Medium Access Control Priority Mechanism for a DQMAN-Based Wireless Network


Abstract

The analytical evaluation of an access priority mechanism for a high-performance Medium Access Control (MAC) protocol, the Distributed Queuing MAC protocol for wireless Ad Hoc Networks (DQMAN), is presented in this letter. DQMAN comprises a hierarchical, dynamic and spontaneous master-slave clustering algorithm together with an embedded tree-splitting collision resolution algorithm based on access minislots. The responsibility of being master entails extra functionality, and thus, it must be shared in a dynamic manner among all the stations of the network. By allowing those stations acting as master stations to avoid contention to get access to the channel, their average packet transmission delay can be effectively reduced compared to that of slave stations. Consequently, stations may be encouraged to operate in master mode regardless of the extra functions they may have to carry out.

I. INTRODUCTION

There exist in the literature several Medium Access Control (MAC) protocols for wireless networks based on the use of blocked-access tree-splitting collision resolution algorithms. In those protocols, collisions are split into groups, and those groups are further split into new groups and so on, forming a tree of resolutions [1], [2]. A specific class of these algorithms confines the collisions to a dedicated time slot in order to avoid the waste of resources caused by the collision of data packets. Typically, the contention slot is further divided into access minislots wherein users contend for the channel. Therefore, collisions of data packets are completely avoided, and thus, the performance of the network is improved.

Among other examples, the Distributed Queuing Collision Avoidance (DQCA) protocol is based on the use of tree-splitting algorithms with access minislots [3]. This protocol is the extension of the Distributed Queuing Random Access Control Protocol (DQRAP) [4] protocol, designed for the distribution of Cable-TV, to operate in infrastructure-based Wireless Local Area Networks (WLANs). DQCA achieves a near-optimum use of the resources by running the collision resolution algorithm and the data transmissions in parallel (orthogonally in time).
In DQCA, time is divided into consecutive MAC frames. Each MAC frame consists of three parts: i) a contention window further split into $m$ access minislots, ii) a data part and iii) a control feedback part.

In order to gain channel access, a station with data to transmit randomly selects one out of the $m$ minislots (all of them with equal probability) and broadcasts a pseudo-random Access Request Sequence (ARS) at the beginning of the next frame. An ARS consists of a detectable sequence of bits that allows the Access Point (AP) to determine the state of each access minislot as:

1) *idle*, if no ARS has been sent in that minislot.
2) *successful*, if only one ARS has been sent in that minislot.
3) *collision*, if more than one ARS have been sent in that minislot.

The methodology to design the ARS and the mechanism to carry out the minislot detection are the subject of a patent by G. Campbell in [5]. Therefore, the outcome of any access request is broadcasted by the AP in a feedback packet at the end of each frame. It is worth emphasizing that the feedback packet only contains the information regarding the state of each of the $m$ access minislots.

With this information, stations execute the set of rules described in [3] that allow them to get self-organized into two distributed queues, namely, the Collision Resolution Queue (CRQ) and the Data Transmission Queue (DTQ). These queues are represented at each station by four simple integer numbers which correspond to the number of positions occupied in each queue and the current position of the station in each queue, respectively.

If a collision occurs, the involved stations enter the CRQ and resolve their collision in subsequent frames according to a blocked-access $m$-ary tree collision resolution algorithm. After a successful ARS, the corresponding station enters the DTQ. Any station at the head of the DTQ transmits its packet in the data part of the next frame.

As it was shown in [4], if $m \geq 3$, the collision resolution works faster than the data transmissions, and therefore, a near-optimum performance is attained.

In order to extend DQCA into the context of infrastructure-less wireless ad hoc networks, a hierarchical, dynamic and spontaneous master-slave clustering algorithm can be used at the MAC layer. The main idea is that whenever a station has data to transmit, it becomes the master of a spontaneous temporary cluster where a variation of DQCA can be executed. However, despite the virtual cluster structure, communications are done by establishing peer-to-peer links between source and destination within the cluster and thus, the master only acts as an indirect coordinator. This idea was analyzed in [6], where the Distributed Queuing MAC protocol for Ad Hoc Networks (DQMAN) was presented. Analytical evaluation of the protocol shows that DQMAN attains high-performance in infrastructureless networks.

However, further analysis of the protocol shows that, since the role of operating in master mode entails extra power consumption, it is necessary to give some incentive to stations to become masters when they have data to transmit and there is no other station already operating in master mode.

A simple but effective mechanism to encourage stations to become master when necessary consists in providing them with prioritized channel access when operating as masters so that their average transmission delay can be reduced.
Recall that a master station operates as a regular station to get access to the channel whenever it has data to transmit and, as a consequence, perceives the same average access delay than the rest of the stations. It is important to emphasize that the master has no control on the access to the channel but it only broadcasts the state of each of the $m$ access minislots.

Therefore, since a master station broadcasts the state of each of the $m$ access minislots at the end of each MAC frame, it can report a successful request in an empty minislot whenever it has data to transmit without actually sending any ARS. Therefore, contention is completely avoided and the average transmission delay is effectively reduced.

The analysis of the efficiency of this priorization mechanism is the main contribution of this letter.

To do so, the average time a master needs to find an empty minislot where to report a successful request is presented in the next section. This calculation allows comparing the access delay of masters with that of slave stations, which can be computed with the delay model presented in [7], within the context of DQRAP.

II. AVERAGE CONTENTION TIME FOR MASTERS

It is assumed that each MAC frame has $m$ access minislots. In each one of these minislots an ARS may be sent by any station needing to transmit a packet. Then, the average number of frames needed to find at least one free minislot (without including the frame with the empty one), denoted by $\mathbb{E}[T_{fm}]$, is calculated as

$$
\mathbb{E}[T_{fm}] = \sum_{k=0}^{\infty} k (1 - P_f)^{k-1} P_f = 1 - \frac{1}{P_f} - 1. \tag{1}
$$

$P_f$ is defined as the probability of finding at least one free access minislot in a given frame, and, according to the total probability theorem, it can be computed as

$$
P_f = \sum_{k=0}^{\infty} P_{f|k} P(k). \tag{2}
$$

$P(k)$ is the probability of having exactly $k$ arrivals (access requests) into the system in a given MAC frame. This value depends on the specific arrival distribution from all active stations in the network.

On the other hand, $P_{f|k}$ is defined as the probability of having at least one free access minislot given that there are $k$ arrivals into the system. This probability is 1 for $k < m$. Otherwise, it can be computed turning to combinatorics. Indeed, the $k$ arrivals can be parcelled out into any non-empty subset of minislots, being the subsets of interest all those that leave out, at least, one empty access minislots. Therefore, for any $k \geq m$, the probability $P_{f|k}$ can be computed as

$$
P_{f|k} = \frac{\sum_{j=1}^{m-1} j! \binom{m}{j} S(k,j)}{\sum_{j=1}^{m} j! \binom{m}{j} S(k,j)}. \tag{3}
$$
The terms $S(k, j)$ in (3) are the Stirling Numbers of the Second Kind \[8\]. They are defined, for any $k \geq j$, as the number of ways of partitioning a set of $k$ elements into $j$ non-empty subsets and can be calculated as follows:

$$S(k, j) = \frac{1}{j!} \sum_{i=0}^{j} (-1)^i \binom{i}{j} (j - i)^k.$$ (4)

This formula considers neither the order of the access minislots in which the arrivals are partitioned nor the order of the arrivals within each minislot. However, since the minislots are ordered in time, ordering must be taken into account, and a multiplying factor $j!$ has to be added to (4). In addition, the factor is necessary to consider all the possible combinations of selecting $j$ minislots out of $m$. These two additions have been already included in (3). The accuracy of (3) is shown in Fig. 1 where its value has been compared to that obtained by computer simulation as a function of the number of arrivals $k$ and for different number of access minislots.

Finally, using (3) and (4) into (2), and recalling (1), the average number of frames needed to find at least one free minislot (including the frame with the empty one) can be rewritten as

$$E[T_{fm}] = \frac{1}{P_f} - 1 = \left[ \sum_{k=0}^{m-1} P(k) + \sum_{k=m}^\infty \frac{m}{j} \sum_{i=0}^{j} (-1)^i \binom{j}{i} (j - i)^k P(k) \right]^{-1}.$$ (5)

This expression is used in the next section to assess the efficiency of the priority mechanism presented in this letter.

### III. Performance Evaluation

A configuration of DQMAN with $m = 3$ minislots is considered. A total of 10 stations, all of them within the transmission range of each other, generate traffic following a Poisson arrival distribution with aggregate rate $\lambda$ messages/frame. It is considered that all the stations contribute to the aggregate offered load with the same input rate. The transmission time of a message is exponentially distributed with average $1/\mu$, and thus, the system utilization factor, i.e. the probability that at least one station has data to transmit or is already transmitting data, is denoted by $\rho$ and can be defined as

$$\rho = \frac{\lambda}{\mu}.$$ (6)

The average transmission delay of DQMAN within a cluster can be accurately computed with the delay model presented in \[7\] within the context of DQRAP. However, in order to compute the average transmission delay of a station when it operates in master mode, the expression of the average contention time presented in \[7\], which is valid as it is for stations operating in slave mode, should be substituted by the expression in (5).

The reduction (expressed in percentage) of the average transmission delay of a station operating as master, with respect to the average transmission delay perceived by slaves is plotted in 2 as a function of the system utilization factor. Different curves have been represented for different average message lengths. Despite the existence of an
optimal system utilization factor at which the delay reduction is maximized, the average transmission delay of a station operating in master mode is lower than that of slaves in all cases. As it could be expected, the average message length, which corresponds to the average number of packets transmitted per successful access request, has a remarkable impact on the efficiency of the proposed prioritization technique. For short message lengths (1 or 2 packets per message), the technique is very efficient in reducing the average transmission delay of the master. For example, for an average length of one packet per message a master experiences an average transmission delay reduction of up to approximately 22%.

On the contrary, as the average message length grows, the contention time becomes smaller compared to the actual data transmission time, and thus, the benefits of the prioritization technique become less significant. However, since messages tend to be short in wireless communications in order to combat the channel impairments (fading, shadowing, path loss), the potential benefits that a master can achieve in terms of average transmission delay may payoff its extra energy consumption.

IV. Conclusions

The analytical evaluation of the reduced average transmission delay of stations operating in master mode in DQMAN-based wireless ad hoc networks has been presented in this letter. By allowing masters to avoid contention when getting access to the shared medium, their average packet transmission delay can be reduced in comparison to that of slaves. Consequently, stations are encouraged to get the energy-consuming role of being master for bounded periods of time. It is worth emphasizing that the proposed technique and the analysis presented in this letter could be applied to any MAC protocol based on access minislots.

References

Fig. 1. Probability of finding an empty minislot in a given frame

Fig. 2. Reduction of the average message transmission delay for masters compared to slaves