, it is desirable that the architecture for QoS support follows the same principles in the wireless network as in the wireline Internet, assuring compatibility among the wireless and the wireline parts. The Differentiated Services (DiffServ) architecture for the wireline Internet aims at providing simple and scalable service differentiation by discriminating and treating the data flows according to their service class [3]. DiffServ makes a trade-off: QoS for individual packets is not necessarily guaranteed, but the DiffServ architecture scales well and is easy to implement. Because of these reasons, DiffServ is an increasingly popular approach for providing QoS in the Internet.

DiffServ standardization is currently an ongoing effort. Up to date, two Per-Hop Behaviors (PHBs) have been standardized: the Expedited Forwarding PHB [4] and the Assured Forwarding PHB [5], and several Per-Domain Behaviors (PDBs) have been proposed for standardization: the Virtual Wire PDB [6], the Bulk Handling PDB [7] and the Assured Rate PDB [8].

This paper proposes a DiffServ Extension for the MAC layer of the IEEE 802.11 standard (DIME: DIffServ MAC Extension). The proposed extension consists of two parts: *1)* the Expedited Forwarding extension, in line with the Virtual Wire PDB, which guarantees to its user low delays and delay variations within a given throughput, and *2)* the Assured Forwarding extension, in line with the Assured Rate PDB, which guarantees a specific throughput. In addition to these two extensions, the architecture we propose supports Best Effort (Bulk Handling PDB) as the default service.

The rest of the paper is organized as follows. We first introduce the state of the art; we recall the basics of the IEEE 802.11 standard and present a review on related work. Then, in Section III we explain the DIME architecture and the three service classes it supports (Expedited Forwarding, Assured Forwarding and Best Effort). The details of the algorithms used in the DIME architecture for the support of Expedited Forwarding and Assured Forwarding, DIME-EF and DIME-AF, are described in detail in Sections IV and V. In Section VI we present our simulations results and, finally, the conclusions section closes the paper.

II. STATE OF THE ART

A. The IEEE 802.11 MAC layer

The basic IEEE 802.11 Medium Access mechanism is called Distributed Coordination Function (DCF) and is based on the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol. In the DCF mode, a station must sense the medium before initiating the transmission of a packet. If the medium is sensed idle for a time interval greater than the DCF Inter Frame Space (DIFS), then the station transmits the packet. Otherwise, the transmission is deferred and a backoff process is started.

Specifically, the station computes the backoff interval as an equally distributed random value taken from the range of 0 to the so-called Contention Window (CW), where the backoff time is measured in slot times[1]. This backoff interval is then used to initialize the backoff timer. This timer is decreased only when the medium is idle and is frozen when it is sensed busy. Each time the medium becomes idle for a period longer than a DIFS, the backoff timer is periodically decremented, once every slot-time.

---

[1] Note that the 802.11 MAC protocol is slotted, i.e. the access to the channel can happen only at specific instants.

As soon as the backoff timer expires, the station starts to transmit. A collision occurs when two or more stations start transmission simultaneously in the same slot. To avoid collisions, a Request To Send (RTS) and a clear to send (CTS) can be exchanged between source and receiving station prior to the actual frame transmission. In addition, an Acknowledgement (Ack) is transmitted to the source after successful reception of the frame to detect collisions. The Ack scheme can additionally be used to control the retransmission of erroneous frames. The RTS/CTS scheme is also used for hidden node handling.

If a CTS or acknowledgment is not received by the source station, it assumes that the transmission attempt was not successful and re-enters the backoff process. To reduce the probability of collisions, the CW is doubled after each unsuccessful transmission attempt until a predefined maximum (CW_{max}) is reached. After a successful frame transmission, if the station still has frames buffered for transmission, it must execute a new backoff process.

The second access mechanism specified in the IEEE standard is built on top of DCF and it is called Point Coordination Function (PCF). It is a centralized mechanism, where one central coordinator polls stations and allows them undisturbed, contention free access to the channel. With this mechanism, collisions do not occur since the access is controlled by the coordinator, which receives a prioritized access to the channel by using a shorter inter frame space (the PIFS: PCF Inter Frame Space).

In addition to the DIFS and the PIFS, the IEEE 802.11 standard defines a third inter frame space: the SIFS (Short Inter Frame Space). The SIFS is the shortest inter frame space, and, as a consequence, corresponds to the highest access priority. After a SIFS, only acknowledgements, CTS and data frames in response to poll by the PCF may be sent.

B. Related work

Current trends in wireless networks indicate a desire to provide a flexible wireless infrastructure that can support emerging multimedia services along with traditional data services. In such a multiservice wireless environment, QoS support becomes critical.

One possible approach for supporting QoS in Wireless LAN is based on the Integrated Services architecture proposed for the wireline Internet [9]. In this approach, the control over wireless resources is very strict, motivated by the argument that strict control, with complex and sophisticated mechanisms and protocols, is required to maintain good quality in the wireless environment.

Another approach for QoS support in Wireless LAN is based on the Differentiated Services architecture, which provides service differentiation using more simple mechanisms. There have been several proposals for service differentiation in wireless networks, like in [10]. These mechanisms, however, rely on centralized control and polling of backlogged mobile hosts. In contrast to these proposals, the architecture we propose is based on distributed control. We argue that distributed control results in a more productive use of radio resources.

[11], [12], [13], [14] and [15] are other proposals for service differentiation relying on distributed control. These architectures are based on the idea of modifying the backoff time computation of the 802.11 standard to provide service differentiation, which is also the basis of our Assured Forwarding extension (DIME-AF).

In [11] the backoff time computation is modified by assigning shorter CWs to low delay real-time service. The main difference between [11] and DIME is that [11] does not decouple real-time (EF) traffic from data (AF and BE) traffic and, as a consequence, the service quality of real-time traffic in [11] is sensitive to the changing conditions of data traffic. In addition, [11] does not provide different priorities for data traffic.

[12] and [13] propose the use of different CWs and different backoff increase parameters, respectively, for different priorities in data traffic. The fact that the parameters in [12] and [13] are statically set makes the throughput received by a high quality station uncertain, as opposed to DIME-AF, in which the desired throughput is achieved by modifying dynamically the CW.

The Distributed Fair Scheduling (DFS) approach [14] proposes a dynamic algorithm for the backoff time computation in order to allocate bandwidth to the different stations proportionally to their weights. The main difference between the service provided by DFS and DIME-AF is that DFS provides relative throughput guarantees, while DIME-AF aims at providing absolute guarantees. One drawback of DFS as compared to DIME-AF is that in DFS each node has to monitor all transmitted packets and read the so-called finish tag of each packet. In addition, DFS requires the header format of 802.11 to be modified in order to include this finish tag in the packet header.

[15] provides relative priorities for delay and throughput in a multi-hop wireless network. This approach piggybacks scheduling information onto RTS/DATA packets and then uses this information to modify the computation of the backoff times. [15] has the same drawbacks commented for DFS, since it requires all nodes to monitor all transmitted packets in order to extract the scheduling information, and it requires the modification of the 802.11 header formats. Another drawback of [15] is that it does not provide backwards compatibility.

The Black Burst scheme in [16] introduces a distributed solution to support real-time sources over 802.11, by modifying the MAC for real-time sources to send short transmissions to gain priority. This method can offer bounded delay. The disadvantage of [16] is that it is optimized for isochronous sources, preferably with equal data rates, which can be a significant limitation for applications with variable data rates.

III. THE DIME ARCHITECTURE

In this section we introduce the basic design of the DIME architecture to provide Expedited Forwarding, Assured Forwarding and Best Effort functionality. The architecture details are further explained in the following sections.

A. Expedited Forwarding

In order to satisfy the requirement of Expedited Forwarding (EF) for low delay, EF packets should be given a prioritized access to the channel. The only solution in the current 802.11 MAC protocol that allows prioritized handling is the PCF mode by using a shorter inter frame space (the PIFS). Note that the PCF mode is currently not supported in most wireless cards, and it was shown in [17] that the cooperation between PCF and DCF modes leads to poor throughput performance.

Following the above considerations, Expedited Forwarding in DIME is supported by reusing the PIFS of the current standard. The only requirement is that the PCF as defined by 802.11 must not be used together with our proposal in the same Basic Service Set. This can easily be achieved because it is the Access Point (AP) which decides on the use of either method. If the original PCF is being used by the AP, all stations within reach must not use the DIME-EF MAC scheme proposed here. In IEEE 802.11 ad hoc networks, the PCF can not occur and, hence, DIME-EF can be used as required.

Reusing the PIFS solves the contention between EF packets and packets belonging to other service classes by giving a higher priority to the former. However, different stations with EF traffic may still collide when trying to access the channel after the PIFS. For this reason, a contention resolution algorithm is needed for EF stations. This algorithm is explained in detail in Section IV.

B. Assured Forwarding and Best Effort

The idea of the Assured Forwarding (AF) extension is to provide a user with a guaranteed throughput. In the DCF approach, the throughput received by a station depends on its CW: the smaller the CW, the higher the throughput. In DIME, Assured Forwarding is supported by the DCF function of the current standard with minor changes in the computation of the CW in order to give to each station the desired throughput. In Section V we present in detail the algorithm for computing the CW for DIME-AF.

Best Effort (BE) in DIME is also supported by means the DCF function. The CWs used by BE are the ones calculated according to the current IEEE 802.11 standard. With this CWs, 802.11 terminals behave as Best Effort terminals in the DIME architecture, providing thus backward compatibility.

According to the above explanation, BE and AF packets use the same inter frame space (the DIFS) but compete with each other with different CWs. The CW of Best Effort traffic cannot be arbitrarily increased for backward compatibility reasons. Also, the CW of AF traffic cannot be arbitrarily decreased, since this would lead to an unstable situation with permanent collisions. These limitations in the CWs makes it impossible to totally control the capacity given to each terminal. Therefore a certain level of impact of Best Effort to AF is unavoidable. This impact has been considered in the simulations of AF (see Section VI-B).

C. Admission Control

Our approach requires admission control for both EF and AF in order to ensure that the sum of the throughputs committed to EF and AF is not larger than the total throughput available in the Wireless LAN. Additionally, for EF, admission control is also required to limit the amount of EF traffic to some percentage of the channel capacity, and thus keep the number of EF packets waiting to be served (and consequently the delay experienced by them) small. In Section VI we propose a rule of thumb admission control based on simulation results for EF.

D. Protocol Operation

The combination of the Expedited Forwarding (EF), Assured Forwarding (AF) and Best Effort (BE) mechanisms we propose in the DIME architecture leads to the protocol operation shown in the example of Figure 1. In this example, after the end of the previous transmission, some station has an EF packet to transmit and it accesses the channel at the end of the PIFS. After the end of the transmission, the receiver answers with an acknowledgement after a SIFS.

In the next access cycle, there is no EF traffic to be transmitted, so the channel can be accessed by AF and BE (AF competing with a smaller CW). In the example, it is an AF packet that accesses first the channel. The last packet is finally a BE packet, which uses a CW calculated according to the IEEE 802.11 standard.

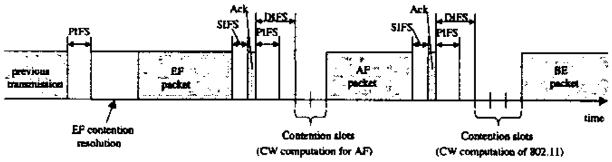

Fig. 1. Protocol Operation.

IV. CONTENTION RESOLUTION ALGORITHM FOR DIME-EF

The principle of the DIME-EF contention resolution scheme is shown in Figure 2. This scheme uses principles of the EY-NPMA MAC protocol described in [18], and, according to [19], has a residual collision rate almost independent from the number of contending stations. However, the parameters of the contention resolution (CR) scheme as used in [18] will be adapted to the requirements of the EF Scheme.

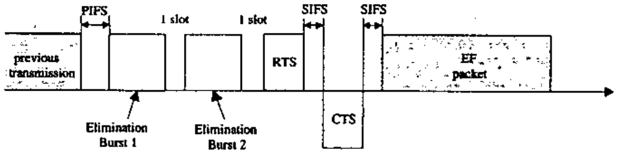

Fig. 2. Contention resolution scheme for EF traffic.

A station with EF traffic starts its contention cycle when a PIFS has passed after the end of a previous transmission. DIME-EF uses two bursts for elimination, elimination burst (EB) 1 and EB2. These bursts may consist of a random data or pseudo-noise pattern. Their only purpose is to occupy the channel such that stations not using the DIME-EF cannot sense the channel idle for longer than a PIFS duration and, hence, do not interfere a DIME-EF access attempt.

The duration of the EBs are multiples of the Slot Duration defined in the 802.11 standard. The duration of EB1 is calculated according to the following probability density:

$$P_{E1}(n) = \begin{cases} p_{E1}^{n-1}(1 - p_{E1}) & ; 1 \leq n < m_{E1} \\ p_{E1}^{m_{E1}-1} & ; n = m_{E1}, \end{cases} \quad (1)$$

where n is the number of slot durations EB1 shall last, p_{E1} is a probability parameter between 0 and 1 and m_{E1} is the maximum number of EB1 slots. Note that the above formula requires that EB1 lasts at least one slot. This is necessary in order to occupy the channel and keep terminals not using EF from making an access attempt.

The duration of EB2 shall be calculated according to the probability density

$$P_{E2}(n) = \frac{1}{m_{E2}} \quad \text{for } 1 \le n \le m_{E2}, \tag{2}$$

i.e. it is taken from an equally distributed variable in the range between 1 and the maximum number of EB2 slots, m_{E2}. Note that here the duration is at least one slot for the same reasons as for EB1.

A station that makes an access attempt, first chooses the duration of EB1 and EB2. If it senses the channel free for at least a PIFS, it transmits its EB1. After this transmission, the station senses the channel for one slot duration. If the channel is sensed free, it continues to send its EB2 after the sensing slot. After the transmission of EB2, it senses the channel again. If it is free, it starts to transmit its RTS after a slot duration and the transmission continues as defined for the data transmission using the DCF. If, however, the station senses the channel busy after its transmission of EB1 or EB2, it withdraws its transmission attempt and defers until the channel has been free for at least a PIFS. Using this mechanism, the station which chooses the longest EB1 and EB2 among all contending stations wins the contention and is allowed to transmit.

If two stations happen to have the same EB1 and EB2 durations, they collide. However, due to the importance of the packets, we use the already defined mechanisms in 802.11 for collision detection, i.e. the RTS/CTS handshake and the transmission of an Ack after the packet reception. In fact, the Ack will be transmitted in any case if a packet is being transmitted from a station using the new scheme to a station using the old scheme and, hence, shall be kept for the sake of backwards compatibility.

The DIME-EF scheme will be analyzed mathematically in the following clause. It will be shown that the performance of the scheme depends on the values of p_{E1}, m_{E1} and m_{E2}, but also on the number of stations entering a contention cycle. The goal of the mathematical analysis is to parameterize the DIME-EF scheme appropriately for the application in the IEEE 802.11 extension. It will also be a basis to justify the selection of this specific scheme, even if there might have been other possible candidates.

A. Mathematical Analysis

The idea for the analysis has been taken from [19], although the results differ significantly.

Assume that N_1 stations enter the EB1 period of a specific DIME-EF contention resolution cycle where the duration of the EB1 of each station is given according to Equation 1. Then, the probability that the EB1 period ends after i slots and exactly k stations survive, is given by:

$$P_{E1,i,k}(i,k) =$$

$$\begin{cases} (1-p_{E1})^k & ; i=1;\ k=N_1 \\ \binom{N_1}{k}\left[p_{E1}^{i-1}(1-p_{E1})\right]^k \cdot \left(1-p_{E1}^{i-1}\right)^{N_1-k} & ; 1 < i < m_{E1} \\ \binom{N_1}{k}\left(p_{E1}^{i-1}\right)^k \cdot \left(1-p_{E1}^{i-1}\right)^{N_1-k} & ; i = m_{E1} \end{cases} \tag{3}$$

Consequently, the probability that exactly k stations survive EB1, can be represented as:

$$P_{E1,k}(k) = \begin{cases} \sum_{i=2}^{m_{E1}} P_{E1,i,k}(i,k) & ; 1 \le k < N_1 \\ \sum_{i=1}^{m_{E1}} P_{E1,i,k}(i,k) & ; k = N_1 \end{cases} \tag{4}$$

These k stations are the ones that enter the EB2 period. The average duration $\bar{T}_{E1}$ of EB1 can be calculated as:

$$\bar{T}_{E1} = \mathrm{E}\left(P_{E1,i}(i)\right) \cdot T_{slot} = T_{slot} \cdot \sum_{j=1}^{m_{E1}} j \cdot P_{E1,i}(j) \tag{5}$$

where $\mathrm{E}(\cdot)$ denotes the expected value, T_{slot} a slot duration in IEEE 802.11 and

$$P_{E1,i}(i) = \begin{cases} (1-p_{E1})^{N_1} & ; i = 1 \\ \sum_{k=1}^{N_1} P_{E1,i,k}(i,k) & ; 1 < i \le m_{E1} \end{cases} \tag{6}$$

is the probability that the EB1 period ends after i slots.

The same calculation is now being performed for the EB2 cycle, cf. Equation 2. Let N_2 denote the number of stations entering the EB2 cycle. Then, the probability that EB2 ends after i slots with k stations left, is given by:

$$P_{E2,i,k}(i,k,N_2) = \begin{cases} \left(\frac{1}{m_{E2}}\right)^k & ; i=1;\ k=N_2 \\ \binom{N_2}{k}\frac{(i-1)^{N_2-k}}{m_{E2}^{N_2}} & ; 1 < i \le m_{E2} \end{cases} \tag{7}$$

The expected duration of an EB2 cycle depends on the outcome of the EB1 cycle in terms of numbers of surviving stations and can be represented as:

$$\begin{aligned} \bar{T}_{E2}(N_2) &= T_{slot} \cdot \sum_{N_2=1}^{N_1} P_{E_1,k}(N_2) \cdot \mathrm{E}\left(P_{E_2,i}(i,N_2)\right) \\ &= T_{slot} \cdot \sum_{N_2=1}^{N_1}\left[P_{ES,k}(N_2) \cdot \sum_{i=1}^{m_{E2}} \cdot P_{E2,i}(i,N_2)\right] \end{aligned} \tag{8}$$

where

$$P_{E2,i}(i,N_2) = \begin{cases} \left(\frac{1}{m_{E2}}\right)^k & ; i=1;\ k=N_2 \\ \sum_{k=1}^{N_2} P_{E2,i,k}(i,k,N_2) & ; 1 < i \le m_{E2} \end{cases} \tag{9}$$

denotes the probability that the EB2 cycle ends after i slots. The overall collision probability P_c is the situation where more than one station survive the EB2 cycle and can be calculated as:

$$P_c = \sum_{N_2=2}^{N_1} (1 - P_{E_2,k}(1, N_2)) \cdot P_{E1,k}(N_2) \tag{10}$$

with

$$P_{E_2,k}(k, N_2) =$$

$$\begin{cases} \sum\limits_{i=2}^{m_{E2}} \binom{N_2}{k} \frac{(i-1)^{N_2-k}}{m_{E2}^{N_2}} & ; 1 \leq k < N_2 \\ \left(\frac{1}{m_{E2}}\right)^{N_2} + \sum\limits_{i=2}^{m_{E2}} \binom{N_2}{k} \frac{(i-1)^{N_2-k}}{m_{E2}^{N_2}} & ; k = N_2 \end{cases} \tag{11}$$

as the probability that k out of N_2 stations survive the EB2 cycle.

The overhead O_1 of a single access attempt depends on three main values:

- The expected duration $T_{DIME-EF}$ of a successful DIME-EF cycle, i.e. a cycle which finishes without a collision. This is given by the sum of $\bar{T}_{E1}$, see Equation 5, $\bar{T}_{E2}$, see Equation 8, and $2 \cdot T_{slot}$ for the carrier sensing slots after EB1 and EB2.
- The time it takes to detect a collision of the DIME-EF scheme, T_{coll}. A collision will be detected if the RTS of the sender is not answered by a CTS of the receiver. The medium can be accessed again by a EF station after a PIFS following the RTS. This time is denoted by T_{RTS}, and $T_{coll} = T_{DIME-EF} + T_{RTS}$.
- The collision probability P_c according to Equation 10.

The overhead for a single access attempt in terms of average duration of the DIME-EF scheme, then, is calculated as:

$$O_1(m_{E1}, p_{E1}, m_{E2}, N_1) = P_c \cdot T_{coll} + (1 - P_c) \cdot \bar{T}_{DIME-EF} \tag{12}$$

Iterating this overhead of a single access cycle for subsequent access cycles, weighted with the residual collision probability for a single attempt, yields the average overhead O:

$$O(m_{E1}, p_{E1}, m_{E2}, N_1) = O_1 \cdot \frac{1}{1 - P_c} \tag{13}$$

The overhead O can be interpreted as a function that weighs the overhead of the collision avoidance scheme against the additional overhead that needs to be spent if collisions occur. It is clear that, the more overhead is spent for the collision avoidance, the smaller the collision probability. On the other hand, each collision adds a certain well-known amount of overhead because it can be detected due to the RTS/CTS scheme used. Note that O depends on the parameters given in Equation 13. The optimum parameter set for m_{E1}, p_{E1} and m_{E2} is found when the overhead O reaches its minimum for a given N_1, i.e. we seek $\min(O(m_{E1}, p_{E1}, m_{E2}, N_1))$. The function O has been computed and has the following properties:

- There is always a dedicated minimum for a given value of N_1.
- The minimum is very stable for values of m_{E1} and m_{E2} bigger than the ones for the minimum.
- The value of p_{E1} can be chosen from a big range around the optimum value without significant impact on the overhead.
- The bigger N_1, the bigger the value of the optimum value for p_{E1}. The optimum values for m_{E1} and p_{E1} remain almost unchanged.
- The residual collision probability P_c decreases with increasing values of m_{E1}, p_{E1} and m_{E2}. The increase of m_{E2} has the biggest impact.

The selection of N_1 depends on the usage scenario. For the optimization of the DIME-EF scheme, it was assumed that 10 EF stations almost completely occupy the available data rate of the Basic Service Set. This scenario was assumed to be the worst case for the 2 Mbit/s DS modus of IEEE 802.11 and was chosen as the reference scenario. By iteratively performing simulations and adapting m_{E1}, p_{E1} and m_{E2}, it was found that in this scenario, on average approximately seven stations enter each DIME-EF cycle. Therefore, $N_1 = 7$ was chosen. The resulting optimum values for m_{E1}, p_{E1}, m_{E2}, the resulting overhead O and the residual collision probability P_c are:

$$\begin{aligned} p_{E1} &= 0.43 \\ m_{E1} &= 5 \\ m_{E2} &= 4 \\ O &= 218.69\mu s \\ P_c &= 10.4\% \end{aligned} \tag{14}$$

All simulations presented in Section VI-A have been performed using this parameter set.

B. Rationale for choosing the DIME-EF scheme

The DIME-EF scheme can be compared to other schemes with only one elimination phase. The bursting is necessary because of the carrier sensing of legacy stations that may interrupt a DIME-EF cycle. Assume the following very simple scheme with similar overall overhead as the DIME-EF scheme with the parameters according to Equation 14: Each station chooses a random number of slots for its elimination burst duration out of 9 possible slots. After the bursting, the stations sense the channel for 1 slot. If it is free, they immediately continue with the transmission of an RTS, otherwise they withdraw their access attempt. Assuming an equal distribution according to Equation 2, the overhead can be calculated according to the equations given above for the EB2 cycle. The overhead O for the reference scenario, then, has a value of approx. $226\mu s$ at a residual collision rate of $P_c = 34.6\%$. It is immediately obvious that the overhead is bigger and the number of access attempts for a single packet to be transmitted is higher than with the DIME-EF scheme. A similar calculation with similar results can be performed for a geometric distribution.

It is the combination of the two EB cycles with the probability distributions according to Equations 1 and 2 which makes the DIME-EF scheme very efficient. The EB1 cycle has the property that the probability for a low number of surviving stations entering the EB2 cycle, is high, almost independent from the number N_1 of contending stations. The EB2 cycle, then, is well suited to sort out a single station out of a low number N_2 of remaining stations.

C. Hidden Node Algorithm

Hidden stations are a severe problem in distributed Wireless LAN systems. In the above explanation of the DIME-EF proposal we did not consider this problem. The Hidden Node Algorithm (HNA) proposed in the context of HIPERLAN [20] could also be used to handle hidden stations in DIME-EF. With this solution, the performance of non-hidden stations is preserved, and only the hidden nodes see their performance significantly affected.

V. CONTENTION WINDOW COMPUTATION FOR DIME-AF

In the DCF mode of the 802.11 standard, the size of the CW determines the probability for a station to win the contention. The smaller the CW is, the higher the probability of getting access to the channel. As a consequence, there is a direct relationship between the CW assigned to a station and the bandwidth that this station will receive in a specific scenario. AF can therefore be provided by assigning to a station the CW corresponding to the bandwidth requested by this station.

The difficulty of this approach, however, relies in determining the CW that will lead to the specified bandwidth. Note that this value depends on the number of stations that compete for accessing the channel and their CWs, which is a changing condition.

The solution that we present in this section for the computation of the CW has been more thoroughly studied by the authors in [1].

A. Contention Window Computation

The approach we have chosen for the calculation of the CW in DIME-AF is a dynamic one: each station monitors the bandwidth experienced and modifies its CW in order to achieve the desired throughput. For each packet transmitted, we estimate the sending rate of the terminal; in the case that the estimated rate is smaller than the desired one, we slightly decrease the CW, while in the opposite case, we increase it slightly.

The above explanation describes the basics of the algorithm. However, in the adjustment of the CW, there are additional aspects that have to be taken into account:

- We do not want the CW to increase above the values used by the Best Effort terminals, since this would lead to a worse performance than Best Effort. On the other hand, as explained in Section III, for backward compatibility reasons, the CW for Best Effort should be the one defined by the 802.11 standard.
- If the low sending rate of the application is the reason for transmitting below the desired rate, then the CW should obviously not be decreased.
- When estimating the sending rate, it would be desirable to control the allowed burstiness of the source.
- CWs should not be allowed to decrease in such a way that they negatively influence the overall performance of the network.

Considering all the above issues, we have designed an algorithm for the computation of the CW, which is inspired in the token bucket algorithm. In our scheme, we use the number of bytes in the bucket (bucket length) and the occupancy of the transmission buffer (queue length) as input parameters in the algorithm (see Figure 3). This is further explained in the following points:

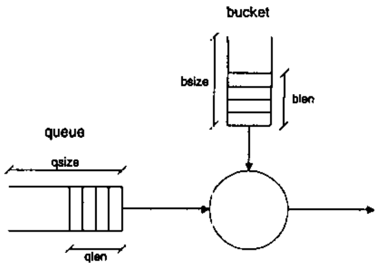

Fig. 3. Token bucket algorithm for AF.

- The token bucket gets filled at the desired transmission rate. For each successful transmission, the length of the transmitted packet in bytes is subtracted from the bucket. Thus the bucket length ($blen$) represents the resources that the user has for transmitting packets.
- The user has resources to transmit a packet only if the bucket has enough bytes in it (we have taken a certain limit $blim$ to represent the minimum needed).
- The bucket size ($bsize$) determines the accepted burstiness of the source; the maximum length allowed to a burst is equal to $bsize - blim$.
- The queue length ($qlen$) expresses the willingness of a station to transmit packets. The CW is only decreased if the queue is not empty (if the queue is empty, the user is not filling it, which means that the current CW satisfies the sending needs of the user).
- When increasing the CW, the value assigned to it can never exceed the size of the CW used for Best Effort.
- If the channel is detected to be below its optimum limit of throughput due to too small values for the CWs (i.e. overutilization), the CW is increased. This aspect is discussed in detail in the following clause.

The above considerations lead to the following algorithm. This algorithm computes a value p which is used to scale the CW values defined in 802.11. Note that, besides this scaling of the CW, the backoff time computation algorithm is left as defined in the 802.11 standard (i.e. the Contention Window is doubled after each unsuccessful transmission attempt for a given number of times).

$$\begin{gathered}
\texttt{if}\ (qlen = 0)\ \texttt{then}\ p = (1 + \Delta_1)p \\
\texttt{else if}\ (blen < blim)\ \texttt{then}\ p = (1 + \Delta_2)p \\
\texttt{else}\ p = (1 - \Delta_3)p \\
p = min\{p, 1\} \\
CW = p \cdot CW_{802.11}
\end{gathered} \tag{15}$$

where Δ_1 is a constant and Δ_2 and Δ_3 are calculated in the following way

$$\Delta_2 = \frac{blim - blen}{blim} \Delta_1 \tag{16}$$

$$\Delta_3 = \frac{blen - blim}{bsize - blim} \Delta_1 \quad (17)$$

The presented algorithm depends on a number of parameters, namely $blim$, $bsize$ and Δ_1. Simulations have shown that the tuning of these constants is not critical for the performance of the protocol as long as they have reasonable values. In the simulations results presented in Section VI-B we have taken $blim$ equal to $token\_size$, $bsize$ equal to $5 * token\_size$, $token\_size$ equal to 1072 bytes and Δ_1 equal to 0.025.

B. Overload

So far we have not discussed one important issue which is the *overload*. In fact, due to the nature of our algorithm and, in particular, due to the dynamic way of adjustment of the size of the CW, a mechanism for controlling the overload is necessary.

As we can see in (15), each station adjusts its CW only on the basis of its own requirements. Such "selfishness" can lead to an unstable state, due to the following side effect of the CWs. We have been arguing so far that, the smaller the CW for a given station, the bigger the probability for this station of seizing the channel before any other station. But another consequence of such a procedure is that the more stations with a small CW, the bigger the probability of a collision. If there is a large number of AF stations, this can lead to an absolute blockage of the channel. Once all of the stations start decreasing their CWs in order to get the requested bandwidth, the number of collisions will start increasing, and this will decrease the overall throughput of the channel, and, as a consequence, the bandwidth experienced by each station. This will lead to even smaller CWs, and therefore, to an unstable state with continuous collisions. A solution to avoid this situation, which we have called *overload*, is to extend (15) with the following condition:

$$\begin{array}{c} \texttt{if } (overload) \texttt{ then } p = (1+\Delta_4)p \\ \texttt{else if } (qlen = 0) \texttt{ then } p = (1+\Delta_1)p \\ \texttt{else if } (bsize < blim) \texttt{ then } p = (1+\Delta_2)p \\ \texttt{else } p = (1-\Delta_3)p \\ p = min\{p, 1\} \\ CW = p \cdot CW_{802.11} \end{array} \quad (18)$$

where $\Delta_4 = 0.25$ is again a constant.

The above equation requires of some way to detect when we are in a situation of overload. As mentioned before, in a situation of overload each station experiences a large number of collisions. Therefore, if we now provide each station with a collision counter[2], which determines how many collisions in average a packet experiences before it is successfully transmitted, we can write the following simple condition to determine overload

$$\texttt{if } (av\_nr\_coll > c) \texttt{ then } overutilization = true, \quad (19)$$

where c is a constant that has to be properly adjusted. If c is too low, AF stations will not be allowed to decrease their CWs sufficiently, and as a consequence they will not be able to achieve the desired bandwidth. On the other hand, if c is too large, the number of collisions in the channel will be very high and the overall performance will be harmed. This constant, therefore, represents a tradeoff between the level of differentiation of AF against Best Effort and the efficiency (i.e. total throughput) of the channel. This tradeoff has been studied via simulation in [1]; results show that a value of $c = 5$ leads to a good tradeoff between throughput differentiation and total throughput. This is the value for c that we have used in this paper.

The average number of collisions, (av_nr_coll), in Equation 19 is calculated after each successful transmission in the following way

$$av\_nr\_coll = (1-t) * num\_coll + t * av\_nr\_coll \quad (20)$$

where in order to smoothen its behavior, we use some sort of memory, taking into account the last calculated value of av_nr_coll (on the rhs of Equation 20). The constant t is a small number (in our case $t = 0.25$) playing the role of a smoothening factor.

VI. SIMULATIONS

To test the performance of the DIME architecture presented in this paper, we have simulated it on a network consisting of a number of wireless terminals in a 2 Mbps Wireless LAN communicating with a fixed node. These simulations have been performed in ns-2 [21].

A. Expedited Forwarding

For the purpose of simulating the contention resolution scheme described in Section IV, the existing implementation of the 802.11 MAC protocol in ns-2 has been extended by the functions necessary for the DIME-EF. In all simulations, stations using the normal 802.11 MAC protocol (i.e. Best Effort in our architecture) coexist with stations using EF, in such a way that each station uses either EF or Best Effort. The Best Effort stations always have something to transmit. The packet length of the Best Effort stations is set to 500 bytes for all simulations. No packet dropping is applied for EF stations. The traffic of the EF stations is of UDP type, since UDP is usually applied in conjunction with EF.

As a quality criterion, we set a maximum delay of 25 ms. This limit shall not be exceeded by 3% or more of the packets. Therefore, the emphasis of all simulations is on delay. The total number of stations in all simulations is 20, i.e. the number of stations using Best Effort is 20 minus the number of EF stations. The stations are located such that they are all within communication distance of each other.

Simulation results for constant bit rate (CBR) sources with 500 bytes packet length are shown in Figures 4 and 5. The simulation results in Figure 6 are obtained for 100 bytes packet length. Each of them shows an inverse distribution of the delay in seconds for a given number of EF stations with the data rate of a single station as parameter. The interpretation of the graphs is as follows: If one is interested to know how many percent of the packets have a delay higher than the one selected, one must

---

[2] Note that in 802.11 collisions can only be detected through the lack of the Ack. However, a missing Ack can also be caused by other reasons different than a collision. In [1] we study the impact into our algorithm of having missing Acks due to errors in the channel.

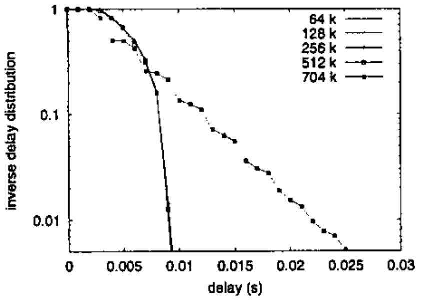

Fig. 4. Inverse delay distribution for 2 EF stations, CBR, 500 byte packet length

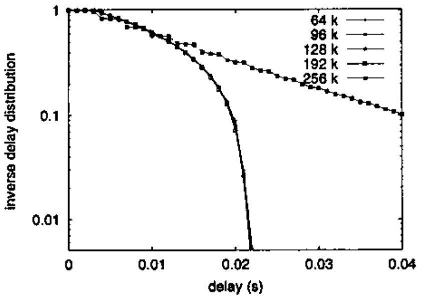

Fig. 5. Inverse delay distribution for 6 EF stations, CBR, 500 byte packet length

pick the time on the x-axis and read the corresponding number on the y-axis. A number of 0.1 on the y-axis means that 10% of the packets were later than the selected time.

The simulations show that the air interface can bear a EF saturation throughput of about 1300 Kbps for 500 bytes packet length and of below 700 Kbps for a packet length of 100 bytes. In general, the saturation throughput decreases with decreasing packet length because each packet carries a certain, more or less constant overhead. However, it is likely that some real-time applications, such as voice over IP, use short packet lengths and, hence, the performance of the DIME-EF scheme for short packets is important.

As long as the required total data rate of the EF stations remains below the saturation throughput, the actual throughput of each EF station corresponds to its required data rate. The data rate left is being used by the Best Effort stations and is shared almost equally between them. If the required data rate of the EF stations exceeds the saturation throughput, the EF stations share the maximum data rate equally, whereas the Best Effort stations get no throughput at all.

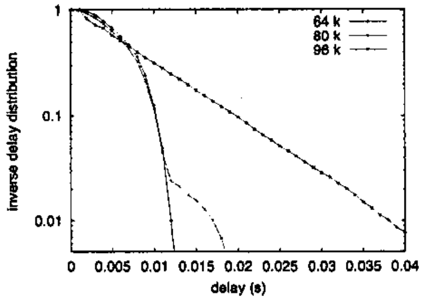

Fig. 6. Inverse delay distribution for 6 EF stations, CBR, 100 byte packet length

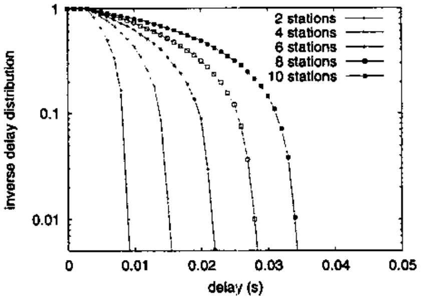

Fig. 7. Inverse delay distribution 64 Kbps with varying numbers of EF stations, CBR , 500 bytes packet length

Figure 4 shows the results for two stations using EF. The data rates range from 64 Kbps up to 704 Kbps per EF station. As can be seen, the delay for all data rates up to 512 Kbps remains below 10 ms in all cases. The data rates achieved by the EF stations corresponds to the data rate delivered by the traffic sources, i.e. they can deliver all offered packets to the destination. The delay increases at a data rate of 704 Kbps but still remains below the allowed limit. In this case, however, the throughput of the EF stations is limited to approximately 650 Kbps, i.e. half of the saturation throughput for each station. The Best Effort stations could not deliver any packet during this simulation run.

The curves depicted in Figure 5 show a similar situation. The delays are higher than with two stations but still remain in the allowed region. If each station uses 256 Kbps, the saturation throughput is exceeded and the delays increase significantly. The same scenario with a packet length of 100 bytes per EF station is shown in Figure 6. The quality criterion can be met for 6 stations with 64 Kbps each but the saturation throughput is already reached with a data rate of 96 Kbps per EF station.

The fact that the delay distribution is almost independent from the data rate of each terminal is quite in line with the results obtained in [19]. An interpretation of the results is that, as long as the EF stations do not always have something to transmit, they do almost not interfere with each other and the transmission delay of the packets plus the waiting time for the end of an onging transmission is decisive. Since the packet length is constant for all EF and Best Effort stations, all have to wait equally long on average. Furthermore, the delay depends on the number of Best Effort stations contending for the channel. The dependence of the transmission delay on the packet length and on the number of Best Effort stations is a subject for further study in this project.

The graphs in Figure 7 show the delay distributions for a data rate of 64 Kbps per station at a packet length of 500 bytes and varying numbers of stations. The same situation for a packet length of 100 bytes is depicted in Figure 8. It can be seen from the graphs that the quality criterion is met for 6 EF stations. For 8 EF stations and more, the quality criterion can not be met. Whereas the curves for 8 and 10 stations with 500 bytes packet length decrease very rapidly, the curves for 100 byte packets have a longer tail.

The graph in Figure 9 shows the inverse delay distribution for ON/OFF traffic for 6 EF stations with a data rate of 64 Kbps and 100 and 500 bytes packet lengths. The ON/OFF sources

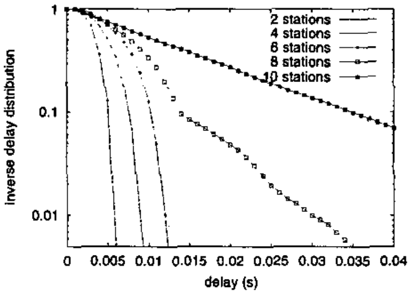

Fig. 8. Inverse delay distribution 64 Kbps with varying numbers of EF stations, CBR , 100 bytes packet length

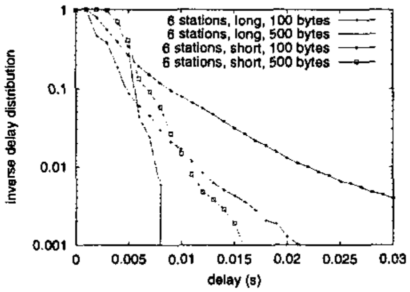

Fig. 9. Inverse delay distribution 64 Kbps for 6 EF stations, ON/OFF traffic, 100 and 500 bytes packet lengths

have exponentially distributed burst and idle times. The curves denoted with "long" have an average burst time of 2 s and idle time of 8 s, whereas the curves with "short" have average burst and idle times of 500 ms. The data rate is the average data rate of the EF stations, i.e. the peak data rate is higher than the average rate. The curves for 500 bytes packet length are rather steep, whereas the 100 bytes curves have a rather long tail. All scenarios meet the quality criterion.

Table I shows which combinations of numbers of EF stations and data rate per EF station meet the quality criterion for all possible scenarios, i.e. for 500 and 100 bytes packet lengths as well as for CBR and ON/OFF traffic. A cross in the table entry means that the criterion is met. As a rule of thumb, an admission control derived from this table would allow not more than six stations and not more than a data rate of 64 Kbps per station to use EF. As can be seen from the delay graphs above, a more sophisticated admission control scheme would have to consider additional criteria, such as the packet length and burstiness of EF applications.

	data rate (kbps)				
stations	32	64	128	256	512
2	x	x	x	x	x
4	x	x	x	x	
6	x	x			
8	x				
10					

TABLE I

OVERVIEW WHICH CONFIGURATIONS MEET THE QUALITY CRITERION

It can be seen from the simulation results that the contention resolution scheme described in Section IV can meet the quality criterion reliably for at least 6 EF stations with data rates up to 64 Kbps per station for CBR and ON/OFF traffic with packet lengths of 100 and 500 bytes. If the available bandwidth is used by EF stations up to or close to the saturation throughput, however, the EF stations use most or all of the available bandwidth and the service quality of Best Effort stations drops dramatically.

B. Assured Forwarding

For the simulation of the CW computation algorithm for AF described in Section V, Equation 18 was inserted into the existing implementation of the 802.11 MAC DCF protocol in ns-2. In the simulations performed, stations using the normal 802.11 MAC protocol (i.e. Best Effort in our architecture) coexisted with stations using AF, in such a way that each station used either AF or Best Effort. The packet length was set to 1000 bytes.

We chose to use the RTS/CTS mechanism. This mechanism, optional in the 802.11 standard, increases bandwidth efficiency in case of many collisions, since with this mechanism collisions occur with the relative small control packets rather than with long data packets. Since our architecture may lead to larger number of collisions than the normal 802.11 MAC DCF, this mechanism can be especially beneficial in our case.

Figure 10 shows the resulting bandwidth distribution when a varying number of AF and BE stations send UDP CBR traffic. AF stations receive a bandwidth assurance such that a total amount of 1 Mbps is assigned to AF (i.e. in the case of 1 AF station, this station receives a bandwidth assurance of 1 Mbps; in the case of 2, each receives a bandwidth assurance of 500 Kbps; in the case of 4, 250 Kbps; and in the case of 8, 125 Kbps).

It can be seen that the total bandwidth received by AF (ideally 1 Mbps shared among the AF stations) decreases with the number of Best Effort stations. This is due to the fact that, as argued in Section III, it is impossible to avoid a certain level of impact of Best Effort stations on the Assured Service.

In the point corresponding to 8 AF stations and 2 Best Effort, AF receives a throughput much higher than the one committed (1.3 Mbps). Note, however, that if only the committed 1 Mbps was given to AF, the 2 Best Effort stations would experience each a higher throughput than an AF station, since they would share the remaining 1 Mbps. The nature of the mechanism we have proposed in DIME-AF for the CW computation ensures that this undesirable situation does not occur: with our algorithm, the leftover bandwidth is equally shared between AF and Best Effort stations such that a Best Effort station never receives a better treatment than an AF one.

Figure 11 shows the bandwidth received by an AF station and the one received by a Best Effort station in the case of 2 AF stations and varying the total number of stations. It can be seen that it is not only AF stations but also Best Effort which see their bandwidth decreased when the total number of stations increases. Note that even though with 50 stations AF stations get about half of the committed bandwidth (250 Kbps each), they still get a throughput 10 times higher than Best Effort stations, which get about 25 Kbps each. This result could be interpreted as a good tradeoff between differentiation (AF stations get a

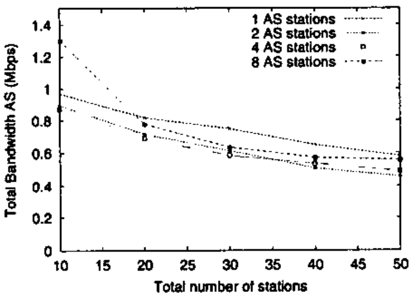

Fig. 10. Impact of Best Effort to the bandwidth for Assured Service.

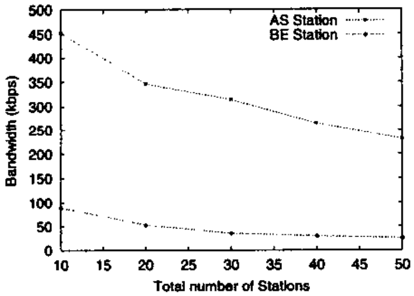

Fig. 11. Assured Service vs. Best Effort.

much higher throughput) and fairness (Best Effort stations do not starve).

For additional simulation results on the DIME-AF algorithm, the reader is referred to [1].

VII. Conclusions

In this paper we have proposed the DIME architecture for providing QoS in Wireless LAN. Since Wireless LAN may be considered as just one hop in the communications path, the goal of DIME is to provide for Wireless LAN the same Per-Hop Behaviors that have been standardized for DiffServ: Expedited Forwarding and Assured Forwarding.

The DiffServ architecture for wired links is based on simple mechanisms with minimal control and signaling, and does not require to keep per-flow state at core nodes. The design of DIME has been done following the same principles: it is based on distributed control, minimizing thus the signaling overhead at the MAC layer, and it does not require to keep per-flow state at the MAC level.

Also like DiffServ, the DIME architecture provides a soft kind of QoS, i.e. statistical QoS guarantees are given to traffic aggregates, but an individual packet does not receive any kind of guarantee. Note that this fits well the type of QoS that can be achieved with a distributed and connectionless MAC.

DIME consists of two extensions to the IEEE 802.11 standard: DIME-EF and DIME-AF. The fact that these extensions have been designed as independent modules gives the manufacturer the option to omit one of them if either EF or AF are not needed, simplifying thus the migration effort from the existing standard.

DIME-EF reuses the PIFS of the 802.11 standard in a distributed manner. We show that distributed control can meet the requirements of real-time services. Simulations have proved that with proper admission control the proposed extension satisfies the user's requirements for low delay.

DIME-AF modifies the CW computation of the DCF mode of the standard. The algorithm for the computation of the CW has been designed in such a way that 802.11 terminals behave as Best Effort terminals in the proposed architecture. The simulations performed show that this extension provides bandwidth assurance to AF terminals in normal circumstances, while the leftover bandwidth is shared equally between Best Effort and AF. Furthermore, starving Best Effort terminals is avoided in case of overload by trading off the bandwidth assurance of AF.

References

[1] A. Banchs and X. Pérez, "Providing Throughput Guarantees in IEEE 802.11 Wireless LAN," in *Proceeding of IEEE Wireless Communications and Networking Conference (WCNC 2002)*, Orlando, FL, March 2002.

[2] R. Braden, D. Clark, and S. Shenker, "Integrated Services in the Internet Architecture: an Overview," RFC 1633, June 1994.

[3] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An Architecture for Differentiated Services," RFC 2475, December 1998.

[4] V. Jacobson, K. Nichols, and K. Poduri, "An Expedited Forwarding PHB," RFC 2598, June 1999.

[5] J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski, "Assured Forwarding PHB Group," RFC 2597, June 1999.

[6] V. Jacobson, K. Nichols, and K. Poduri, "The Virtual Wire Per-Domain Behavior," Internet draft, July 2000.

[7] B. Carpenter and K. Nichols, "A Bulk Handling Per-Domain Behavior for Differentiated Services," Internet draft, January 2001.

[8] N. Seddigh, B. Nandy, and J. Heinanen, "An Assured Rate Per-Domain Behavior for Differentiated Services," Internet draft, February 2001.

[9] T. Nandagopal, S. Lu, and V. Bharghavan, "A Unified Architecture for the Design and Evaluation of Wireless Fair Queuing Algorithms," in *Proceedings of ACM MOBICOM*, Seattle, WA, August 1999.

[10] S. Lu, V. Bharghavan, and R. Srikant, "Fair Scheduling in Wireless Packet Networks," in *Proceedings of ACM SIGCOMM*, Cannes, France, August 1997.

[11] M. Barry, A. Veres, and A. T. Campbell, "Distributed Control Algorithms for Service Differentiation in Wireless Packet Networks," in *Proceedings of INFOCOM*, Anchorage, Alaska, April 2001.

[12] A. Ayyagari, Y. Bernet, and T. Moore, "IEEE 802.11 Quality of Service - White Paper," IEEE 802.11-00/028.

[13] A. Imad and C. Castelluccia, "Differentiation Mechanisms for IEEE 802.11," in *Proceedings of INFOCOM*, Anchorage, Alaska, April 2001.

[14] N. H. Vaidya, P. Bahl, and S. Gupta, "Distributed Fair Scheduling in Wireless LAN," in *Proceeding of ACM MOBICOM*, Boston, MA, August 2000.

[15] V. Kanodia, C. Li, B. Sadeghi, A. Sabharwal, and E. Knightly, "Distributed Multi-Hop with Delay and Throughput Constraints," in *Proceeding of ACM MOBICOM*, Rome,Italy, July 2001.

[16] J.L Sobrinho and A.S. Krishnakumar, "Real-Time Traffic over the IEEE 802.11 Medium Access Control Layer," *Bell Labs Technical Journal*, Autumn 1996.

[17] M. A. Visser and M. E. Zarki, "Voice and Data transmission over an 802.11 Wireless network," in *Proceeding of PIMRC*, Toronto, Canada, September 1995.

[18] ETSI, "Broadband Radio Access Networks (BRAN); HIgh PERformance Radio Local Area Network (HIPERLAN) Type 1; Functional Specification," European Norm 300 652 (V1.2.1), ETSI, 1998.

[19] S. Chevrel et al., "Analysis and Optimisiation of the HIPERLAN Channel Access Contention Scheme," Wireless Personal Communications 4, pp. 27-39, Kluwer Acadamic Publishers, 1997.

[20] J. Wozniak, "Performance Analysis of IEEE 802.11 and ETSI HIPERLAN Networks in the Presence of Hidden Stations," *Systems-Science*, vol. 24, no. 14, pp. 81–96, 1998.

[21] "Network simulator ns-2," `http://www.isi.edu/nsnam/ns/`.