Analysis of a Distributed Queuing Medium Access Control Protocol for Cooperative ARQ

J. Alonso-Zárate*, L. Alonso†, Ch. Skianis‡, and C. Verikoukis*
*Centre Tecnològic de Telecomunicacions de Catalunya (CTTC), Castelldefels, Barcelona, Spain
{jesus.alonso, cveri}@cttc.es
†Dept. of Signal Theory and Communications
Universitat Politècnica de Catalunya (UPC), Castelldefels, Barcelona, Spain
luisg@tsc.upc.edu
‡University of the Aegean
cskianis@aegean.gr

Abstract—In this paper we analyze the performance of a distributed queuing medium access control (MAC) protocol designed to execute cooperative ARQ (C-ARQ) schemes at the MAC layer. Due to the broadcast nature of the wireless channel, a user that receives a data packet with unrecoverable errors can request retransmission from any of the other users in the transmission range of the transmitter that overheard the original transmission. These users can act as spontaneous relays and provide the communication with cooperative diversity gains. Upon retransmission request, the relays have to contend for the access to the channel. The DQCOOP protocol has been proposed in the literature as a high-performance MAC protocol for this kind of scenario. In this paper we theoretically evaluate its performance. The analytical results are supported by computer-based simulation that show the accuracy of the analysis.

I. INTRODUCTION

Cooperative transmission is an efficient technique to realize diversity gain in wireless fading channels in a distributed way. This cooperative gain can translate into higher transmission rates, lower transmission delays, more efficient power consumption, or even coverage extension [1].

The work we present in this paper is focused on a specific sort of cooperative communications: Cooperative Automatic Retransmission reQuest (C-ARQ) schemes. C-ARQ schemes exploit the broadcast nature of the wireless channel in the following manner: once a station receives a data packet with unrecoverable errors, it requests retransmissions from any of the users which overheard the original transmission and can act as helpers or relays. Eventually, the destination might either receive a correct copy of the original packet from a relay or be able to properly combine the different retransmissions from the relays to successfully decode the original packet. The fundamental concepts and theoretical bounds of C-ARQ schemes have been extensively studied in the past [2]–[4]. These works consider simplified network topologies with one transmitter, one receiver, and a single relay, or, when considering more than one relay, they assume ideal scheduling

strategies. However, in the case that more than one relay exists in the network, it is necessary to execute a distributed Medium Access Control (MAC) protocol to tackle with the contention among those stations willing to cooperate. Some MAC designs have been already presented in the literature to cope with this problem [5]–[8]. All these protocols are based on the Distributed Coordination Function (DCF) of the IEEE 802.11. This means that, while they might be simple to implement in actual hardware, they also inherit the same inefficiencies of the DCF which may compromise the benefits of C-ARQ schemes when the size of the contention window is not well adjusted to number of potential candidates to help. However, the knowledge of the number of possible helpers for a failed transmission is hardly predictable, and thus a protocol whose operation is independent of the number of contending station would be desirable. This was the main motivation for the design of the Distributed Queuing MAC Protocol for Cooperative Networks (DQCOOP) presented in [9]. DQCOOP is based on the near-optimum DQCA MAC protocol designed for the uplink channel of centralized WLANs [10], which in its turn, is based on DQRAP [11]. Simulation results reported in [9] show that DQCOOP outperforms any 802.11-based MAC protocol in terms of average packet transmission delay when cooperation is requested and, what is most important, its operation is almost independent of the number of relays and behaves as a perfect TDMA scheduler with some overhead. However, the work presented in [9] was only based on computer simulations. Therefore, there is a lack of an analytical model that allows evaluating the performance of the protocol. This is the main motivation for the work presented in this paper, whose main contribution is development of an analytical model that allows estimating the average delay of DQCOOP in C-ARQ scenarios. This estimation could be very useful for a user to decide upon whether initiating a C-ARQ upon the reception of a packet with errors, or discarding the packet for the benefit of the backlogged data.

The remainder of the paper is organized as follows. The C-ARQ model is described in Section II. Then, an overview of the DQCOOP protocol is presented in Section III. Section IV is devoted to the main contribution of this paper which is

the comprehensive analysis of the performance of DQCOOP. Simulation results are presented in Section V to validate the accuracy of the analysis. Finally, Section VI concludes the paper and gives some final remarks.

II. C-ARQ MODEL

The C-ARQ scheme under evaluation could be executed in any kind of network architecture, either centralized or distributed, as the players of the communication are: *i)* a source, *ii)* a destination, and *iii)* a number of active helpers or relays. Any message that is delivered to the MAC layer is fragmented into fixed length data packets that are passed to the PHY layer for transmission through the air interface. As in [12], and for the sake of simplicity, we focus on Stop and Wait ARQ mechanisms wherein the next packet is not processed until the current packet in transmission is either acknowledged by the intended destination or discarded for transmission. All the stations willing to cooperate listen to all ongoing transmissions. When a destination receives a data packet with unrecoverable errors, it initiates a cooperative phase by broadcasting a Call for Cooperation (CFC) packet. All those stations which decode both the original packet transmitted from the source station and the CFC packet become active relays and form a temporary cooperative cluster. These relays will persistently attempt to get access to the channel to forward the original failed packet transmission. Indeed, any retransmission scheme could be applied at this stage, either plain or hybrid ARQ. As discussed in [12], the reception of different retransmissions from the relays (which may be received through uncorrelated channels) can be mapped into an increase of the average Signal-to-Noise ratio (SNR) received at destination, and thus the destination station might be able to combine the different copies to decode the original packet. Whenever a destination station successfully decodes the original packet, an ACK packet is broadcast to inform the relays of the end of the current cooperation phase. A maximum cooperation time-out is defined so that the network does not fall into deadlock whenever a destination station cannot decode a packet.

According to this model, upon the reception of a CFC packet, all the active relays will try to get access to the radio channel simultaneously. Therefore, a multiple relay access control (MRAC) problem arises, as discussed in [13]. DQCOOP has been proposed as a high-performance protocol to solve this problem. An overview of its operation is presented in the next section.

III. DQCOOP OVERVIEW

The original description of DQCOOP can be found in [9] and, for the sake of completeness, we include a summary in this section. Two modes of operation are defined: *master* and *slave*. Any station should be able to operate in any of these states and change from one to another when necessary. Whenever a destination station receives a data packet with unrecoverable errors and initiates a cooperation phase, it becomes master. All the stations which successfully receive

and decode the CFC become slaves. A temporary cooperative cluster is then established and the protocol operation follows the frame structure illustrated in Fig. 1. It is composed of five parts, ordered chronologically:

1) **Part 1:** broadcast of the CFC by the master. A cooperation phase is initiated.
2) **Part 2:** an initial contention window composed of m_0 access minislots wherein every station willing to cooperate randomly selects (with equal probability) one out of the m_0 minislots where to send an Access Request Sequence (ARS). These ARS can be the simplest sequence of bits that can be detected by a station sensing the channel and allows distinguishing between the reception of just one ARS and the reception of more than one ARS simultaneously, i.e., a collision. The fact that they do not have to convey any information allows for allocating very short time slots for their transmission (no decoding has to be performed). A method for operating with these ARS is the subject of a patent [14].
3) **Part 3:** a FeedBack Packet (FBP) is broadcast by the master station with the feedback information regarding the state of each of the m_0 previous minislots. For each minislot, this information can have one out of three values: it can be empty (E), i.e., no ARS transmitted, success (S), i.e., exactly one ARS transmitted, or collision (C), i.e., more than one ARS transmitted in the same minislot (no matter how many).
4) **Part 4:** a number of consecutive frames containing: *i)* a contention window (further divided into m access minislots), *ii)* a data slot reserved for the transmission of data, and *iii)* a FBP broadcast by the master attaching information regarding the state of each of the minislots of the current frame. Note that the contention window of these frames has m minislots, where in general $m \leq m_0$, although this is not a mandatory condition.
5) **Part 5:** the transmission of a positive or negative ACK (ACK or NACK, respectively). Whenever the destination is able to successfully decode the original packet, it broadcasts an ACK packet indicating the end of the cooperation phase. A NACK is transmitted if the packet cannot be decoded at some point in time.

Short Inter Frame Spaces (SIFS) are left between each of the parts of the cooperation phase to compensate non-negligible propagation and data processing delays and turnaround times to switch the radio transceiver from receiving to transmitting mode.

Within the context of this newly defined MAC frame structure, the operation rules of DQCOOP are essentially the same as those defined in DQCA [10] and DQRAP [11], but with the modifications reported in [9]. Without diving into the details of the rules, the essential idea is described as follows. All those users willing to cooperate send an ARS in the initial contention window by selecting at random (with equal probability) one of the m_0 minislots. The first FBP contains the feedback information regarding the state of each of the m_0
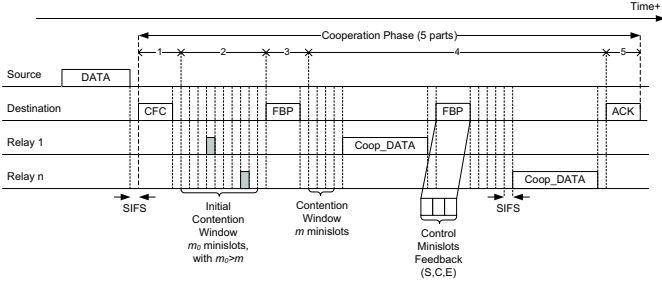
Fig. 1. DQCOOP Frame Structure

minislots. Those stations which succeed in the transmission are queued in the Data Transmission Queue (DTQ). This is a logical queue and it is characterized at each station by two integer numbers that define the total size of the queue and the individual position of the station within the queue. All the information required to update these two integer values is attached to the FBP broadcast by the master. Orderly in time, the stations in the DTQ transmit a data packet in the next frames (Part 4 of the DQCOOP operation). Since each position of the DTQ is allocated to a single station, data collisions are completely avoided. However, these collisions can occur within the initial contention window if two or more relays select the same minislot to transmit the ARS. All the stations whose ARS collide are queued in the Collision Resolution Queue (CRQ), another logical queue. The stations involved in a collision (those which selected the same minislot) are queued in the same position of the queue. Orderly in time, the stations in each position of the CRQ attempt to solve their collision by retransmitting an ARS in one of the m access minislots of the following frames. Therefore, a tree of resolutions is created. The key performance of DQCOOP is that data transmissions and the resolution of collision operate in parallel (orthogonally in time) and thus attain very high-performance. Note that there are no backoff periods and data collisions are completely avoided.

IV. ANALYSIS OF THE AVERAGE COOPERATION DELAY

The analysis presented in this section constitutes the main contribution of this paper.

A. Definitions

The **cooperation delay** is defined as the time elapsed **from** the moment a packet is firstly received with errors at destination, **until** it is either positively or negatively acknowledged to the transmitter after receiving an arbitrary number K of retransmissions from the relays.

An accurate estimation of the average value of this cooperation delay would allow stations to assess whether initiating a cooperative phase is worth the obtained benefit or not. Under some circumstances it might be more efficient (from the delay point of view) to request a retransmission from the source station or even to just discard the packet.

In MAC level terms, the cooperation delay can be defined as the time elapsed from the start of the transmission of the CFC packet until the end of the reception of the corresponding ACK/NACK packet transmitted by the destination. Therefore, the value of this delay is denoted by T_{COOP} and it can be written as

$$T_{COOP} = T_{CFC} + T_{SIFS} + T_{cont} + T_{SIFS} + T_{ACK}, \quad (1)$$

where T_{CFC} and T_{ACK} are the transmission times of CFC and ACK packets, respectively, and T_{SIFS} is the duration of a SIFS. T_{cont} is the contention time required to achieve an arbitrary number of K successful retransmissions which are needed to successfully decode the original packet without errors. The actual value of K depends on many parameters, and its study is not within the scope of this analysis. As reported in [13], the value of K mainly depends on the channel conditions between the source and both the destination and the relays and between the relays and the destination, on the retransmission scheme applied, and on the combination technique used at destination to combine the retransmissions.

Assuming a constant transmission rate, all the terms in (1) have deterministic values except for T_{cont}. Therefore, the average cooperation delay can be written as:

$$E[T_{COOP}] = T_{CFC} + T_{SIFS} + E[T_{cont}] + T_{SIFS} + T_{ACK}. \quad (2)$$

The value of $E[T_{cont}]$ depends on the number of relays, the MAC protocol used for their coordination, and the number of required retransmissions (K). Note that in a traditional non-cooperative ARQ scheme all the retransmissions are performed by the source, usually sequentially in time, and thus $E[T_{cont}] = KT_S$.

According to these definitions, the average packet transmission delay given that cooperation is executed is denoted by $E[T_D]$ and is defined as

$$E[T_D] = T_S + T_{SIFS} + E[T_{COOP}], \quad (3)$$

where T_S is the transmission time of the original (failed) transmission from the source, assumed to be constant.

B. DQCOOP Analysis

The value of $E[T_{cont}]$ within the context of DQCOOP can be expressed as

$$E[T_{cont}] = T_{ini\_frame} + (K + L)T_{frame}, \quad (4)$$

where T_{ini_frame} is the duration of the first frame with the initial contention window and can be computed as

$$T_{ini\_frame} = m_0 T_{mslot} + T_{SIFS} + T_{FBP}, \quad (5)$$

and T_{frame} is the duration of the $(K + L)$ subsequent frames where the retransmissions are performed and can be computed as

$$T_{frame} = 3T_{SIFS} + mT_{mslot} + T_{SR} + T_{FBP}. \quad (6)$$

L represents the number of frames without an actual data retransmission (empty frames) due to the fact that the contention among the relays has not been solved yet. We will elaborate further on this term later. T_{mslot} is the duration of an access minislot, and T_{SR} is the transmission time of each
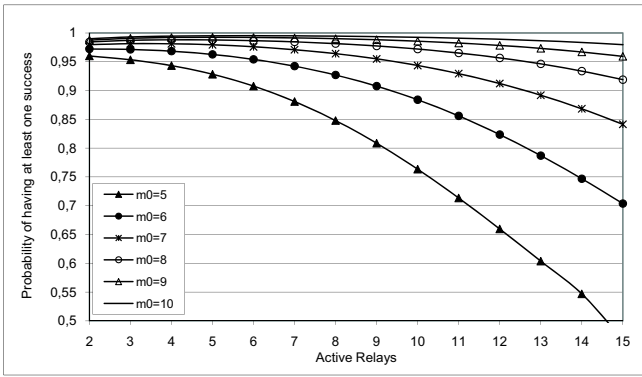
Fig. 2. Probability of having at least one success in the first frame

retransmission from a relay assuming a common and constant transmission rate for the retransmissions. This rate may be different from that of the source, and thus $T_S \neq T_{SR}$, since the relays might be selected with good channel conditions with the destination. Indeed, typically it may hold that $T_S > T_{SR}$.

The exact calculation of $\mathrm{E}[T_{cont}]$ in the context of DQ-COOP should consider the amount of empty data frames that may occur due to the heavy-traffic contention process that comes up upon cooperation request, i.e., the value of L. If there are no successful access requests in the first frames, there will be one or more empty data frames until at least one relay can be queued in the data transmission queue. However, the probability that there are no successful access requests in the first frame is negligible under some conditions and thus the computation of $\mathrm{E}[T_{cont}]$ can be approximated by

$$\mathrm{E}[T_{cont}] \approx T_{ini\_frame} + KT_{frame}, \qquad (7)$$

Note that the probability that just one out of a total of $(n+1)$ relays succeeds when requesting access to the channel in the first frame upon cooperation request is denoted by $P_{s|n}$ and can be computed as

$$P_{s|n} = m_0 \left(\frac{1}{m_0} \right) \left(1 - \frac{1}{m_0} \right)^n. \qquad (8)$$

Therefore, the probability that *at least* one relay is successful in the first frame can be computed as

$$P_{SK} = 1 - \left(1 - P_{s|n} \right)^{n+1} = 1 - \left[1 - \left(1 - \frac{1}{m_0} \right)^n \right]^{n+1}. \qquad (9)$$

The value of (9) is plotted in Fig. 2 as a function of the number of active relays $(n+1)$ for different values of m_0. As it can be seen in the figure, $P_{SK} \approx 1$ when $m_0 \geq 10$, regardless of the number of active relays. Therefore, the approximation in (7) is valid as long as the number of access minislot is high enough. Indeed, according to (9), it also holds that $P_{SK} \approx 1$ when $m_0 \approx (n+1)$, especially as the number of relays grows. This means that data retransmissions can start from the first data frame, emulating a near-perfect TDMA scheduling just with a small extra overhead.

Therefore, the average packet transmission delay when cooperation is requested in a C-ARQ scheme applying DQCOOP

is lower than that of performing retransmissions only from the source if either $T_S \gg T_{SR}$ or if $K_{SOURCE} < K_{RELAYS}$, being K_{SOURCE} the number of retransmissions required from the source and K_{RELAYS} the number of retransmissions required from the relays. Note that the former condition can be achieved if the active relays are those with better channel conditions with the destination, and the latter condition may be satisfied due to the independent transmission paths provided by the relays. In addition, the expression in (7) is independent of the number of active relays. This is a major characteristic of the protocol, especially for its application in C-ARQ schemes, since it alleviates the requirements of the relay selection algorithm.

V. MODEL VALIDATION

The analysis derived in the previous section is validated in this section through computer simulations based on MATLAB where the protocol rules are actually executed without any approximation.

A. Scenario

The values of the parameters used both to compute the value of $\mathrm{E}[T_{cont}]$ in (7) and to configure the simulations are summarized in Table I. These parameters have been set according to the standard values defined for the IEEE 802.11g [15]. In order to focus on the evaluation of the cooperation phases, a single-hop network wherein all the data transmissions from a fixed source to a fixed destination are received with errors is considered. That is, the destination always broadcasts a CFC packet upon the reception of every original data packet received from the source station. Moreover, the source has always a packet ready to be transmitted to the destination. In addition, it is assumed that a constant number of relays are activated within each cooperation phase and that the destination needs a constant number of retransmissions from the relay set to decode the original packet. Recall that the number of retransmissions is denoted by K.

B. Results

The difference between the value obtained with (2) to compute of the average packet transmission delay when cooperation is requested (using the approximation in (7)) and the simulated average packet transmission delay is illustrated in Fig. 3. The value of K has been set to 3, although other values of K have been also simulated leading to the same conclusions presented here. The plots have been omitted to avoid redundancy in the discussion. As it can be inferred from the figure, the worst case is for $m_0 = 3$, where the difference gets up to 9% for a total number of 15 active relays. However, for $m_0 \geq 7$ the difference between the model and the simulation is below 2%, being lower than 1% in all cases when $m_0 \geq 10$. In fact, as it will be further discussed later, this condition is also necessary to ensure the good performance of the protocol and to make it independent of the number of active relays. Therefore, the approximation in (7) provides a very simple equation that allows any station to properly estimate the

TABLE I

SIMULATION PARAMETERS

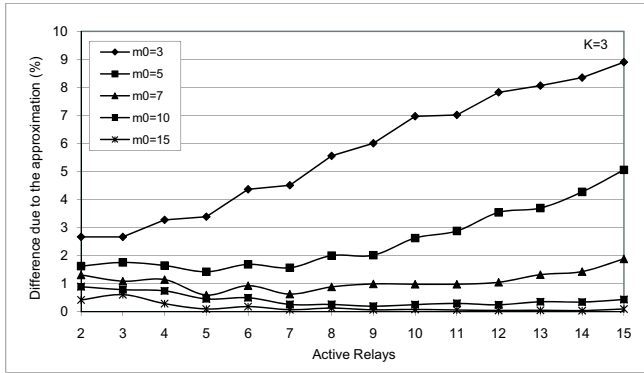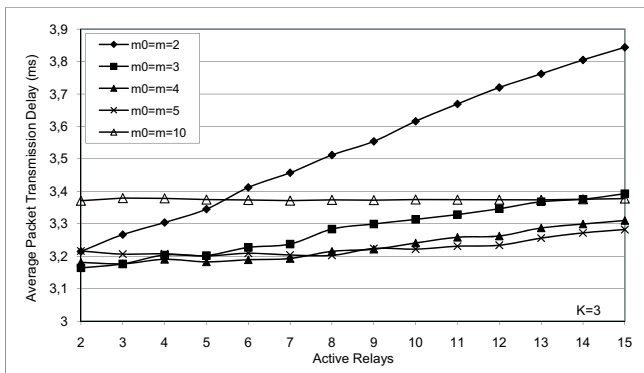| Parameter	Value	Parameter	Value
Data Tx. Rate Source	24 Mbps	Ctrl. Tx. Rate Source	6 Mbps
Data Tx. Rate Relays	54 Mbps	Ctrl. Tx. Rate Relays	6 Mbps
MAC header	34 bytes	PHY preamble	96 μs
ARS	10 μs	SIFS	10 μs
ACK length	14 bytes	Data packets	1500 bytes
FPB length	14 bytes	CFC length	14 bytes

Fig. 3.   Model validation

Fig. 4.   Performance of DQCOOP

average packet transmission delay if cooperation is initiated. This, in turn, allows a station to easily assess the suitability of actually initiating a cooperation phase upon error occurrence or not.

Although a comprehensive performance evaluation of DQ-COOP was already presented in [9], we have found interesting to include in this paper the key result of the protocol to demonstrate the accuracy of the model. The average data packet transmission delay for $K = 3$ is plotted in Fig. 4 for different number of active relays. It is worth seeing that the performance of DQCOOP is almost independent of the number of active relays given that the values of m and m_0 are sufficiently large. In the operational range of 1 to 15 relays, a value of $m_0 = 10$ ensures a performance almost independent of the number of relays. For any value $2 < m = m_0 < 10$, the performance is rather flat, although lower delays are attained for lower number of relays as the number of contending users is decreased.

VI. CONCLUSIONS

The DQCOOP protocol was presented in [9] as an extension and adaptation of DQRAP and DQCA to efficiently coordinate the contention among the relays in a C-ARQ scheme. In this paper, we have analyzed its performance and we have obtained a simple equation that allows estimating the expected data packet transmission delay if cooperation is executed. This estimation will allow a user to decide whether to initiate a co-operation phase or not when a packet is received with unrecov-erable errors. The model has corroborated the results obtained previously by simulation by which the number of minislots of the initial contention window can be tuned depending on the number of relays to ensure the optimal performance of the protocol. In addition, the derivations presented in this paper show that the performance of DQCOOP is almost independent of the number of active relays.

REFERENCES

[1] A. Nosratinia, T. E. Hunter, and A. Hedayat, "Cooperative communica-tions in wireless networks," *IEEE Communications Magazine*, pp. 74 – 80, Oct. 2004.

[2] E. Zimmermann, P. Herhold, and F. Fettweis, "On the performance of cooperative relaying protocols in wireless networks," *European Trans-actions on Communications*, vol. 16, pp. 5–16, Jan. 2005.

[3] P. Gupta, I. Cerruti, and A. Fumagalli, "Three transmission scheduling policies for a cooperative ARQ protocol in radio networks," in *Proc. of the WNGG*, Oct. 2004.

[4] I. Cerruti, A. Fumagalli, and P. Gupta, "Delay model of single-relay cooperative ARQ protocols in slotted radio network with poisson frame arrivals," *IEEE/ACM Transactions on Networking*, vol. 16, pp. 371–382, Apr. 2008.

[5] X. Wang and C. Yang, "A mac protocol supporting cooperative diversity for distributed wireless ad hoc networks,," in *Proc. of the PIMRC*, Berlin, Germnay, Sept. 2005.

[6] J. Alonso-Zárate, E. Kartsakli, C. Verikoukis, and L. Alonso, "Persistent RCSMA: A MAC protocol for a distributed cooperative ARQ scheme in wireless networks," *EURASIP Journal on Advanced Signal Processing, Special Issue on Wireless Cooperative Networks*, p. 13, Dec. 2008.

[7] S. Shankar, C.-T. Chou, and M. Ghosh, "Cooperative communication MAC (CMAC) - A new MAC protocol for next generation wireless LANs," in *Proc. of the International Conference in Wireless Networks, Communications and Mobile Computing*, 2005.

[8] N. Agarwal, D. ChanneGowda, L. N. Kannan, M. Tacca, and A. Fu-magalli, "IEEE 802.11b cooperative protocols: a performance study," in *Proc. of the IFIP/TX6 NETWORKIG 2007*, vol. 4479, Atlanta, GA, USA, May 2007, pp. 415–426.

[9] J. Alonso-Zárate, E. Kartsakli, C. Verikoukis, and L. Alonso, "A novel near-optimum medium access control protocol for a distributed cooperative ARQ scheme in wireless networks," in *Proc. of the PIMRC*, Cannes, France, Sep. 2008.

[10] L. Alonso, R. Agusti, and O. Sallent, "A near-optimum MAC protocol based on the distributed queueing random access protocol (DQRAP) for a CDMA mobile communication system," *Selected Areas in Communi-cations, IEEE Journal on*, vol. 18, no. 9, pp. 1701 –1718, Sep. 2000.

[11] W. Xu and G. Campbell, "A near perfect stable random access protocol for a broadcast channel," in *Proc. of the ICC*, 1992, pp. 370–374.

[12] M. Dianati, X. Ling, K. Naik, and X. Shen, "A node-cooperative ARQ scheme for wireless ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 46, pp. 1032–1044, May 2006.

[13] J. Alonso-Zárate, J. Gomez, C. Verikoukis, L. Alonso, and A. Pérez-Neira, "Performance evaluation of a cooperative scheme for wireless networks,," in *Proc. of the PIMRC*, Helsinki, Finland, Sep. 2006.

[14] G. Campbell *et al.*, "Method and apparatus for detecting collisions and controlling access to a communications channel," U.S. Patent US6 408 009 B1, 2002.

[15] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Standard 802.11-2007.