Providing QoS in Ad Hoc Networks with Distributed Resource Reservation

Hamidian, Ali; Körner, Ulf

Published in:
[Host publication title missing]

DOI:
10.1007/978-3-540-72990-7_30

2007

Providing QoS in Ad Hoc Networks with Distributed Resource Reservation

Ali Hamidian and Ulf Körner

Department of Communication Systems
Lund University, Sweden
Box 118, 221 00 Lund
{ali.hamidian, ulf.korner}@telecom.lth.se

Abstract. As the use of WLANs based on IEEE 802.11 increases, the need for QoS becomes more obvious. The new IEEE 802.11e standard aims at providing QoS, but its contention-based medium access mechanism, EDCA, provides only service differentiation, i.e. soft QoS. In order to provide hard QoS, earlier we have proposed an extension called *EDCA with resource reservation* (EDCA/RR), which enhances EDCA by offering also hard QoS through resource reservation. In this paper, we extend EDCA/RR to cope with the hidden terminal problem, outline a solution for multi-hop scenarios, and compare the proposed scheme with EDCA.

Keywords: QoS, IEEE 802.11e, ad hoc network, distributed resource reservation

1 Introduction

Nowadays many of us use portable devices that support *wireless local area networks* (WLANs) based on the IEEE 802.11 standard [1] more or less frequently in universities, homes, cafés, train stations and airports. However, we usually do not reflect over the fact whether they support *quality of service* (QoS) - at least not until we start using an application that has strict QoS requirements. One important step towards solving the QoS issue is the IEEE 802.11e standard [2]. This standard enhances the *medium access control* (MAC) sublayer of 802.11 by specifying the *hybrid coordination function* (HCF) and its two medium access mechanisms: *enhanced distributed channel access* (EDCA) and *HCF controlled channel access* (HCCA). The contention-based EDCA provides QoS by delivering traffic based on differentiating user priorities while the contention-free HCCA provides QoS by allowing for reservation of transmission time.

Although HCCA is an important enhancement that aims at providing QoS guarantees in WLANs, it is EDCA that has received most attention so far. Thus, it is possible that EDCA's destiny will be similar to the one of its predecessor: the *distributed coordination function* (DCF). In other words, EDCA might be implemented by the majority of the vendors, whereas HCCA might be somewhat neglected just as the *point coordinator function* (PCF) - despite the fact that HCCA is a great improvement compared to its predecessor. In addition, EDCA is a distributed channel access method and can be used in ad hoc networks while HCCA is centralized and thus only usable in infrastructure networks. Hence, the focus of this paper lies on EDCA.

The motivation of our work is based on the fact that EDCA provides only service differentiation so it cannot guarantee any QoS. Our proposed scheme, called *EDCA with resource reservation* (EDCA/RR), extends EDCA by allowing for resource reservation with the aim of providing QoS guarantees. When talking about QoS guarantees, we must keep in mind that since a wireless medium is much more unpredictable and error-prone than a wired medium, QoS cannot be guaranteed as in a wired system, especially not in unlicensed spectra. However, it is possible to provide techniques that increase the probability that certain traffic classes get adequate QoS and that can provide QoS guarantees in controlled environments.

The remainder of this paper is organized as follows: Section 2 discusses some related work and Sect. 3 gives an overview of the 802.11e standard with focus on EDCA. In Sect. 4 we present EDCA/RR and also a solution for extending the scheme to handle multi-hopping and releasing unused reservations. The simulation results are presented and discussed in Sect. 5. Finally, Sect. 6 concludes the paper and gives some directions for future work.

2 Related Work

There has been a lot of research on providing QoS in ad hoc networks. However, some of these suggest proprietary protocols - based on time division multiple access, multiple channels, etc. It is our belief that any realistic proposal must be based on the widely spread de facto 802.11 standard(s). Among the work that is based on the existing standards, many are focused on infrastructure-based networks. Regarding ad hoc networks, most proposed solutions provide only service differentiation and not QoS guarantees. Below we shortly mention two such works but also two that aim at providing QoS guarantees through resource reservation.

To improve the performance of EDCA, Romdhani et al. [3] adjust the value of the *contention window* (CW) taking into account both application requirements and network conditions. Iera et al. [4] follow another, but similar, approach by dynamically adapting the priority class instead of the CW value, also taking into account both application requirements and network conditions. The improvement of EDCA is still based on service differentiation so it is not possible to provide QoS guarantees.

Hiertz et al. present the *Distributed Reservation Request Protocol* (DRRP) [5], which is a decentralized MAC scheme based on 802.11. The scheme is similar to EDCA/RR, allowing stations to reserve access to the medium. Whenever a station (A) needs to reserve medium access for communication with another station (B), it sends a data frame containing reservation request information. Upon reception of such a reservation request, B sends an *acknowledgment* (ACK) frame that also contains information about the reservation request. The ACK frame is overheard by the neighbors of B and thus, the stations hidden to A are also informed about its reservation request. The reservation request includes information regarding the duration and repetition interval of the next transmission. All neighbors receive the reservation information by overhearing the transmissions between A and B. However, since the neighbors do not acknowledge the overheard frames and these can be lost, the reservation request is transmitted periodically so the information about its periodicity is also included in the reservation request.

The main disadvantage with DRRP is that it does not have any admission control and does not aim to provide QoS guarantees. Moreover, DRRP introduces a new frame structure, which is not compatible with the existing standards.

The *Distributed end-to-end Allocation of time slots for REal-time traffic* (DARE) [6], by Carlson et al., is another decentralized MAC scheme based on 802.11. DARE allows stations to reserve periodic time slots. In particular, DARE extends the RTS/CTS reservation concept of 802.11 to a multi-hop end-to-end perspective. To reserve resources for a real-time flow over several hops, the routing protocol at the source must first find a route to the destination. The route is assumed to be symmetric. Once such a route is established, the source sends a *request-to-reserve* (RTR) frame, which includes the requested duration and periodicity of a time slot as well as the address of the destination node. When an intermediate node receives the RTR frame, it checks whether the request is conflicting with already existing reservations. If the intermediate node can make the requested reservation, it processes the RTR frame and forwards it; otherwise the request is rejected. Once the destination receives the RTR frame, it responds with a *clear-to-reserve* (CTR) frame. When the source receives the CTR frame, it can start transmitting real-time traffic at the next reserved interval. DARE is also able to repair and release unused reservations. One of the main disadvantages with DARE is the very complex and inefficient method for multiple reservations. A requested reservation may conflict with existing ones so, for example, if a requested receive slot cannot be admitted, an *Update-Transmit-Request* (UTR) frame must be sent back to the node that proposed the receive slot in order to re-schedule (shift in time) the slot. The UTR frame suggests at least another receive slot but the suggested slot might still be inappropriate. So nodes might have to send messages back and forth trying to find a suitable reservation slot, and this can happen at every hop! The authors mention that slot shifting becomes necessary more frequently as the number of reservations increases. Thus, new reservations can only be admitted if they can squeeze in between existing ones. Furthermore, although multi-hopping is one of the advantages of DARE, there is no mechanism for the routing protocol to consider the QoS requirements of the requested reservation during the route discovery process. Thus, the routing protocol might find a route that cannot support the requested reservation. Another disadvantage is that there is no RTS/CTS, ACK or retransmission for real-time frames (could have been optional), which can result in lost real-time frames.

In [7] we proposed a mechanism that supports QoS guarantees in a WLAN operating in ad hoc network configuration, i.e. in a single-hop ad hoc network. Although single-hop ad hoc networks might be seen as limited, we must remember that the main application area for EDCA is a WLAN and not a multi-hop ad hoc network. However, in this paper we extend our earlier proposal to, among others, cope with the hidden terminal problem and we also outline a solution for multi-hop scenarios. The scheme is based on the existing standards 802.11 and 802.11e and consequently, it can be integrated into existing systems without much difficulty. To give an example of the application area for single-hop ad hoc networks where our scheme can be used, we can mention network gaming where players can use their laptops to play demanding network games with each other at no cost anywhere they want; i.e. without needing to worry about (neither wired nor wireless) Internet connections.

3 An Overview of IEEE 802.11e

Since the 802.11 standard did not address the QoS issues sufficiently, the 802.11 working group wrote the new standard 802.11e. This new standard introduces the concepts of *transmission opportunity* (TXOP) and *traffic specification* (TSPEC) in order to solve a few problems in 802.11 related to the provisioning of QoS.

To solve the problem with unknown transmission times in 802.11, the concept of TXOPs was introduced. A TXOP is a time interval defined by a starting time and a maximum duration. During a TXOP, a station may send several frames as long as the duration of the transmissions does not extend beyond the maximum duration.

In the 802.11 standard, there is no way for a station to send its QoS requirements to an *access point* (AP). To solve this problem, the 802.11e standard allows stations to use TSPECs to describe the QoS characteristics of a traffic stream. This is done by specifying a set of parameters such as frame size, service interval, service start time, data rate and delay bound. Most of the above-mentioned parameters are typically set according to the requirements from the application while some are generated locally within the MAC. The service interval specifies the time interval between the start of two consecutive TXOPs. The service start time specifies the time when the service period starts, i.e. when the station expects to be ready to send frames.

3.1 Enhanced Distributed Channel Access (EDCA)

The distributed and contention-based medium access mechanism of 802.11e, EDCA, is an enhanced variant of DCF. The main problem with DCF, regarding QoS provisioning, is that it cannot provide any service differentiation since all stations have the same priority. In addition, DCF uses one single transmit queue. To overcome this problem, in EDCA each station has four *access categories* (ACs) and four transmit queues that contend for TXOPs independently of the other ACs. Thus, each AC behaves like an enhanced and independent DCF contending for medium access. Before entering the MAC sublayer, frames are assigned a *user priority* (UP) and based on these UPs each frame is mapped to an AC. Besides using the UPs, the frames can be mapped to ACs based on frame types. The management type frames, for example, shall be sent from AC_VO (without being restricted by any admission control though). The four ACs can be used for different kind of traffic: AC_BK for background traffic, AC_BE for best effort traffic, AC_VI for video traffic and AC_VO for voice traffic. Differentiated medium access is realized by varying the contention parameters[1] for each AC:

- CWmin[AC] and CWmax[AC]: the minimum and maximum value of the CW used for calculation of the backoff time. These values are variable and no longer fixed per PHY as with DCF. By assigning low values to CWmin[AC] and CWmax[AC], an AC is given a higher priority.
- *arbitration interframe space number* (AIFSN[AC]): the number of time slots after a SIFS duration that a station has to defer before either invoking a backoff or starting

[1] In our simulations, EDCA uses the default values for these parameters according to the 802.11e standard.

a transmission. AIFSN[AC] affects the *arbitration interframe space* (AIFS[AC]), which specifies the duration (in time instead of number of time slots) a station must defer before backoff or transmission: AIFS[AC] = SIFS + AIFSN[AC] × slot_time. Thus, by assigning a low value to AIFSN[AC], an AC is given a high priority.

– $TXOP_{limit}[AC]$: the maximum duration of a TXOP. A value higher than zero means that an AC may transmit multiple frames (if all belong to the same AC since a TXOP is given to an AC and not to a station) as long as the duration of the transmissions does not extend beyond the $TXOP_{limit}[AC]$. Thus, by assigning a high value to the $TXOP_{limit}[AC]$, an AC is given a high priority.

4 EDCA with Resource Reservation (EDCA/RR)

In a previous work we have enhanced EDCA to provide QoS guarantees by reserving TXOPs for traffic streams with strict QoS requirements [7]. In this section we start by giving an introduction to EDCA/RR in order to facilitate the reading and understanding of the rest of this paper. Then we describe our solution to a problem in EDCA/RR related to hidden stations. Next, we present a conceivable solution to extend EDCA/RR such that it can be used in multi-hop ad hoc networks. Finally, we identify some problems that might occur due to mobile stations leaving and entering a network.

The EDCA/RR scheme works like EDCA as long as there is no station that needs to reserve TXOPs for its high-priority traffic stream. Once a station (sender) wishes to reserve TXOPs to be able to send traffic with strict QoS requirements, it requests admission for its traffic stream. The request is not sent to any central station such as an AP, but is handled internally within the sender by an admission control algorithm. It should be mentioned that our scheme is not dependent on any specific admission control or scheduling algorithm; thus, it is possible to use any proposed enhancement (such as those presented in [8], [9], [10] and [11]) to the reference design algorithms provided in the 802.11e standard. However, in our EDCA/RR implementation, the reference admission control and scheduling algorithms have been used because they are specified in the standard, making them widely known and giving them certain acceptance.

In case the traffic stream is rejected, the sender can try to lower its QoS demands and retry. On the other hand, if the traffic stream is admitted, the sender schedules its traffic by setting the *service interval* (SI) and the *service start time* (SST) parameters. Details about the calculation of these parameters can be found in [7]. Next, the sender broadcasts an *add traffic stream* (ADDTS) request containing a TSPEC element with information such as mean data rate, nominal frame size, SST and SI. All stations that receive the ADDTS request store the information of the sender's SST and SI, and schedule the new traffic stream exactly as the sender. This ensures that no station starts a transmission that cannot be finished before a reserved TXOP starts and thus collision-free access to the medium is offered to the streams with reserved TXOPs. In order to make sure that all stations have similar schedules, all neighbors have to unicast an ADDTS response back to the sender to acknowledge a received ADDTS request.

Every time the sender receives an ADDTS response from a neighbor, it stores the address of the neighbor. After receiving a response from all neighbors, the sender waits until the SST specified in the TSPEC element and initiates a transmission. If the time instant when all responses are received occurs later than the advertised SST, the transmission is delayed until the next TXOP. Once the TXOP is finished, the station waits until the next TXOP, which occurs after an SI.

When a transmission failure occurs during a TXOP, the station does not start a backoff procedure. Instead, it retransmits the failed frame after SIFS if there is enough time left in the TXOP to complete the transmission.

4.1 The Hidden Station Problem in EDCA/RR

In the original version of EDCA/RR, the hidden station problem was handled through the exchange of *request to send* (RTS) and *clear to send* (CTS) frames, i.e. the same way as in 802.11/e. However, this method is not sufficient since in EDCA/RR, stations hidden to a station that has reserved TXOPs can cause other problems than the well-known hidden station problem (causing collisions). Contending stations that have received a TSPEC from the reserving station do not start a transmission unless it finishes before a TXOP starts. But unfortunately, stations hidden from the reserving station do not receive any TSPEC so they do not know when a TXOP starts. Therefore, they might start a transmission that extends across a TXOP.

To illustrate the problem, suppose there are three stations in a row (see Fig. 1): A, B and C, where A and B as well as B and C are within each others transmission range but A and C cannot hear each other. Assume further that A wants to send traffic with QoS requirements to B so it has broadcasted an ADDTS request and B has replied with an ADDTS response. However, C (that is hidden from A) is unaware of A's TXOP reservation since it has not received A's ADDTS request so there is a chance that C starts transmitting just before a TXOP reserved by A is about to start. In that case a collision would occur during A's reserved TXOP meaning that A would no longer have collision-free access to the medium. In order to prevent C from transmitting just before a reserved TXOP is about to start, it must become aware of A's TXOP reservation. In other words, the reservation schedule of any sender must be known by all stations within two hops from the sender.

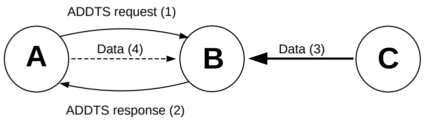

Fig. 1: C is hidden from A and can start transmitting just before A's TXOP starts.

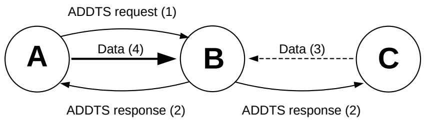

Fig. 2: C is informed about the TXOP reservation of A and defers during A's TXOP.

There are different ways of achieving this goal, i.e. to spread the TSPEC to stations outside of the reserving station's transmission range. One approach can be to rebroadcast the ADDTS request sent by the reserving station during the TXOP reservation. Hence, in our example B would rebroadcast the ADDTS request of A to let also C receive the request frame. However, there are many problems related to this approach.

First, should C send an ADDTS response to B just like B has to send an ADDTS response to A? We must remember that there might be many stations at the same distance from A as B and C respectively (i.e. one and two hops away from A respectively). This means that if C has to respond to B then every other station two hops away from A should also respond to B because those are also hidden stations. Moreover, this procedure would continue until all stations one hop away from A rebroadcast the ADDTS request from A, and all stations two hops away from A send back an ADDTS response. Obviously, this would lead to a lot of overhead and a significant increase in the reservation delay. On the other hand, if C does not have to send a response to B, then B cannot be sure whether the rebroadcasted ADDTS request was received by C or not.

Another approach to spread the TSPEC is to let the ADDTS responses contain the TSPEC and let all stations overhear these frames (see Fig. 2). This way, the TSPEC is known to all stations within two hops from the sender with no additional signaling and with limited increase of overhead. Thus, when B sends an ADDTS response back to A, C will hear this frame and save the information included in the TSPEC, i.e. the SST and SI of A. This approach is much less complex and results in less overhead than the previous approach. However, again B cannot be sure whether the ADDTS response was received correctly by C or not. Therefore, we let reserving stations transmit special RTS/CTS frames extended to contain a TSPEC (RTS_TSPEC and CTS_TSPEC), in the beginning of a TXOP. This way, a station with an out-of-date reservation schedule has the chance to update its schedule. Although one might think that this increases the overhead, we must remember that the RTS_TSPEC and CTS_TSPEC frames are sent only at the beginning of a TXOP and not for every single data frame.

4.2 Further Enhancements

The first goal of this paper was to enhance the EDCA/RR scheme operating in single-hop ad hoc networks. In particular, we wanted to solve the problems that could occur due to hidden stations. We have achieved this goal and presented our solution above. Another goal was to present an extension to the scheme such that it can be used to provide QoS guarantees in a multi-hop ad hoc network.

QoS Provisioning in Multi-hop Ad Hoc Networks. To extend EDCA/RR for multi-hop ad hoc networks, we need an ad hoc routing protocol that can find a route between the communicating stations. In this paper we assume that the routing protocol is reactive, like for example *Ad hoc On-Demand Distance Vector* (AODV) [12] and *Dynamic MANET On-demand* (DYMO) [13]. Using one of these protocols, during the route discovery process the source broadcasts a *route request* (RREQ) throughout the network to find the destination. When the destination receives the RREQ, it responds with a *route reply* (RREP) unicast toward the source.

To illustrate the idea of our solution, let us assume there are three stations in a row: A, B and C, where A and B as well as B and C are within each other's transmission range but A and C cannot hear each other. Assume further that A wants to send high-priority traffic (with QoS requirements) to C.

To reserve resources along a multi-hop route, the QoS requirements of A's traffic stream must be known by the routing protocol so that it can use the requirements during the route discovery process. However, the routing protocol shall not start its route discovery process before the traffic stream has been admitted by the local admission control mechanism at A's MAC sublayer. In EDCA/RR, the resource reservation process (including admission control) starts when the first packet of a traffic stream with QoS requirements reaches the MAC sublayer, i.e. after it has been handled by the routing protocol at the network layer. To prevent the routing protocol searching for a route using its usual metrics (and thus not considering the QoS requirements of the traffic stream), it must be modified to co-operate with the protocol at the MAC sublayer (EDCA/RR in our case). Therefore, once the first high-priority packet of a traffic stream in station A reaches its network layer, the TSPEC information provided by the application (e.g. mean data rate and nominal frame size) is copied from the packet into the RREQ that is generated. These fields describe some characteristics of the application that are needed during the calculation of SI and TXOP duration at the MAC sublayer. In addition, the fields contain information about how the route discovery procedure should proceed in case there are not enough resources to reserve. This is necessary because although some applications need a certain minimum level of QoS for functioning, others can function despite that the QoS level is not sufficient. Therefore, if a traffic stream cannot be admitted, an application may prefer a normal route (not taking into account the QoS requirements of the application) rather than aborting.

When the RREQ extended with TSPEC information is sent down to the MAC sublayer, the admission control algorithm in EDCA/RR checks whether the requested traffic stream can be admitted. If the traffic stream is rejected, depending on the requirements of the application, there are two possible cases: either the application is notified to abort or the station can search for a normal route to the destination. On the other hand, if the traffic is admitted, the route discovery process is started to search for a route that can handle the QoS requirements. In that case, A broadcasts an extended RREQ and when B receives the message, its MAC sublayer uses the TSPEC information in the message to check whether the traffic can be admitted or not. In case the traffic is rejected, again depending on the QoS requirements of A's application, either it is notified to abort or B searches for a normal route to the destination. On the other hand, if the traffic is admitted, the extended RREQ is rebroadcasted and received by C. Station C processes the extended RREQ mainly as in B. However, if the traffic is admitted, the MAC sublayer schedules the traffic stream of A and notifies the routing protocol to send an extended RREP, containing information about the successful reservation, back to the source (i.e. station A).

When B receives the extended RREP, its MAC sublayer uses the TSPEC information in the message to schedule the traffic stream, which now has been admitted by all stations from the source to the destination. Then the RREP is forwarded to A, which processes the message just as in B. Finally, the resource reservation is finished and the traffic stream can start transmitting during its reserved TXOPs. Thus, to summarize, the RREQs check whether the requested resources are available while the RREPs confirm the TXOP reservations and do the actual scheduling of the traffic.

Leaving and Entering the Network. An important issue that needs special attention is mobility. If a station with reserved TXOPs leaves the network, the other stations must become aware of that because otherwise they will defer from transmitting although they should not and network capacity will be wasted. A conceivable solution to this problem is to let the stations in the network listen for frames in the beginning of each TXOP in order to determine whether the reserved TXOPs are still in use. (The station with the reserved TXOP may transmit dummy packets when it does not have any data to send.) If several consecutive TXOPs are determined to be unused, the receiver can ask the sender if it still has something to send. If the sender does not respond despite several attempts, the receiver and other stations can assume that the sender has left the network and delete the reservation by deallocating the TXOPs. Furthermore, possible TXOPs after the terminated TXOP, shall be moved back within the SI in order to avoid time gaps between TXOPs. Moving the streams is done pretty easily thanks to the distributed characteristic of EDCA/RR. There is no need for any signaling; each station performs the rescheduling itself in a distributed manner.

On the other hand, if a station enters a network with existing reserved TXOPs, it cannot start transmitting because such a transmission may collide with the transmission in a reserved TXOP. Instead, the station must update its schedule before it is allowed to transmit any frame. This can be done by setting a schedule update bit in beacons or other frames exchanged during the initialization process.

5 Evaluation

In order to evaluate the performance and effectiveness of EDCA/RR, we implemented our scheme in the widely-used network simulator ns-2 and compared it with EDCA. Since the 802.11 implementation in ns-2 is rather simple, we used another more advanced and detailed 802.11 implementation (developed by Mike Moreton), which also implements 802.11a/b/g and some features of 802.11e. Based on this detailed implementation, we implemented EDCA/RR. Various scenarios have been studied though just one is presented in this paper.

5.1 Simulation Setup

The studied scenario consists of a variable number of stations where each traffic stream is sent from a unique source to a unique destination. There is always one high-priority source while the number of low-priority sources is varied from zero to five. All low-priority sources are hidden to the high-priority source.

The stations use 802.11b DSSS (with short preamble) in the physical layer and 802.11e EDCA or EDCA/RR in the MAC sublayer. The routing protocol AODV is used at the network layer. At higher layers, the low-priority streams, which are sent from AC_BE, generate TCP segments with a size of 1000 bytes according to an FTP application that always has data to transmit. This is because we want to stress the compared MAC schemes regarding the QoS provisioning to a high-priority stream, by increasing the proportion of low-priority traffic load in the network. On the other hand, the high-priority stream generates UDP packets according to a *constant bit rate* (CBR) traffic generator. The high-priority stream is sent from AC_VO and models voice traffic encoded using the G.711 codec. The codec generates voice packets at a rate of 64 kbit/s, with packet size of 160 bytes and packet interarrival time of 20 ms. The protocols at each layer add some overhead to the voice packet so in total we have: G.711 (160 B) + RTP (12 B) + UDP (8 B) + IP (20 B) + 802.11 MAC (28 B) + 802.11b PHY (short preamble: 96 μs).

In this paper, we have studied the stationary behaviour of EDCA and EDCA/RR. More specifically, we studied the impact of an increasing number of low-priority streams on one high-priority stream. We calculated the average end-to-end delay[2], jitter and squared coefficient of variance of the end-to-end delay ($C^2[d]$) for the high-priority stream when the number of low-priority streams was varied between zero and five. For each of the six data points we ran 300 simulations each during 300 simulated seconds. Since we were interested in studying the behaviour of the network in steady state, the first 30 seconds of the simulations were ignored. Then, we calculated the average of the 300 averaged values for each data point. Because of the extensive simulations, we could calculate the 99% confidence interval of the average end-to-end delay.

5.2 Simulation Results

In Tables 1 and 2 we can see the average end-to-end delay and its corresponding 99% confidence interval for the high-priority stream. Table 1 shows the results for the case when the medium is error-free while Table 2 shows the corresponding results for the case when the packet error rate is 5 %. Both tables show that, as the number of low-priority streams increases, the average end-to-end delay for the high-priority stream increases rapidly for EDCA. This is a typical behaviour for contention-based medium access schemes and it is this kind of behaviour that we would like to avoid. Another typical, but more advantageous, behaviour for random-access schemes is that they have very low medium access delays when the network load is light. This is also shown in the tables. On the other hand, EDCA/RR provides an almost constant average end-to-end delay to the high-priority stream. The result shows that EDCA/RR succeeds offering periodic medium access to the high-priority stream no matter how large the network load is. Furthermore, it shows that EDCA/RR can handle hidden stations.

Table 1: 99% confidence interval of the average end-to-end delay of the high-priority stream - 0 % packet error

nbr of LP-streams	delay (ms)		confidence interval (ms)	
	EDCA	EDCA/RR	EDCA	EDCA/RR
0	0.69	12.33	(0.69,0.69)	(12.13,12.53)
1	6.21	12.22	(6.20,6.22)	(12.02,12.42)
2	11.17	12.27	(11.14,11.19)	(12.08,12.47)
3	13.93	12.22	(13.90,13.96)	(12.01,12.42)
4	17.12	12.38	(17.08,17.16)	(12.19,12.57)
5	20.51	12.25	(20.46,20.56)	(12.06,12.45)

Table 2: 99% confidence interval of the average end-to-end delay of the high-priority stream - 5 % packet error

nbr of LP-streams	delay (ms)		confidence interval (ms)	
	EDCA	EDCA/RR	EDCA	EDCA/RR
0	0.99	12.55	(0.99,0.99)	(12.37,12.73)
1	4.68	12.44	(4.68,4.69)	(12.27,12.61)
2	5.25	12.54	(5.24,5.25)	(12.35,12.73)
3	5.59	12.34	(5.58,5.60)	(12.16,12.52)
4	5.92	12.64	(5.91,5.93)	(12.45,12.82)
5	6.28	12.53	(6.27,6.29)	(12.34,12.72)

[2] The end-to-end delay is calculated as the time when a frame is received by the destination's application layer minus the time when the frame was generated at the application layer of the source.

Comparing the two tables reveals that, under EDCA/RR, when the packet error increases to 5 % the end-to-end delay increases marginally. This is a result of the fact that retransmissions most often take place within the same TXOP as the discarded packet. However, with further increasing packet error probabilities, not even EDCA/RR may handle the QoS requirements. Under EDCA, the situation is a bit different. The tables show that the average end-to-end delay decreases with increasing packet error probabilities. This might seem surprising at first but the explanation is that, when the packet error rate is 5 % for EDCA, the low-priority frames are affected much more negatively than the high-priority frames. This is because the CW, which is doubled after each lost frame, rapidly becomes much larger for the low-priority streams than for the high-priority stream[3] resulting in longer medium access delays. Moreover, since the number of transmitted, and thus lost, low-priority frames are larger than the corresponding number of high-priority frames, the low-priority connections are broken more often. The consequence is that the number of simultaneously active low-priority streams is almost always lower than the actual number. Thus, to the high-priority stream the network is less loaded than it would be if the channel would be error-free. This also explains why EDCA shows a lower end-to-end delay than EDCA/RR. Still, one might argue that since there are no error-free channels in the real world, why use EDCA/RR when EDCA results in a lower average end-to-end delay. First, we could have set up much tougher QoS requirements, say 5 ms end-to-end delay and EDCA/RR would have met them. Second, we measure delays for those packets that really reach the receiver, though after a number of retransmissions. Under EDCA the number of packets that really reach the receiver, is much lower than for EDCA/RR.

Table 3 shows the end-to-end jitter and the corresponding squared coefficient of variance of the end-to-end delay ($C^2[d]$) for the high-priority stream when the channel is error-free. As expected, we see that when the high-priority stream is the only active stream in the network, the jitter and the $C^2[d]$ are very low. The explanation for this is the same as why the average end-to-end delay is very low in this situation; i.e. the frames get instant access to the medium. However, the general view is that while EDCA/RR provides very low variations those for EDCA increase significantly with the number of low-priority streams. The behaviour of EDCA is not acceptable for multimedia applications with QoS requirements on low jitter.

Table 3: Jitter and $C^2[d]$ for the high-priority stream - 0 % packet error

nbr of LP-streams	jitter ($10^{-6}s^2$)		$C^2[d]$	
	EDCA	EDCA/RR	EDCA	EDCA/RR
0	0.02	48	0.05	0.32
1	40	48	1.04	0.32
2	180	48	1.45	0.32
3	276	48	1.42	0.32
4	406	49	1.38	0.32
5	577	49	1.37	0.32

[3] The default CWmax value for AC_VO and AC_BE is equal to 15 and 1023 respectively.

6 Conclusion

In this paper, we have extended the previously proposed EDCA/RR scheme, which allows multimedia applications to reserve medium time. EDCA/RR has been enhanced to prevent hidden stations causing collisions during reserved TXOPs. The main idea was to spread the information about the TXOP reservation such that also hidden stations become aware of the reservation and thus, defer during the reserved TXOPs. EDCA/RR was designed for single-hop ad hoc networks. Although such a scheme can be useful, our aim was to extend it to be useful also in multi-hop ad hoc networks since these networks are expected to offer new communication possibilities. Thus, we have presented an extension to the scheme such that it can be used together with an ad hoc routing protocol to find multi-hop QoS-enabled routes between the communicating stations. Thus, a station sending traffic with QoS requirements will be able to reserve TXOPs for deterministic medium access along a multi-hop route to the destination. The aim in future work will be to incorporate this extension into the existing EDCA/RR implementation.

References

1. ANSI/IEEE: Std 802.11 Part11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. (1999)
2. IEEE: Std 802.11e Part11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications - Amendment 8: Medium Access Control (MAC) Quality of Service (QoS) Enhancements. (2005)
3. Romdhani, L., Ni, Q., Turletti, T.: Adaptive EDCF: Enhanced Service Differentiation for 802.11 Wireless Ad-Hoc Networks. In: Proceedings of IEEE Wireless Communication and Networking Conference. (2003)
4. Iera, A., Molinaro, A., Ruggeri, G., Tripodi, D.: Improving QoS and Throughput in Single- and Multihop WLANs through Dynamic Traffic Prioritization. IEEE Network **19**(4) (July/August 2005)
5. Hiertz, G.R., Habetha, J., May, P., Weib, E., Bagul, R., Mangold, S.: A Decentralized Reservation Scheme for IEEE 802.11 Ad Hoc Networks. In: Proceedings of the 14th International Symposium on Personal, Indoor and Mobile Radio Communications. (2003)
6. Carlson, E., Prehofer, C., Bettstetter, C., Wolisz, A.: A Distributed End-to-End Reservation Protocol for IEEE 802.11-based Wireless Mesh Networks. Journal on Selected Areas in Communications (JSAC), Special Issue on Multi-Hop Wireless Mesh Networks (2006)
7. Hamidian, A., Körner, U.: An Enhancement to the IEEE 802.11e EDCA Providing QoS Guarantees. Telecommunication Systems **31**(2-3) (March 2006)
8. Skyrianoglou, D., Passas, N., Salkintzis, A.: Traffic Scheduling in IEEE 802.11e Networks Based on Actual Requirements. Mobile Location Workshop Athens (2004)
9. Grilo, A., Macedo, M., Nunes, M.: A Scheduling Algorithm for QoS Support in IEEE 802.11e Networks. IEEE Wireless Communications Magazine (June 2003)
10. Fan, W.F., Gao, D., Tsang, D.H.K., Bensaou, B.: Admission Control for Variable Bit Rate Traffic in IEEE 802.11e WLANs. IEEE LANMAN (April 2004)
11. Ansel, P., Ni, Q., Turletti, T.: FHCF: A Fair Scheduling Scheme for 802.11e WLAN. Technical Report 4883, INRIA Sophia Antipolis (July 2003)
12. Perkins, C., Belding-Royer, E.M., Das, S.: Ad hoc On-Demand Distance Vector (AODV) Routing. IETF. RFC 3561.
13. Ogier, R., Lewis, M., Templin, F.: Dynamic MANET On-demand (DYMO) Routing. IETF. Internet draft.