

On the shortcoming of EDCA queues in the presence of multiservice traffic

Agnieszka Brachman¹ and Lukasz Chrost²

¹ Silesian University of Technology, Institute of Informatics, ul. Akademicka 16,
44-100 Gliwice, agnieszka.brachman@polsl.pl

² Silesian University of Technology, Institute of Informatics, ul. Akademicka 16,
44-100 Gliwice, lukasz.chrost@polsl.pl

Abstract. The paper aims at depicting certain limitations of EDCA queues related to the different traffic classes. Through simulation, we show main flaws of identical, medium size, Drop Tail queues used by default for each class, with reference to different traffic category's requirements. We propose some substitutes such as differentiated, basically smaller buffer sizes and Active Queue Management mechanisms that can be employed to improve the overall performance. We present a strategy for EDCA queues to maintain the desired quality of transmission for traffic belonging to different categories.

1 Introduction

Multimedia applications such as Voice over IP (VoIP), video streaming, Video on Demand (VoD) are gaining more and more popularity. At the same time data transmission performed by email, file transfer and web browsing applications is continually increasing. It brings out the necessity to provide different priorities to different applications, users or data flows or to guarantee a certain level of performance to a data flow; in short, to provide Quality of Service (QoS).

The wireless technology based on the IEEE 802.11 standard [1] remains the fundamental solution for wireless LANs. It is widely adopted in homes, offices and public areas causing ousting of wires, although it provides much slower and faultier transmission. High bit error rate along with slower transmission rates hamper the overall QoS and fairness provisioning. QoS must be provided at every stage of transmission and when considering 802.11 WLANs, the weak point lies at the link layer, in the contention-based medium access algorithm.

The IEEE 802.11e amendment introduced new enhancements to enable QoS support in wireless LAN applications through modifications to the Media Access Control (MAC) layer. The amendment has been incorporated into the IEEE 802.11 standard published in 2007. It defines new Hybrid Coordination Function (HCF), where there are two access methods within: HCF Controlled Channel Access (HCCA) and Enhanced Distributed Channel Access (EDCA). The two aforementioned methods represent two different kinds of QoS support: parametrized and prioritized.

HCCA provides contention-free access to the medium by polling the stations and granting so called transmission opportunity of fixed size. Polling is performed in a way that allows fulfilling negotiated QoS requirements. The scheduling process is fairly sophisticated. Despite the fact that the method seems a solution for the problem of providing QoS, it has not been implemented in any device so far. This is probably due to the complexity of the scheduling algorithm. HCCA is beyond the scope of the paper.

EDCA provides a prioritized QoS service by using an independent transmission queue for each traffic category called Access Category (AC). Traffic belonging to a higher priority class has a higher probability of accessing the medium, thus achieving a higher throughput when competing with lower priority traffic. It is achieved by differentiating inter-frame space (AIFS), minimum and maximum contention window size (CW_{min} and CW_{max}) and transmission duration time (TXOP). With EDCA, each station can have up to 4 queues, mapped to different traffic classes. Service differentiation can be achieved by assigning different set of values for every queue contending for access. Traffic with different AC contends independently for the medium access within a station. Performance evaluation of EDCA can be found in [2–5]. Simulation results along with the outcome in real testbeds confirm that EDCA provides prioritized QoS support.

However medium conditions can drastically change over time which can lead to throughput degradation. As the network becomes overloaded, contention window size increases which leads to long backoff counters and high queue occupation. This especially concerns lower priority queues. This may lead to poor network performance and significantly limit bandwidth utilization. In section 3 we show influence on delay, drop rate, goodput under medium and high congestion in WLAN. We depict how low priority queues are overloaded which leads to throughput oscillations and degradation.

In this paper, through simulation, we depict main flaws of identical, medium size, Drop Tail queues used for each class by default, with reference to different traffic category's requirements. In next section we describe substitutes such as differentiated, basically smaller buffer sizes and Active Queue Management mechanism that can be employed to improve the overall performance. Simulation results for proposed enhancements are presented in section 3.

2 Solutions for EDCA queues

We investigated how queue management is solved at routers and tried applying the theory into EDCA queues. Routers avoid a serious throughput reduction in the case of a mild congestion, by employing queue management mechanisms, namely Drop Tail and various AQM (Active Queue Management) schemes. Queuing directly influences packet transmission delays, experienced drop rate and dropping scheme. The usage of Drop Tail or any AQM scheme raises many new issues one of which is the buffer sizing. Below we present a short description of evaluated mechanisms as well as discussion concerning buffer sizes.

Implementing a queue management algorithm to obtain a very low level of the buffer occupancy allows achieving high quality of offered services. However, it can also lead to the link underutilization. In this paper RED (Random Early Detection) as an example of Active Queue Management (AQM) algorithm, is evaluated along with the Drop Tail strategy. A short description of each mechanism is presented below.

The idea of RED is to notify the TCP sender about incoming congestion by dropping packet before the buffer overflow occurs. RED calculates the current network load by counting the exponential moving average of the queue size - avg . If avg is smaller than min_{th} threshold all the packets are enqueued, if $avg > max_{th}$ all the packets are dropped. When avg is between the two thresholds, packets are dropped with linearly increasing probability. More detailed description of RED can be found in [6].

Although active queue management allows achieving the desired average queue length, the simple FIFO queue of fixed buffer size remains the most popular strategy applied at routers. The incoming packets are buffered until the queue is full. When there is no space left, packets are dropped.

The AQM mechanisms allow dynamically adjusting target queue length and packet drop probability and thus outperform Drop Tail mechanism in case of light and moderate congestions. The buffer size and the target queue size are two most important parameters considering queue management. While the use of very large buffers lowers the packet drop probability, it also significantly influences packet transmission delays. Truncation of the buffer size stabilizes the packet delays but it is believed to lead to bandwidth underutilization. However we think that small buffer sizes along with AQMs work for EDCA.

The idea of using RED for EDCA queues is not new. Yang *et al.* [7] presented a Priority Random Early Detection (PRED) algorithm, which integrated Random Early Detection (RED) with EDCA queues. PRED provides a queuing algorithm for the priority of packets within each node. Authors claim that PRED obtains higher throughput especially under heavy traffic load condition, allowing the packet with higher priority to acquire more throughput than Drop Tail. In the presented scenario queues size were set to 70 packets. Queueing delay was not analyzed and we believe that such large buffer sizes would lead to high delays and delay variations, especially for low priority traffic. The larger delay the more probable that packet will be considered lost and retransmitted. This maintains high throughput but significantly lower goodput, due to the unnecessarily retransmitted packets. Throughput is calculated as overall transmission and goodput as the effective transmission (without unnecessary retransmissions). Moreover, the usage of AQM algorithm for all traffic classes is in our opinion unjustified, since VoIP and video transmission can use UDP instead of TCP protocol and UDP does not react to early packet drops.

3 Performance evaluation of EDCA queues

The ns-2 simulator was used to study EDCA queues performance [8]. Since the EDCA model is not included in standard ns-2 release, we based on Sven Wiethlter and Christian Hoene's model, developed at Technical University of Berlin [9]. The model provides a dedicated, priority-driven queue management algorithm, implemented in set of classes. The main class used for the queue management operations (PriQ), stands for a configurable interface for 4 priority-specific Drop Tail queues. We have extended the model, providing a simple mechanism allowing utilization of any built-in AQM scheme for selected EDCA queue.

The paper focuses on simulation model where WLAN is used as a last-hop network. We proposed several scenarios to depict the main flaws and limitations of EDCA queues. Detailed simulation topology and scenarios are described below, along with simulation results.

3.1 Simulation scenarios and topology

The network topology is shown in fig. 1. The wired node is connected to the router with 10 Mbps links. The link propagation delay is set to 1 ms. The link from router R to Access Point AP has the propagation delay of 10 ms and the bitrate 10 Mbps. The wireless bandwidth is set to 11 Mbps which provides around 5-6 Mbps of effective bandwidth. The wireless nodes are uniformly distributed around AP. The wireless network is the bottleneck for each connection.

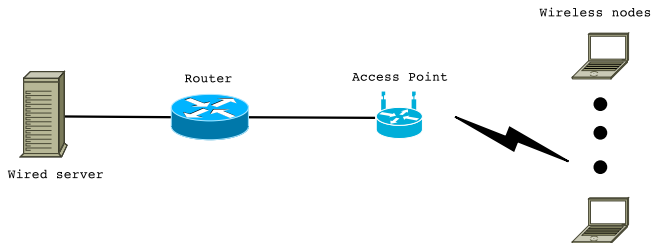


Fig. 1: Network topology

Firstly we investigated the behaviour of EDCA queues related to Access Categories where each queue is a simple FIFO (Drop Tail - DT) queue. The MAC parameters for EDCA, used in the simulations are summarized in table 1.

We focused on three basic scenarios, all with bidirectional traffic. Scenario I and II simulate heavy network congestion, scenario III represents mild network congestion.

Table 1: Default EDCA values for AC queues

| AC | Prio | $AIFS[AC]$ | $CW_{min}[AC]$ | $CW_{max}[AC]$ | $TXOPLimit[AC]$ |
|-------|------|------------|----------------|----------------|-----------------|
| AC_BK | 3 | 7 | 15 | 1023 | 0 |
| AC_BE | 2 | 3 | 15 | 1023 | 0 |
| AC_VI | 1 | 2 | 7 | 15 | 3.008 |
| AC_VO | 0 | 2 | 3 | 7 | 1.504 |

Scenario I - Traffic is composed of FTP connections. Every wireless node transmits and receives data. Four uplink and downlink connections exist in the network, each belonging to different traffic category. TCP SACK is used at transport layer, packet size is 1500 bytes.

Scenario II - Traffic is composed of Constant Bit Rate (CBR) connections. Every wireless node transmits and receives data with constant bitrate set to 1 Mbps. Packet size is 1000 bytes. Four uplink and downlink connections exist in the network, each belonging to different traffic category.

Scenario III - Simulation traffic is composed of Variable Bit Rate (VBR), FTP and WWW connections. Two VBR, one FTP and one WWW connections transmit and receive on every wireless node resulting in eight connections for every node. Parameters for aforementioned connection set are summarized in table 2. VBR streams are generated using Pareto sources. WWW traffic is generated using Pareto-sized bursts of data, mean burst size is 50 KB and pareto shape parameter is 1.3. The burst occurrence is defined by poisson process with mean interval of 1.6 second.

Table 2: Parameters for each connection set in Scenario III

| | Mean bitrate | Packet size | Protocol | Priority |
|-----|--------------|-------------|----------|----------|
| VBR | 64 kbps | 210 | UDP | 0 |
| VBR | 300 kbps | 1000 | UDP | 1 |
| WWW | 500 kbps | 1500 | TCP | 2 |
| FTP | - | 1500 | TCP | 3 |

All connections are active during the entire simulation. Each simulation is run for 110 seconds. Statistics are collected during the last 100 seconds.

3.2 Simulation results

We investigated one AQM algorithm - RED and compared the results with simple FIFO (DT) queue. The simulations were run twice for each scenario, setting a standard and a small buffer limit for DT queue as well as a short and a medium

target queue size for RED. Comparing the gathered results, we proposed a hybrid solution for EDCA queues described in section 3.2.

Results for DT-EDCA queues Figures 2, 3 and 4 show the average queue length measured on the access point, in packets. The results are presented for all scenarios, when buffer size is set to 10 and 50 packets accordingly, for each EDCA queue. Queue limit of 50 packets is considered a standard limit for EDCA queues, therefore the limit of 10 packets should be regarded as a small buffer. 0 indicates the highest and 3 the lowest priority.

TCP connections tend to full link utilization which significantly influences the low priority traffic (fig. 2). The use of small buffers reduces the disparity among flows of different priority. Also the higher priority the lower queue oscillations, which results in small delay variation.

UDP connections do not react to packet drops and do not reduce transmission rate, ergo lower priority connections always utilize the whole buffer (fig. 3). It can lead to excessive queueing delays and to the situation where although the packet arrives, it is no longer valid (i.e. in VoIP or video transmission). Packets are transmitted in both directions therefore network overload is high and download throughput is low. It results in some spikes in the average queue length for second highest access category and variation in queueing delay. It is worth mentioning that the large buffer results in higher spikes and consequently higher delay variation.

In scenario III a mixture of TCP and UDP connections is introduced. Average queue length is showed in figure 4, tables 3 and 4 summarize measured average transmission parameters for uplink and downlink connections belonging to different traffic categories.

UDP connections have the highest priority and they utilize approximately 20% of wireless bandwidth. The size of related queues does not influence delay neither drop rate. Both uplink and downlink flows achieve the maximum throughput. The buffer limit mainly affects transmission delay for the lowest priority traffic. For the 50 packets buffer size, average transmission delay of the lowest priority, downlink connections is over 3 seconds which is interpreted by most applications as non-availability. It is also ten times more than the transmission delay noticed by the uplink connections belonging to the same category. Similar situation occurs for the WWW traffic. Reducing the buffer limit to 10 packets decreases the transmission delay as well as the disparity in the uplink and downlink delay at the cost of a higher drop rate for the downlink FTP connections. This is because of the four connections contending for the lowest priority queue at the Access Point. The uplink connections have their own queue on every wireless node.

Results for RED-EDCA queues Figures 5, 6 and 7 show the average queue length on the Access Point, measured in packets for all scenarios accordingly, when the buffer size is set to 50 packets, for each EDCA queue. We set two

different target queue lengths for the RED algorithm: 10 and 25. It is achieved by setting min_{th} and max_{th} values to: 5, 15 or 20, 30 accordingly.

When the small target queue length is used for the TCP connections, higher link utilization is achieved because of the higher average queue length in comparison to DT (figure 5). Higher target queue length lowers the queue length oscillations. Also the lowest priority traffic is allowed to maintain a higher transmission rate. The queue oscillations would be smaller if the number of connections increased.

AQM by no means influences UDP traffic sources, therefore the simulation results showed in figure 6 are very similar and comparable to the simulation results where Drop Tail is used (figure 3).

Measured, average transmission parameters for the uplink and downlink connections belonging to the different traffic categories when RED-EDCA queues are used are summarized in tables 5 and 6.

Similarly to the results described in the previous section, the target queue length does not affect any transmission parameter of UDP connections. Correspondingly, the transmission delay of the lowest priority traffic drops with the target queue length degradation, as well as the disproportion in the transmission delay of the downlink and uplink connections. RED also significantly improves the transmission delay along with the throughput for the larger target queue length. The drop rate is not affected (compare results for the lowest priority, downlink connections in tables 4 and 6). In comparison to DT, when to short target queue level is set, RED improves the drop rate and maintains slightly longer transmission delays and lower throughput for the lower priority traffic.

Hybrid EDCA queues We propose a hybrid solution for EDCA queues to achieve the best results for each traffic category. The high priority traffic is usually transmitted using UDP protocol, therefore no sophisticated AQM scheme is needed. Also the small buffer size results in low transmission delays which is very important for the real-time traffic. For this categories we propose to use the Drop Tail queues, limited in size to 10 packets. For WWW and FTP traffic RED queues with the target queue level of 25 packets are advised. The average queue length is presented in figure 8, the detailed simulation results for uplink and downlink connections in scenario III are summarized in table 7.

UDP connections of the highest priority are not affected. Transmission parameters are comparable to the results achieved in previous scenarios. For the lower priority traffic, the disparity in throughput of the downlink and uplink connections is decreased at the cost of the higher drop rate for the uplink connections. At the same time, the transmission delay is significantly longer for the downlink connections. However it is maintained in acceptable range.

4 Conclusions

We show, that the use of very small buffers (10 packets) for EDCA queues significantly reduces transmission delay especially for the low priority traffic. At

the same time it can influence the throughput and the drop rate both in positive and negative manner. Through simulation we depict, that applying AQM in place of DT EDCA queues for the TCP traffic results in throughput improvement along with the drop rate reduction. Finally, we propose a hybrid solution for EDCA queues. The solution utilizes very short, DT queues for handling the UDP traffic, along with medium-sized RED queues for handling the TCP traffic. It proved to successfully deal with the described shortcomings of the standard DT EDCA model.

Acknowledgement. This work is supported by the Ministry of Science and Higher Education under grant KBN N N516 381134

References

1. IEEE80211: IEEE Standard for Information technology-Telecommunications and information exchange between systems-local and metropolitan area networks-Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999) (12 2007) C1-1184
2. Choi, S., del Prado, J., Shankar, S., Mangold, S.: IEEE 802.11e contention-based channel access (EDCF) performance evaluation. In: Communications, 2003. ICC '03. IEEE International Conference on. Volume 2. (11-15 May 2003) 1151 – 1156 vol.2
3. He, D., Shen, C.: Simulation study of IEEE 802.11e EDCF. In: Vehicular Technology Conference, 2003. VTC 2003-Spring. The 57th IEEE Semiannual. Volume 1. (22-25 April 2003) 685 – 689 vol.1
4. Ni, Q.: Performance analysis and enhancements for IEEE 802.11e wireless networks. Network, IEEE **19** (July-August 2005) 21 – 27
5. Xiao, Y.: IEEE 802.11e: QoS provisioning at the MAC layer. Wireless Communications, IEEE [see also IEEE Personal Communications] **11** (June 2004) 72 – 79
6. Floyd, S., Jacobson, V.: Random Early Detection Gateways for Congestion Avoidance. IEEE/ACM Transactions on Networking **1** (1993) 397–413
7. Yang, C.C., Li, J.S., Rwei-Yi-Li, Lin, S.Y., Wen, J.H.: QoS Performance Improvement for WLAN Using Priority Random Early Detection. Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on (Sept. 2007) 1996–1999
8. ns2: The Network Simulator ns-2: Documentation. <http://www.isi.edu/nsnam/ns/doc/index.html> (2009)
9. Wiethölter, S., Emmelmann, M., Hoene, C., Wolisz, A.: TKN EDCA Model for ns-2. Technical Report Technical Report TKN-06-003, Technische Universität Berlin (June 2006)

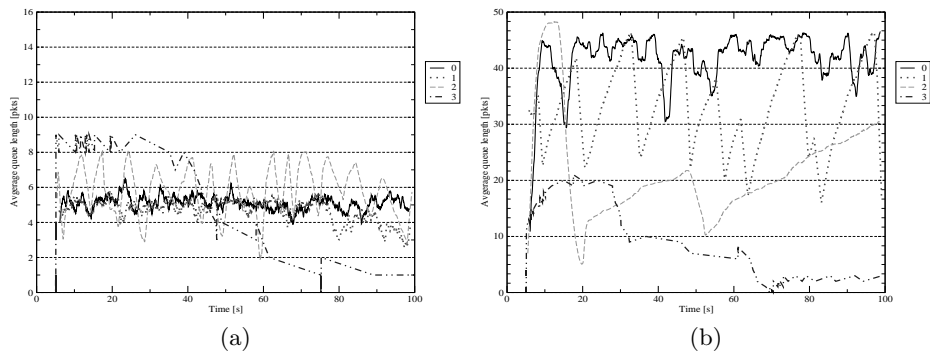


Fig. 2: Scenario I: Average queue length for DT-EDCA queues. The queue limit is set to: (a) 10 and (b) 50 packets.

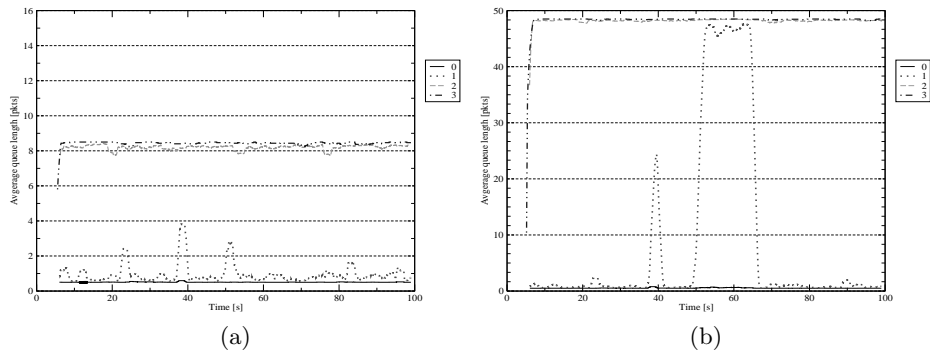


Fig. 3: Scenario II: Average queue length for DT-EDCA queues. The queue limit is set to: (a) 10 and (b) 50 packets.

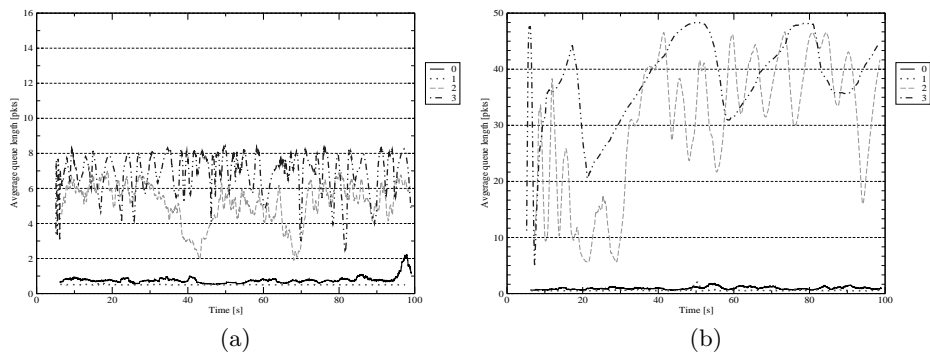


Fig. 4: Scenario III: Average queue length for DT-EDCA queues. The queue limit is set to: (a) 10 and (b) 50 packets.

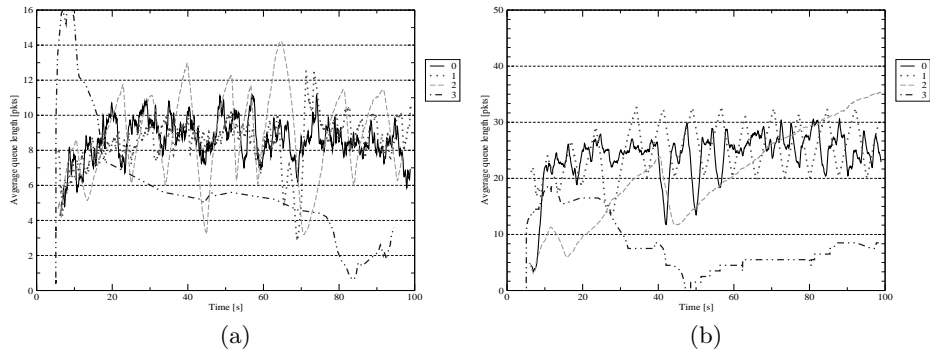


Fig. 5: Scenario I: Average queue length for RED-EDCA queues. The target queue length is set to: (a) 10 and (b) 25 packets.

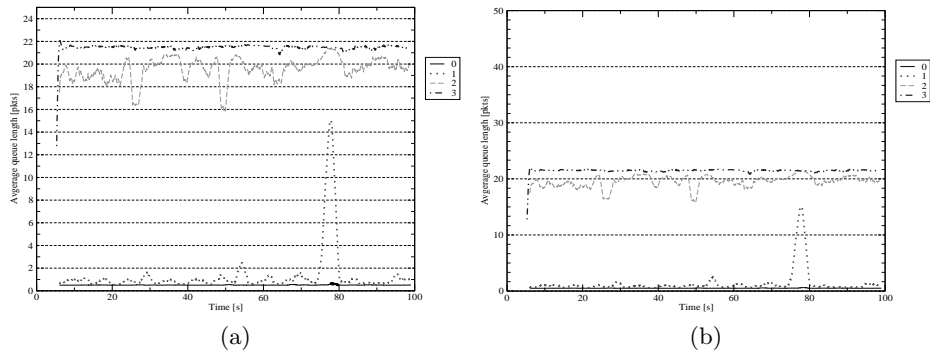


Fig. 6: Scenario II: Average queue length for RED-EDCA queues. The target queue length is set to: (a) 10 and (b) 25 packets.

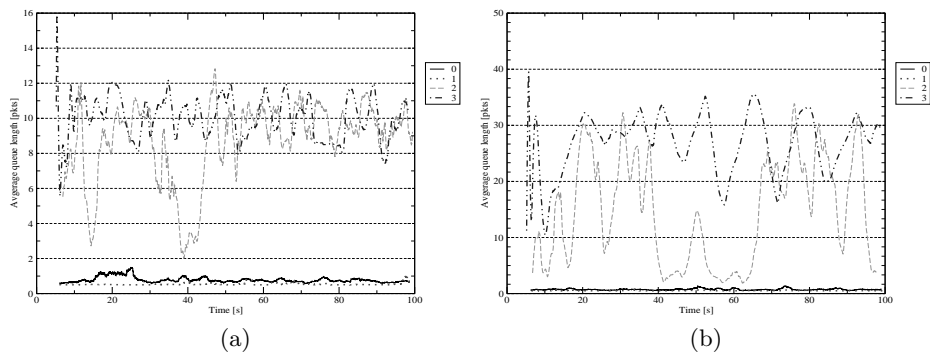


Fig. 7: Scenario III: Average queue length for RED-EDCA queues. The target queue length is set to: (a) 10 and (b) 25 packets.

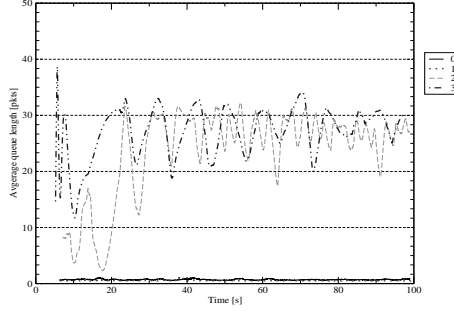


Fig. 8: Scenario III: Average queue lengths for hybrid EDCA queues.

Table 3: Average transmission parameters for the uplink and downlink traffic categories in Scenario III and DT-EDCA queues with the queue limit equal to 10 packets

| Priority: | downlink | | | | uplink | | | |
|----------------|----------|------|-----|------|--------|------|-----|----|
| | 3 | 2 | 1 | 0 | 3 | 2 | 1 | 0 |
| avg thr [kbps] | 206 | 759 | 318 | 107 | 995 | 1006 | 498 | 99 |
| delay [ms] | 460 | 101 | 15 | 11 | 177 | 52 | 14 | 11 |
| drop rate [%] | 12.20 | 6.86 | 0 | 0.12 | 6.54 | 0 | 0 | 0 |

Table 4: Average transmission parameters for the uplink and downlink traffic categories in Scenario III and DT-EDCA queues with the queue limit equal to 50 packets

| Priority: | downlink | | | | uplink | | | |
|----------------|----------|------|-----|------|--------|------|-----|-----|
| | 3 | 2 | 1 | 0 | 3 | 2 | 1 | 0 |
| avg thr [kbps] | 151 | 593 | 485 | 155 | 637 | 1180 | 530 | 141 |
| delay [ms] | 3125 | 602 | 17 | 12 | 276 | 68 | 15 | 11 |
| drop rate [%] | 4.70 | 3.30 | 0 | 0.03 | 6.73 | 5.37 | 0 | 0 |

Table 5: Average transmission parameters for uplink and downlink traffic categories in Scenario III and RED-EDCA queues with the target queue length set to 10 packets

| Priority: | downlink | | | | uplink | | | |
|----------------|----------|------|-----|-----|--------|------|-----|-----|
| | 3 | 2 | 1 | 0 | 3 | 2 | 1 | 0 |
| avg thr [kbps] | 208 | 719 | 451 | 140 | 904 | 735 | 555 | 133 |
| delay [ms] | 638 | 163 | 16 | 11 | 192 | 61 | 14 | 11 |
| drop rate [%] | 8.74 | 5.13 | 0 | 0 | 6.27 | 4.86 | 0 | 0 |

Table 6: Average transmission parameters for uplink and downlink traffic categories in Scenario III and RED-EDCA queues with the target queue length set to 25 packets

| Priority: | downlink | | | | uplink | | | |
|----------------|----------|------|-----|-----|--------|------|-----|-----|
| | 3 | 2 | 1 | 0 | 3 | 2 | 1 | 0 |
| avg thr [kbps] | 227 | 670 | 485 | 122 | 972 | 727 | 710 | 116 |
| delay [ms] | 1436 | 255 | 15 | 10 | 183 | 52 | 14 | 11 |
| drop rate [%] | 4.16 | 0.81 | 0 | 0 | 6.42 | 4.41 | 0 | 0 |

Table 7: Average transmission parameters for uplink and downlink traffic categories in Scenario III and with hybrid EDCA queues

| Priority: | downlink | | | | uplink | | | |
|----------------|----------|------|-----|------|--------|------|-----|-----|
| | 3 | 2 | 1 | 0 | 3 | 2 | 1 | 0 |
| avg thr [kbps] | 254 | 935 | 514 | 121 | 1034 | 603 | 346 | 119 |
| delay [ms] | 1320 | 313 | 15 | 11 | 166 | 49 | 14 | 11 |
| drop rate [%] | 3.57 | 1.51 | 0 | 0.01 | 6.66 | 4.40 | 0 | 0 |