

Handoff Functions for a Distributed Queuing Collision Avoidance Medium Access Control Protocol for Wireless LANs

J. Alonso-Zárate*, E. Kartsakli**, A. Antonopoulos*, L. Alonso** and Ch. Verikoukis*

* Centre Tecnològic de Telecomunicacions de Catalunya (CTTC)

Email: {jesus.alonso, aantonopoulos, cveri}@cttc.es

** Universitat Politècnica de Catalunya (UPC-EPSC)

Email: {ellik, luisg}@tsc.upc.edu

Abstract – The Distributed Queuing with Collision Avoidance (DQCA) Medium Access Control (MAC) protocol has been presented in the literature as a high-performance protocol for uplink traffic in WLANs. Previous works regarding DQCA are focused on the operation of a single cell, where no interaction with neighboring sites is considered. In this paper, we define specific handoff functions (channel sensing, discovery and reassociation functions) that enable the roaming of users in a scenario composed by a number of different DQCA access points deployed in a specific area using different channel frequencies.

Keywords – WLAN, MAC, DQCA, handoff.

I. INTRODUCTION

The Distributed Queuing with Collision Avoidance (DQCA) is a distributed Medium Access Control (MAC) protocol for Wireless Local Area Networks (WLANs) that offers near optimum performance [1]. The protocol implements a reservation scheme that ensures collision-free data transmissions for high traffic load and switches smoothly and automatically to a random access mechanism when the traffic load is light, improving the delay performance under these circumstances.

The original description of DQCA in [1] does not define a handoff process that allows roaming among different access points (APs). The protocol rules described in that paper specify how the nodes get access to the uplink of a single isolated AP. However, no handoff procedure or AP selection mechanisms that must be carried out in cellular environments where several APs are present and users can freely roam have been yet defined for DQCA. This is the main motivation for the work presented in this paper, where we present smart handoff mechanisms for a DQCA network with more than one AP.

Traditionally, handoff decisions rely upon measurements of Received Signal Strength Indicators (RSSI). More specifically, a node moving away from the current AP, called prior-AP, reassociates with another AP, called posterior-AP, whose signaling is received with stronger signal strength than that of the prior-AP [2]-[3]. However, DQCA inherently offers system information that can be used to implement smarter selection mechanisms than those simply based on the traditional RSSI. We present in this paper specific handoff mechanisms for a DQCA network with multiple APs deployed. Not only is the selection of the AP with the best channel quality based on the RSSI, but also to other DQCA specific parameters such as the load balancing among different APs.

The rest of the paper is organized as follows. In Section II we review the DQCA protocol. Then, in Section III we describe the general functions that a handoff process must consider. Based on this, in Section IV we describe the specific handoff functions required to allow the roaming of users among different access points in a multiple-cell site. Finally, Section V concludes the paper.

II. DQCA OVERVIEW

The purpose of this section is to highlight the basic features of DQCA which are essential for the understanding of the proposed handoff procedures. A detailed explanation of DQCA, along with the protocol operating rules, can be found in [1]. As demonstrated there, DQCA outperforms the widely commercially spread Distributed Coordination Function (DCF) of the IEEE 802.11 Standard [4] and remains stable even when the traffic load occasionally exceeds the channel capacity.

DQCA is a MAC protocol designed in order to manage the access to the channel in the uplink of an infrastructure WLAN. Time is divided into MAC frames, and each frame is divided in three parts separated by a Short Inter Frame Space (SIFS) necessary to tolerate propagation delays, turnaround times, and processing delays. The three parts, depicted in Figure 1, are:

- i) A Contention Window (CW) further divided into m access minislots where the nodes can send a short chip sequence named Access Request Sequence (ARS) to request access to the channel. An ARS is a short chip sequence that contains no explicit information but it has a specific and predefined pattern that allows the AP to distinguish between an *idle* minislot, the presence of just one ARS (*success*), and the occurrence of a *collision* between two or more simultaneous ARS [5].
- ii) A data slot reserved for the transmission of data packets.
- iii) A feedback part where the AP broadcasts a Feedback Packet (FBP) that contains the data acknowledgment, the state of each of the minislots of the CW for the contention resolution algorithm and a ‘final message bit’ that is enabled (set to one) by the AP to identify the last data packet (fragment) of a message. Of course, nodes must also include a ‘final message bit’ in their data packet transmissions in order to advertise the transmission of the final fragment of each message. This is the only informa-

tion necessary for the proper execution of the protocol rules in a distributed manner (locally by each node).

The execution of the protocol is based on two concatenated distributed queues, the Collision Resolution Queue (CRQ) and the Data Transmission Queue (DTQ). The CRQ is responsible for the resolution of collisions among ARS (following a tree-splitting collision resolution algorithm [6]-[8]) while the DTQ handles the transmission of data. The number of occupied positions (or elements) in each queue is represented by an integer counter (RQ and TQ for the CRQ and the DTQ, respectively). Both counters have the same value for all the nodes in the system and are updated according to a set of rules at the end of each frame. In addition, each node must maintain and update another pair of integer counters that reveal its position in the queue (pRQ and pTQ for the CRQ and the DTQ, respectively). The *position* refers to the relative order of arrival (or age) of the node in the respective queue. In the CRQ, each position (or element) is occupied by a set of nodes that suffered an ARS collision (i.e. attempted an ARS transmission in the same access minislot of the same CW). On the other hand, each position in the DTQ contains exactly one node that has successfully reserved the channel through an ARS.

III. LOGICAL FUNCTIONS IN A HANDOFF PROCESS

A handoff algorithm refers to the mechanism performed by a wireless device in order to transfer the connectivity from one AP to another. Handoff is a layer-2 function that involves at least three entities, namely the wireless device, a prior-AP and a posterior-AP. The prior-AP is the AP that the station was connected to before the handoff procedure, while the posterior-AP is the AP that the wireless device gets connected to after the handoff function. In general, the implementation of a handoff mechanism requires three interrelated main functions performed by a roaming node (see Figure 2):

1) **Link Status Monitoring (LSM):** This function defines the procedure through which a node monitors the quality and availability of the connectivity with a given AP. In the wireless channel, the received signal strength and/or the signal-to-noise ratio (SNR) of the signal received from the current AP may degrade due to mobility, channel fading, or inter-cell interference. Accordingly, as soon as the quality of a connection degrades below a certain threshold (also defined by the LSM function), the node should initiate a discovery process to obtain information related to other available APs that may provide better performance. Therefore, it is necessary to keep track of the status of the wireless links from neighbor APs, as well as from the current AP. In general, the LSM functions can be based on any individual parameters, such as received signal strength, SNR, bit error rate, or a combination of them. The most common option is to perform the LSM decisions based on the reception of beacon packets broadcast by the APs.

- 2) **Discovery:** This function is performed by any terminal when it is essential to reassociate with another AP due to a bad connectivity with the current AP (as indicated by the LSM function). During the discovery (or scanning) phase the node listens to the channel waiting for the reception of beacon messages broadcast by APs operating in other channels. Therefore, it can create an ordered list of candidate APs prioritized by any metric such as the received signal strength. The specific metric will be defined by the LSM function. Just as an example, there are two kinds of scanning methods defined in the 802.11 Standard: active and passive. The passive scanning mode implies that nodes just listen to beacon messages transmitted from the APs and infer the channel quality from the received signals. In the active mode, in addition to the listening of the beacon messages, nodes transmit additional *probe broadcast packets* on each channel and receive responses from APs.
- 3) **Reassociation:** This function is comprised of two parts, namely, the re-authentication and the reassociation of a node to a new AP. The re-authentication process typically involves a message exchange performing an authentication and a reassociation to the posterior-AP. The re-authentication phase also includes the transfer of credentials and other state information from the prior-AP to the posterior-AP.

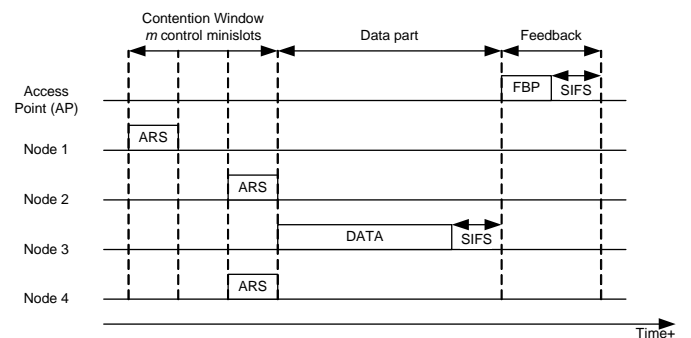


Figure 1 DQCA Frame Structure

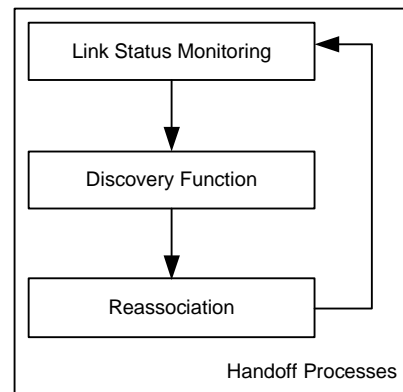


Figure 2 Handoff Processes

IV. DQCA HANDOFF PROCESSES

In this section we specify the handoff operations in the context of a DQCA multi-cellular WLAN system taking into account the general handoff process described in the previous section.

A. Link Status Monitoring (LSM) Function

The frame structure of DQCA facilitates the design of specific LSM functions. More specifically, the FBP broadcast by the AP at the end of each frame can be used as an implicit beacon for the LSM function. This packet can be used by all nodes in order to obtain information regarding the link status with the AP on a frame-by-frame basis. This information may be the received signal strength. However, in the case of DQCA, the values of the TQ and RQ counters of each AP should be also considered in order to select the best AP not only in terms of channel quality, but in terms of traffic load as well. Note that the values of TQ and RQ are representative of the traffic load of an AP. Therefore, by efficiently managing the information regarding the state of the distributed queues, it is possible to attain efficient load balancing in a multicell DQCA network.

In addition to the specific criteria to quantify the *eligibility* of an AP, it is necessary to establish a criterion to decide when a node should initiate the discovery function and start searching for a new AP. Towards this, a parameter called *SNR_Scan_Threshold* is defined in DQCA. Each node continuously compares the SNR measurements obtained from the reception of each FBP with the value of this threshold. When an SNR measurement is detected to be lower than the value of the *SNR_Scan_Threshold*, the node initiates a discovery process. In order to set the value of this parameter, a tradeoff should be managed:

- 1) The threshold should be high enough to allow a node to initiate the discovery phase before losing the connectivity with the current AP.
- 2) A very high threshold may result in nodes initiating a scanning process too frequently. As a result, system efficiency would decrease as nodes would spend too much time seeking for other available resources instead of transmitting their information messages.

B. Discovery Function

The discovery function is initiated once a node has determined that it is necessary to scan for other available APs seeking for a better connection. The information regarding the quality of each neighbor AP is obtained through the LSM function described above.

One of the main constraints that have to be taken into account when designing the discovery process for a DQCA network is that nodes must keep updated some control state information (positions in the queues, queues size, etc.) of the current AP through the feedback information broadcast by the AP at the end of each frame. This means that any node must receive the FBPs from the current AP despite being scanning for new potential APs.

Accordingly, and assuming that all the nodes know all the possible frequency channels in the system, we establish the following procedure for the discovery function:

- 1) Once a node has determined that it must initiate the scanning process, it has to tune its transceiver to the next available frequency channel at the beginning of the next frame. Then, it waits for the reception of the FBP from the AP at this new frequency for a certain period of time denoted by *MAX_SCAN_TIME*. If the node receives an FBP, it measures and stores a certain selection metric (e.g. the received SNR from the FBP). Otherwise, the node may not receive any FBP due to the fact that either the scanning node is not within the radio range of any other AP or the FBP transmission from the scanned AP has not coincided in time with the scanning period of the node. In any of these two cases, the scanning node considers that no AP is available at this frequency and it discards the AP as a candidate to associate with.
- 2) After the scanning period, the node must reconnect to its current AP in order to receive the FBP packet and maintain the MAC status with the current AP. After receiving and decoding the FBP from its current AP, the node must tune its radio transceiver to the next frequency channel in the list of available channels and perform the same procedure described in point 1. The available channels in the system are scanned consecutively. For example, if the WLAN system is configured to operate on channels 1, 6, and 11 and a given node is currently associated with the AP operating on channel 6, it will start scanning for FBPs on channel 11. Then, after reconnecting to channel 6 for the reception of the FBP, it will continue scanning on channel 1, and so on.
- 3) Once all the available channels in the system have been scanned, the node selects the best AP to associate with. Note that to this end, a neighbor AP table should be maintained by the node during the discovery phase.
- 4) The node tunes its transceiver to the selected AP channel and, in case that the selected AP is different than the one the node was connected to, the node resets its queue pointers ($TQ=RQ=pTQ=pRQ=0$), while it waits for the first FBP from the new AP in order to obtain control information to properly adjust the queue pointers.

It is worth mentioning that the value of *MAX_SCAN_TIME* is critical as it sets the duration of the scanning period that a node spends at each frequency channel. As with the *SNR_Scan_Threshold*, a tradeoff must be managed:

- 1) The scanning period must be long enough so that a node can receive and demodulate a complete FBP packet from the scanned AP. The duration of the scanning period has to be at least equal to the duration of a FBP packet, while we have to take into account that the shorter the duration of the scanning process, the lower the probability that a FBP from a neighboring AP can be received.

2) The scanning period must be short enough to allow a node to reconnect to the channel at which it is currently connected once the scanning period has elapsed. Note that the longer the scanning period, the higher the probability that a scanning user detects the presence of an existing AP.

These lower and upper bounds for the value of MAX_SCAN_TIME can be formulated as

$$T_{FBP} \leq MAX_SCAN_TIME \leq (T_{frame} - T_{FBP}), \quad (1)$$

where T_{FBP} is the duration of a FBP and T_{frame} is the duration of a DQCA frame. In order to maximize the probability that an existing AP is detected by a node executing the discovery function, the scanning time should be maximized and thus should be set as close as possible to the upper bound expressed in (1). However, due to the adaptive rate capability of 802.11 systems, the duration of a DQCA frame is not constant. This means that a node in the discovery phase is not able to have any knowledge of the maximum periods of time that it can be scanning another frequency channel. In order to cope with this problem, a conservative approach can be adopted by setting the value of MAX_SCAN_TIME equal to the duration of the control minislots plus the minimum data slot duration. The minimum data slot duration can be computed using the highest available data rate with the shortest data packet. In addition, it is worth noting in the right condition of (1) that the duration of a FBP is subtracted from T_{frame} . This subtraction is necessary to ensure that a scanning node can receive and demodulate the complete FBP packet at the end of the frame.

Figure 3 shows graphically the conservative MAX_SCAN_TIME duration with respect to the DQCA frame duration.

Figure 4 shows an example of operation of the Discovery function where a node initiates the scanning process. Firstly, the node associated to AP1 scans at the frequency channel of AP2 for MAX_SCAN_TIME seconds and receives an FBP. Next, the node reconnects to AP1 channel frequency in order to receive the FBP necessary to maintain its DQCA state.

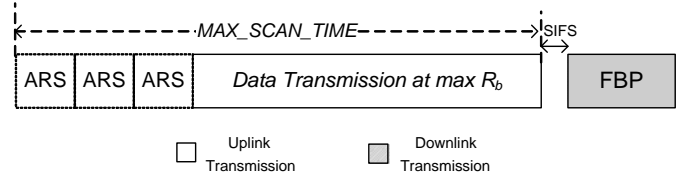


Figure 3 MAX_SCAN_TIME duration and DQCA Frame

Then, the scanning node tunes its transceiver to the AP3 channel frequency and receives an FBP. At this point the node has finished the scanning process and is able to choose the most appropriate AP to associate with. In this example, the node executing the discovery function receives FBPs in each scanning period. However, it may occur that the transmissions of the FBP from the scanned AP channels do not coincide temporarily with the interval of time during which the scanning node listens to that AP.

In addition, it has to be mentioned that this discovery process has some impact on the protocol rules of DQCA, which have to be slightly extended to consider the roaming between APs. Indeed, a node initiating a discovery phase can be in one of the following states (regarding the DQCA rules execution):

- The node has no data message to transmit and thus it has no position in either the DTQ or the CRQ ($pTQ=0$ and $pRQ=0$).
- The node has at least one data message to transmit but has not sent the corresponding ARS for the first message yet, thus has no position in either the DTQ or the CRQ ($pTQ=0$ and $pRQ=0$).
- The node has a position in the RTQ and a resolution of a previously collided ARS ($pTQ=0$, $pRQ>0$) is pending.
- The node has a position in the DTQ, but it is not currently transmitting data ($pTQ>1$, $pRQ=0$).
- The node is transmitting a message because it occupies the first position in the DTQ ($pTQ=1$, $pRQ=0$).

The first two cases (a) and (b) are transparent to the execution of the rules of DQCA. In case (c), if the node initiating the discovery phase has a position in the CRQ, it should postpone the contention until a new association to an AP is com-

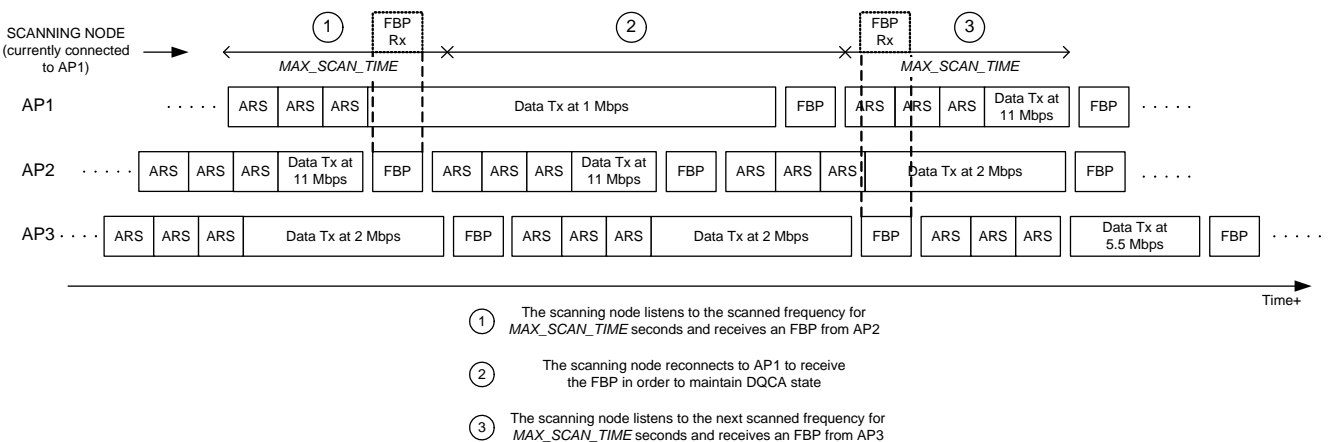


Figure 4 Example: Discovery process

pleted. To do so, it should reset the value of pRQ to zero and reinitiate the contention as soon as it is reassocated to another AP and the regular rules of DQCA can be applied.

Situations (d) and (e) can lead to degradation in the system performance due to the fact that they yield empty data frames, as it will be explained later. However, they can be easily tackled by smartly using the ‘final message bit’ attached to the FBP.

First, note that a scanning node in situation (d) that reaches the first position of the DTQ will not even transmit the first packet and some empty data frames will occur in the system, reducing the throughput of the network. In this case, the AP sets the ‘final message bit’ to one in the very first FBP upon the occurrence of an empty data slot. This allows the rest of nodes present in DTQ to advance one position in the queue. Therefore, the number of empty data slots is reduced to just one. Furthermore, this loss of efficiency can be significantly improved if the time interval that the AP needs to decide that a data slot is empty is minimized by defining a *data timeout* after which, if no data transmission has been sensed, the frame is finished with the immediate transmission of a FBP by the AP.

This mechanism can be also applied to solve situation (e). Recall that in this situation the node initiating the discovery process is already transmitting a data message. In this case, the node that initiates the scanning process aborts the data transmission and empty data slots occur in the system. Similarly to the previous case, the AP detects the situation and sets the ‘final message bit’ to one in the next FBP. The scanning node will reset its pTQ counter to zero to leave the DTQ. It is worth mentioning that, since the node will not have finished the transmission of the entire message, it should broadcast a request access for the pending part of the message. The reconstruction of a message received through different APs is out of the scope of the paper and constitutes an interesting line for future research.

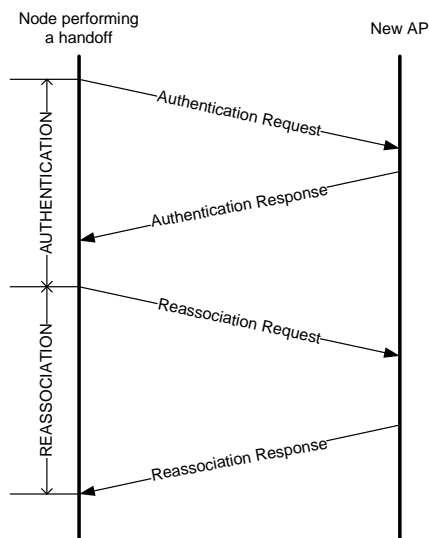


Figure 5 Reassociation Process

C. Reassociation Function

The reassociation function defines the message exchange devoted to authorizing and reassociating a node with an AP. As in 802.11, we define two procedures for the DQCA reassociation process: the authentication process and the reassociation process. These processes are both represented graphically in Figure 5.

Once the node has found an available AP and decides to join it, it initiates the authentication process. This process consists in exchanging information (proof of knowledge of a given password) between the AP and the node.

When the node is authenticated, the association process is started. This process consists in exchanging information regarding the node and AP capabilities. Once the association process is completed, the node is capable of transmitting and receiving data frames.

V. CONCLUSIONS

The handoff processes for DQCA in a multicell environment have been described in this paper. Specific functions, similar to those described in the 802.11 Standard, which take advantage of the specific frame format of DQCA have been defined. More precisely, we have analysed the operation of the link status monitoring, the discovery and the reassociation functions.

In future works we will provide some simulation results for the proposed handoff functions and we will propose advanced cross-layer based reassociation functions for DQCA.

ACKNOWLEDGMENTS

This work has been partially funded by the research Projects R2D2 (CP6-013), PERSEO (TEC2006-10459/TCM), NEWCOM++ (ICT-216715), CENTENO (TEC2008-06817-C02-02), and by the Generalitat de Catalunya under grant 2009-SGR-940.

REFERENCES

- [1] J. Alonso-Zárate, C. Verikoukis, E. Kartsakli, A. Cateura, and L. Alonso, “A Near-Optimum Cross-Layered Distributed Queuing Protocol for WLAN”, IEEE Wireless Communications Magazine Special Issue on Medium Access Control Protocols for Wireless LANs, vol. 15, no. 1, pp. 48-55, February 2008.
- [2] V. Mhatre and K. Papagiannaki. Using smart triggers for improved user performance in 802.11 wireless networks. In ACM Mobisys, June 2006.
- [3] Cisco Systems. Deployment guide: Cisco Aironet 1000 series lightweight access points.
- [4] “IEEE Standard for Information technology - 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), vol., no., pp.C1-1184, June 12, 2007.
- [5] G. M. Campbell and W. Xu, “Method and apparatus for detecting collisions on and controlling access to a transmission channel,” United States Patent 6292493, September 2001.
- [6] J. I. Capetanakis, “Tree Algorithm for Packet Broadcast Channels,” In IEEE Transactions on Information Theory, vol. 25, no. 5, pp. 505-515 September 1979.
- [7] J. I. Capetanakis, “Generalized TDMA: The Multi-Accessing Tree Protocol,” In IEEE Transactions on Communications, vol.27, no.10, pp. 1476-1484, October 1979.
- [8] B. S. Tsybakov and V. A. Mikhailov, “Free Synchronous Packet Access in a Broadcast Channel with Feedback,” In Probl. Information Transmission, vol. 14, no. 4, pp. 259-280, October-December 1978.