This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2010 proceedings

Fairness and convergence of CSMA with Enhanced Collision Avoidance (ECA)

Jaume Barcelo
Universitat Pompeu Fabra
jaume.barcelo@upf.edu

Alberto Lopez Toledo
Telefonica Research
alopezt@tid.es

Cristina Cano
Universitat Pompeu Fabra
cristina.cano@upf.edu

Miquel Oliver
Universitat Pompeu Fabra
miquel.oliver@upf.edu

***Abstract*—This paper presents CSMA/ECA, which combines the efficiency of reservation-based protocols and the simplicity of random access mechanisms. The maximum efficiency of CSMA/CA with optimal parameter adjustment is easily exceeded by CSMA/ECA, even when fixed parameters are used by the latter. CSMA/ECA stations fairly coexist with legacy CSMA/CA and increase the portion of time that is devoted to successful transmissions while decreasing the number of collisions and empty slots. The proposed mechanism initially behaves as a CSMA/CA network, but it progressively converges to a collision-free deterministic operation. The convergence process can be modelled as a Markov Chain to assess the duration of the transitory phase.**

I. Introduction

The key property of Carrier Sense Multiple Access (CSMA) networks, such as the one used in IEEE 802.11 [1], is that the stations listen before transmitting. A station with data to send senses the channel for a given amount of time and, if the channel is idle, the station transmits. If a collision occurs, the station defers the transmission for a random number of slots. The efforts to reduce the number of collisions are motivated by the fact that collisions represent a significant waste of resources in wireless networks. Since stations can only transmit or receive at any point in time, it is not possible to immediately detect a collision and interrupt the transmission.

Despite the possibility of collisions, CSMA/CA is still an appealing protocol for WLANs. It is lightweight, it takes advantage of statistical multiplexing to accommodate bursty traffic and it can be executed in a distributed fashion. CSMA/CA is especially fitted for networks with a large number of stations that sporadically send packets. However, CSMA/CA was not designed to benefit from the fact that some stations have multiple-packet messages [2], [3], *i.e.* stations that store several packets in their transmission queues.

When stations send multiple consecutive packets, it is possible to use the feedback obtained from previous transmissions to adequately schedule future transmissions. For this reason, we propose CSMA with Enhanced Collision Avoidance (CSMA/ECA), a modification to the CSMA/CA protocol that further reduces the number of collisions while maintaining all its versatility and power.

In this paper we will show how CSMA/ECA can achieve collision-free medium access after a transitory phase, and hence exceed the maximum theoretical performance of CSMA/CA. CSMA/ECA is totally distributed and it is robust to the channel characteristics, and can fairly coexist with legacy CSMA/CA stations. Finally, CSMA/ECA is easy to implement with no additional computational costs.

The rest of the paper is organized as follows: Section II defines the CSMA/ECA algorithm, then in Section III a Markov Chain model to predict the length of the transitory phase is described. Implementation issues and the performance evaluation results are discussed in Section IV while Section V presents an overview of the related work in the area. Finally, some conclusions are given.

II. Enhanced Collision Avoidance

In CSMA/CA the channel time is implicitly divided into slots. Three different kinds of slots are differentiated: *empty* when no station attempts transmission; *successful* if one (and only one) station transmits; and *collision* if more than one station simultaneously transmit. The channel time spent in empty slots or collision slots is wasted.

Whenever a collision occurs, the station chooses a random backoff value B from a contention window of size CW, i.e.,

$$B \sim \mathcal{U}[0, CW-1], \tag{1}$$

where $\mathcal{U}$ is the uniform distribution.

For the rest of the paper, we consider that the stations always have a packet ready to transmit (the are in saturation mode). As a consequence, the stations are either transmitting, receiving or backing off; they are never idle. In CSMA/CA, the stations have to backoff both after collisions and successful transmissions. For collisions, the backoff has to be necessarily random to prevent a new collision in the retransmission attempt. However, for the second case, we will show that the backoff value can be deterministically selected.

A. Deterministic Backoff After Successful Transmissions

By choosing a deterministic backoff after a successful transmission and a random backoff otherwise, the system converges to a collision-free operation when the number of active stations is not greater than the value of the deterministic backoff. In the case of a successful transmission, the deterministic behaviour stabilizes the system. Conversely, if there is a collision, the randomness of the backoff provides a change that would (desirably) avoid more collisions.

CSMA/ECA exploits the information gathered from previous transmission attempts to further reduce the collisions, by

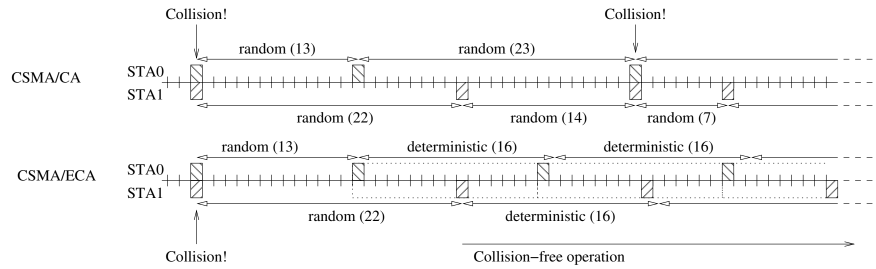

Fig. 1. CSMA/CA is compared to CSMA/ECA in an example in which two saturated stations contend for the channel. When CSMA/ECA is used, after both stations have successfully transmitted, the behaviour of the stations is deterministic and no more collisions occur.

performing a random search to find free slots until collisions disappear. Then the station keeps using a deterministic backoff after each successful transmission, until a collision occurs and the station moves back to the random behaviour. Note that this collision have to necessarily be caused by a station using random backoffs, since collisions among stations that behave deterministically are not possible.

Consider the case depicted in Fig. 1, where (*STA 0* and *STA 1*) contend for the channel. For simplicity, all the slots are represented with equal sizes. Also the transmission attempts are represented as shaded boxes. The backoff value chosen by each station is shown in brackets[1]. The upper channel time line corresponds to legacy CSMA/CA operation, while the lower one shows our proposed CSMA/ECA operation.

In this example, the CSMA/CA stations collide, successfully transmit and eventually collide again. When CSMA/ECA is used, collisions disappear after all stations have successfully transmitted, because the backoff is selected deterministically. It is useful to imagine a virtual frame[2] of V slots (represented with a dotted line in the figure) and observe that after a successful transmission, the stations transmit in fixed slot positions within the virtual frame in a TDMA fashion.

Algorithm 1 shows the protocol that is distributedly executed in each of the contending stations, where b is the backoff counter, CW_{min} and CW_{max} are the minimum and maximum contention windows respectively, a is the number of transmission attempts, A is the maximum number of transmission attempts, and V is the deterministic backoff value.

```
b ← U[0, CW_min − 1];
while there is a packet to transmit do
    a ← 0 ;
    while a < A do
        while b > 0 do
            wait 1 slot ;
            b ← b − 1 ;
        Attempt transmission ;
        if success then
            /* Deterministic backoff.      */
            b ← V;
            break ;
        else
            a ← a + 1 ;
            /* fall to random backoff.     */
            b ← U[0, min(CW_min * 2^a, CW_max) − 1];
```

Algorithm 1: CSMA/ECA

B. Efficiency of CSMA/ECA during Steady-State Operation

Let us define the channel efficiency (ϕ) as the fraction of channel time that is devoted to successful transmissions,

$$\phi = \frac{P_s T_s}{P_e T_e + P_s T_s + P_c T_c}, \tag{2}$$

where P_e, P_s and P_c are the empty, success and collision probabilities, respectively. And T_e, T_s and T_c are the duration of an empty, successful and collision slot, respectively.

Then, for a number of contending stations (ς) not greater than the size of the virtual frame, the efficiency that can be obtained from CSMA/ECA in steady-state collision-free operation is:

$$\phi = \frac{\varsigma \cdot T_s}{\varsigma \cdot T_s + (V - \varsigma) \cdot T_e} \; ; \; \varsigma \leq V. \tag{3}$$

Fig. 2 shows the efficiency obtained with CSMA/ECA compared to CSMA/CA. The CSMA/ECA values include both transitory and stationary operation with 95% confidence interval. The IEEE 802.11b standard has been assumed, together with a data rate of 2 Mbps and a packet size equal to 1500 bytes. The efficiency is represented for an increasing number of *contending* stations (with data to send). The efficiency of CSMA/ECA is computed as presented in (3) using $V = 16$, the upper bound for CSMA/CA with dynamic parameter adjustment is obtained from [4], and the performance of CSMA/CA is obtained as in [5]. Simulations were performed using MAC layer custom simulator implemented in Octave.

Before achieving steady-state, a CSMA/ECA system goes through a transitory operation with and efficiency between that of CSMA/CA and the efficiency in (3). During this transitory phase only a fraction of the collisions is avoided, and the number of stations that successfully transmit (and thus use a deterministic backoff) is a random variable characterized next.

[1] The deterministic backoff after successes is a value that depends on the 802.11 flavor (see Section IV-A). In the example, a value of 16 is used.

[2] Some works refer to data-link layer PDUs as frames. In this article, a frame is a group of slots. Data-link layer PDUs are called packets.

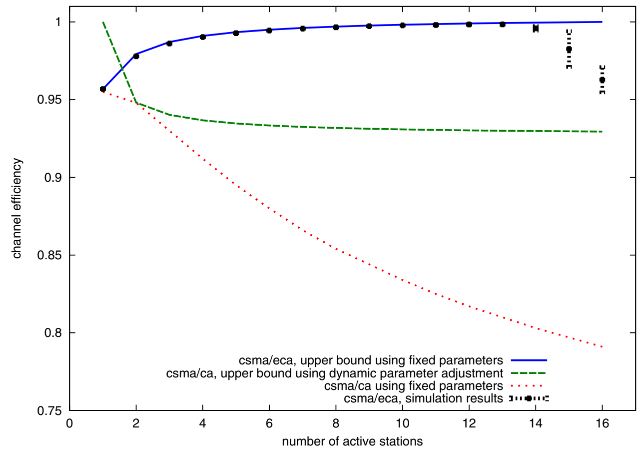

Fig. 2. The performance of CSMA/ECA with fixed parameters is compared to CSMA/CA with fixed and dynamic parameters. Simulations results are provided for CSMA/ECA.

III. A Dissection of the Convergence Process

Consider a scenario with ς saturated stations and a virtual frame size of V slots, $2 \leq \varsigma \leq V$. We will assume that the transition process occurs in a frame-by-frame basis[3]. Let X_n be the random variable that represents the number of stations that successfully transmitted in the frame n. Then we can model the transition process as a time-homogeneous Markov Chain whose state space is

$$\mathbf{S} = \{S_i | 0 \leq i \leq \varsigma\}, \tag{4}$$

with initial state S_0 and a stable state S_ς.

We are interested in computing the transition probability matrix $\mathbf{P}$ which is the matrix of one step transition probabilities $p_{i,j}$ defined by [4]

$$p_{i,j} = Pr(X_{n+1} = j | X_n = i) \; ; \; 0 \leq i, j \leq \varsigma. \tag{5}$$

In order to compute $p_{i,j}$ we will need the following results.

Claim 1: The system is stable when $X_n = \varsigma$, i.e. state S_ς is absorbing.

$$Pr(X_{n+1} = \varsigma | X_n = \varsigma) = 1. \tag{6}$$

Proof: $X_n = \varsigma$ implies that all the stations successfully transmitted in virtual frame n. Therefore, they all will deterministically choose the same transmission slot in virtual frame $n+1$ as they did in virtual frame n. As there were no collisions in frame n, there will be no collisions in frame $n+1$. ■

Claim 2: It is not possible that there is one and only one station that randomly selects the transmission slot in a given virtual frame.

$$Pr(X_n = \varsigma - 1) = 0 \; ; \; n > 0. \tag{7}$$

[3]This is an approximation, as the CSMA/ECA algorithm allows that the same station re-attempts transmission (and eventually succeeds) in the same virtual frame. Also, for ease of analysis, we don't use exponential growing backoffs for the stations after subsequent collisions, although the analysis can be easily extended to include it.

[4]Note that we index the rows of the matrix from 0 to ς. This is for consistency with the numbering of the states of the Markov Chain.

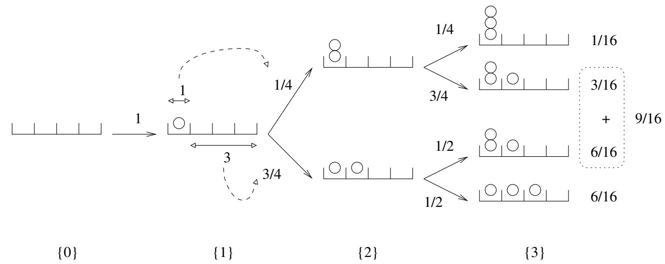

Fig. 3. A tree is used to evaluate the different outcomes that are possible in a system with $\varsigma = 3$ and $V = 4$.

Proof: By contradiction, let us assume that there is only one station that randomly selects the transmission slot in virtual frame n. However, this is not possible, since a collision occurs when a minimum of two stations transmit in the same slot, there must be at least two stations that will randomly select the transmission slot in virtual frame n. ■

A. Computing the Transition Probability Matrix $p_{i,j}$

Consider the state S_i, with i stations that deterministically transmit in i different slots, while the rest of the stations $(\varsigma - i)$ randomly transmit in any of the V slots. Note that for the special case $i = 0$, the problem is reduced to the computation of the number of successes that are obtained when ς stations transmit in V slots and can be solved using the model suggested in [6]. For any other value of i $(i \neq 0)$, the approach in [6] is no longer applicable, since it assumes that there are slots reserved for the stations that successfully transmitted in the previous frame.

For large values of V and ς, the brute force approach is computationally impractical. However, as we are only interested in the number of stations, the specific slots at which they transmit are equivalent, and the same holds for the specific stations. With this in mind, we propose the following method to compute $\mathbf{P}$. Assume that the previous state is S_0 and we want to compute the probabilities $p_{0,j}$ for all values $0 \leq j \leq \varsigma$. Now consider a transmission in the current frame. This transmission can be in any of the V (for now, empty) slots. Since all these slots are empty, the V possible outcomes are equivalent for our analysis. Each of the V outcomes consists of a slot with one transmission and $V - 1$ empty slots.

Following the same reasoning, for a second transmission in the same virtual frame, there are only two outcomes: *a)* the transmission slot is the same as the one for the first transmission (with probability $1/V$) or *b)* the two transmissions are in different slots (with probability $(V-1)/V$). These steps can be repeated to build a tree (of $\varsigma+1$ levels) with all the possible outcomes of interest and their associated probabilities.

Fig. 3 depicts the procedure for $\varsigma = 3$ and $V = 4$. The root represents the $V = 4$ empty slots, and in every level, a new transmission (circle) is included. The levels are labeled from $\{0\}$ to $\{3\}$. The edges of the tree are labeled with their probability. At the first level, there is only one node, since the only possible state is one success and three empty slots. Therefore, the edge from the root to the node at the first level is labeled with probability 1. In the transition from level $\{1\}$ to level $\{2\}$ there are two possible outcomes: *a)* the two transmissions occur in the same slot (with probability $1/4$) and *b)* the transmissions occur in different slots (with probability $3/4$). This process is iterated until all the transmissions are included. The probability of each leaf is computed by following the path from the root to that leaf.

From the tree it can be observed that the transition probability of having zero successes in frame $n+1$ given the fact that there were zero successes in the frame n, is

$$p_{0,0} = Pr(X_{n+1} = 0 | X_n = 0) = \frac{1}{16}. \tag{8}$$

Similarly, it is easy to compute $p_{0,1} = \frac{3}{16} + \frac{6}{16} = \frac{9}{16}$, $p_{0,2} = 0$ and $p_{0,3} = \frac{6}{16}$, which completes the first row of the transition matrix $\mathbf{P}$. In order to obtain the values for the second row, we assume that there was a successful transmission in the previous virtual frame. Therefore, we consider only the subtree with the root at the node of level $\{1\}$. To compute the third row of the matrix we use as a root the lower node of level $\{2\}$. The last row is computed using only one node, which is the lowest leaf. The transition matrix which is obtained[5] for this example is:

$$\mathbf{P}_{\varsigma=3,V=4} = \begin{pmatrix} \frac{1}{16} & \frac{9}{16} & 0 & \frac{6}{16} \\ \frac{1}{16} & \frac{9}{16} & 0 & \frac{6}{16} \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{9}$$

It is not a coincidence that the first two rows of $\mathbf{P}$ are equal.

Claim 3: The first two rows of the transition probability matrix $\mathbf{P}$ are equal.

$$p_{0,j} = p_{1,j} \; ; \; 0 \leq j \leq \varsigma \tag{10}$$

Proof: Consider a tree as the one in Fig. 3. It is easy to see that the subtree with the node of level $\{1\}$ as a root is identical to the one of level $\{0\}$ multiplied by 1. ■

B. Convergence Time

We are interested in evaluating how long does it take for the system to leave the transitory phase and converge to the collision-free operation. Let us consider an initial state S_0 in which all the stations randomly choose their transmission slot and then we use the transition matrix $\mathbf{P}$ to evaluate the marginal distributions in subsequent frames. Let

$$\boldsymbol{\pi}_n = \{Pr(X_n = i), 0 \leq i \leq \varsigma\} \tag{11}$$

be the vector of the marginal probabilities at stage n, and $\boldsymbol{\pi}_0 = [1, 0, ..., 0]$ the initial vector. This means that the initial state is S_0 with probability 1. Then the vector $\boldsymbol{\pi_n}$ can be obtained by:

$$\boldsymbol{\pi}_n = \boldsymbol{\pi}_0 \mathbf{P}^n. \tag{12}$$

The last component of vector $\boldsymbol{\pi}_n$ is precisely the value of interest for our study $Pr(X_n = \varsigma)$, which is the probability

[5]A script in Octave to compute the transition matrix for any value of V and ς is available upon request.

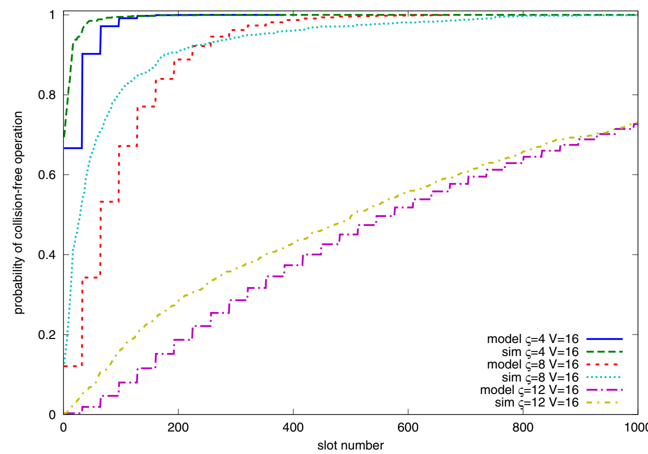

Fig. 4. The transition curves obtained using the model and simulation are compared for a value of $V = 16$ and various values of ς.

that the system has reached the stable collision-free state. One particularity of our evaluation of the transition curve is that we have considered that the transition step contains $2 * V$ slots *i.e.* two virtual frames. This is an approximation of the expected backoff of those stations that suffered a collision. We are implicitly assuming that the probability that the same station suffers multiple successive collisions is low, which is true for low values of ς. Note, however, that as the value of ς approaches the value of V, the assumption is no longer valid.

Fig. 4 plots the probability that the system has reached the collision-free operation in a given slot. The results obtained from the model are compared to those obtained from simulation. It can be observed that the transition process is slower for higher values of ς.

C. Disruption of the Stationary Operation by New Entrants

A new station can disrupt the stationary operation of CSMA/ECA[6] Note that we can assess the recovery curves associated with the event by using an initial state of $S_{\varsigma-1}$, representing the event that all stations but one are using a deterministic backoff. The initial vector is: $\boldsymbol{\pi}_0^D = [0, ..., 0, 1, 0]$. And the marginal probabilities of subsequent steps:

$$\boldsymbol{\pi}_n^D = \boldsymbol{\pi}_0^D \mathbf{P}^n. \tag{13}$$

Given that the current state is S_i, the maximum number of collisions in the previous step is:

$$\kappa_i \approx \lfloor \frac{\varsigma - i}{2} \rfloor, \tag{14}$$

where $\lfloor \cdot \rfloor$ is the floor operator. Then, using the approximation $T_c \approx T_s$, the efficiency of the system in the step $n-1$ is:

$$\phi_{n-1} \approx \sum_{i=0}^{\varsigma} \frac{2 \cdot i \cdot T_s}{(2 \cdot i + \kappa_i) \cdot T_s + (2 \cdot V - 2 \cdot i - \kappa_i) \cdot T_e} \pi_n^D(i), \tag{15}$$

[6]Channel errors can also disrupt the stationarity of CSMA/ECA, however their impact is less of those of new stations, and the analysis is similar to the one for new stations.

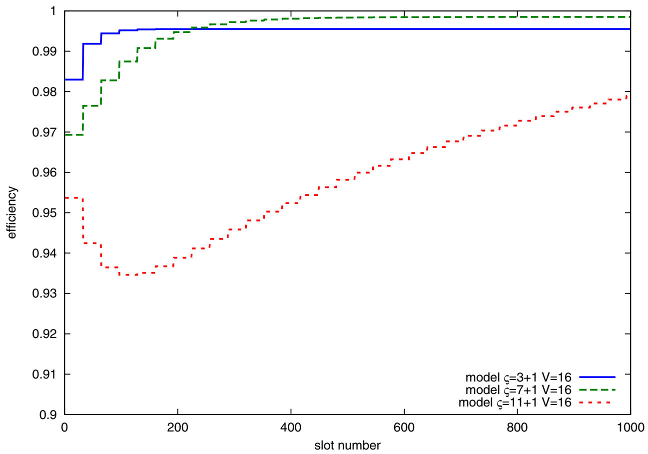

Fig. 5. Recovery curves after a channel error or new entrant.

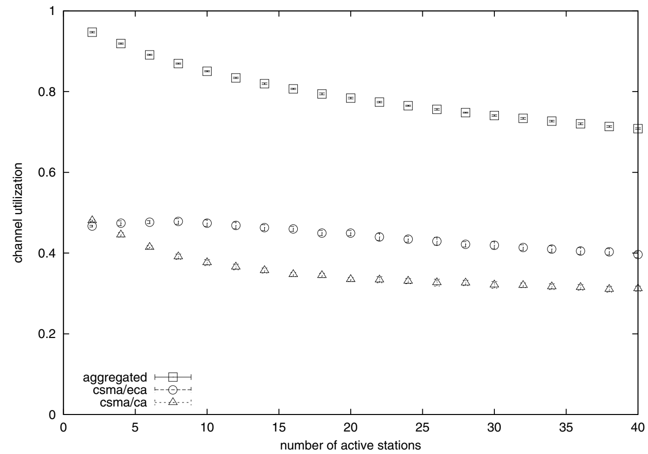

Fig. 6. Half of the stations run CSMA/ECA, while the other half run CSMA/CA. The figure shows the channel utilization achieved by each group.

where the expectation of the backoff of those stations that suffer collisions is considered to be twice as much as V.

Fig. 5 shows the recovery curves obtained from (13)-(15). The transitory phase associated with new incorporations to the contention can be avoided using Smart Entry, as we will see in Subsection IV-B.

IV. Implementation Aspects

A. Coexistence with legacy CSMA/CA stations

Any new version of the medium access control algorithm should be backward compatible with the already existing equipment. Furthermore, to guarantee the smooth coexistence of new and legacy stations, those stations running CSMA/ECA should consume a fair amount of the available bandwith.

The only difference between CSMA/CA and CSMA/ECA can be found in line 11 of Algorithm 1 . CSMA/CA randomly chooses the backoff value from the minimum contention window ($b \leftarrow \mathcal{U}[0, CW_{min}-1]$), while CSMA/ECA deterministically ($b \leftarrow V$). In order to fairly compete with legacy stations, it is desired that

$$V = \lceil E\left[\mathcal{U}[0, CW_{min}-1]\right]\rceil, \qquad (16)$$

where $E[\cdot]$ represents the expectation operator and $\lceil\cdot\rceil$ is the ceiling operator. This selection of the virtual frame size guarantees that the expected number of slots that a station waits after a successful transmissions is approximately the same for both CSMA/CA and CSMA/ECA.

To validate this, we performed simulations for a scenario in which half of of the stations run CSMA/CA while the other half use CSMA/ECA, and the obtained efficiencies are shown in Fig. 6 . The values chosen for the MAC parameters are $CW_{min} = 32$ and $V = 16$. The rest of the parameters are taken from the IEEE 802.11b specification. Each simulation runs for 10000 slots and each scenario is repeated ten times. The number of competing stations range from two to forty.

It can be observed that CSMA/ECA flows obtain higher channel utilization than CSMA/CA flows thanks to the reduced collision probability. This small advantage can be seen as an incentive for legacy stations to switch to CSMA/ECA. We obtained a Jain's fair index [7] higher than 0.98 when comparing the channel utilization of CSMA/ECA and CSMA/CA in a mixed scenario.

The benefits of using CSMA/ECA are greatly diminished in the presence of legacy stations since the collision-free operation is never reached. Nevertheless, a network running a mixture of CSMA/CA and CSMA/ECA stations will offer equal or better performance that a pure CSMA/CA network, since some of the collisions will be avoided.

To assess the benefits of using CSMA/ECA we simulate three different scenarios, namely, a pure CSMA/ECA, a mixed CSMA/ECA and CSMA/CA and a pure CSMA/CA. The results are compared in Fig. 7. It can be observed that, thanks to the enhanced collision avoidance mechanism, a larger fraction of the channel time is devoted to successful transmissions when only CSMA/ECA is used. For a number of active stations up to the size of the virtual frame size V, the efficiency is almost 1.

B. Smart Entry

We have seen in Section III-C that when a station joins the channel, it selects the first transmission slot randomly, posing the collision-free mode of operation of the system at risk. To avoid this situation, the stations that are not actively contending for the channel should keep track of the empty slots in each virtual frame. When one of those stations receives a packet from the upper layer, it already knows which slots are expected to be empty, and can schedule the first transmission accordingly.

If Smart Entry is used, the first line of Algorithm 1 is substituted by Algorithm 2. It includes an array called `slotNumber[]` to keep track of the status of each slot of the frame. The size of this array is the size of the virtual frame V. With the modification presented in Algorithm 2, a station joining the contention transmits in the first empty slot.

Note that while the station is delaying the first transmission attempt, it marks the positions in the array as free. This

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2010 proceedings

```
for i ← 0 to V − 1 do
    slotNumber[i] ← unknown ;
i ← 0 ;
while True do
    if there is a packet ready to transmit then
        if slotNumber[i] is free then
            transmit ;
            break ;
        else
            wait 1 slot ;
            slotNumber[i] ← free ;
    else
        wait 1 slot ;
        if channel sensed busy then
            slotNumber[i] ← busy ;
        else
            slotNumber[i] ← free ;
    i ← (i + 1) (mod V) ;
```

Algorithm 2: Smart Entry

behaviour prevents a deadlock in the case in which all the slots are busy. If there are no free slots, the station will delay its transmission attempt V slots, and then deliberately prompt a collision in order to free some slots for a future transmission attempt.

V. Related Work

In [4] it is shown that there is a fundamental limit on the efficiency of completely random access protocols, in which the transmission slot is chosen without using any prior information. Then, it is explained that CSMA/ECA can overcome that limit by using a random behaviour after failures (to trigger a change) and a deterministic behaviour after successes (to stabilize the system).

In [8], simulations are used to assess the performance of CSMA/ECA in saturated, non-saturated and hybrid (a combination of saturated and non-saturated) scenarios. CSMA/ECA is shown to perform equal or better than CSMA/CA in all the considered scenarios. Specifically, the two protocols deliver the same throughput in those scenarios in which the network is able to absorb all the offered traffic. However, when the traffic load overwhelms the network, CSMA/ECA performs better than CSMA/CA. Traffic prioritization in CSMA/ECA is addressed in [9]. An analytical model to capture the behaviour of CSMA/ECA in stationary operation for both rigid and elastic flows is presented in [10].

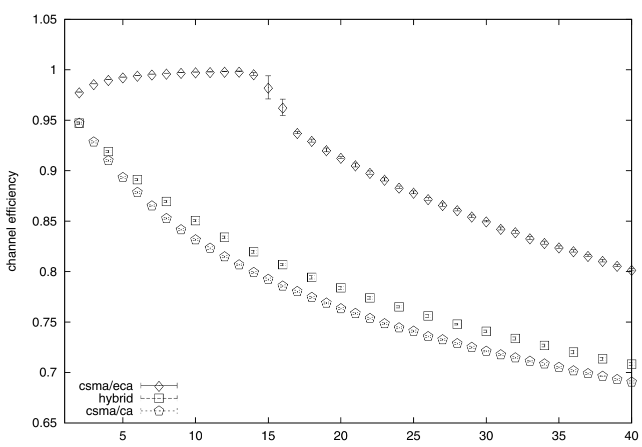

Fig. 7. The channel efficiency obtained for pure CSMA/ECA and pure CSMA/CA scenarios. The efficiency in a hybrid (mixed) scenario is also included.

VI. Conclusions

In this article we address the problem of collisions in CSMA networks. Our finding is that, instead of using a random backoff after all transmission attempts, it is better to use a random one after collisions and a deterministic one after successes. It reduces the chances of collisions as soon as two or more stations successfully transmit. As the system runs, it progressively converges to a collision-free operation that considerably improves the channel efficiency.

The proposed protocol outperforms CSMA/CA and, in the most typical scenarios, it even surpasses the theoretical upper bound associated with CSMA/CA networks that allow for dynamic parameter adjustment. Additionally, CSMA/ECA does not add any additional complexity to the implementation and it can fairly coexist with already deployed networks.

Acknowledgments

The authors benefited from the discussion with A. Banchs, B. Bellalta, A. Lozano and A. Vinel. Alberto Lopez Toledo is supported by the Institució Catalana de Recerca i Estudis Avançats (ICREA). This work was partially supported by the Spanish Government under grant TEC2008-06055/TEC; and the “COMONSENS” project – CONSOLIDER-INGENIO 2010 Program.

References

[1] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification*, IEEE Std. 802.11, 1999 Edition (Revised 2003).

[2] F. Borgonovo and L. Fratta, “A New Technique for Satellite Broadcast Channel Communication,” in *Symposium on Data Communications*, 1977, pp. 2.1–2.4.

[3] S. Tasaka, “Stability and Performance of the R-ALOHA Packet Broadcast System,” *IEEE Trans. Comput.*, vol. C-32, pp. 717–726, Aug. 1983.

[4] J. Barcelo, B. Bellalta, C. Cano, and M. Oliver, “Learning-BEB: Avoiding Collisions in WLAN,” in *Eunice*, 2008.

[5] G. Bianchi, “Performance Analysis of the IEEE 802.11 Distributed Coordination Function,” *IEEE J. Sel. Areas Commun.*, vol. 18, no. 3, pp. 535–547, 2000.

[6] T. Liu, J. Silvester, and A. Polydoros, “Performance Evaluation of R-ALOHA in Distributed Packet Radio Networks with Hard Real-Time Communications,” in *IEEE VTC*, vol. 2, 1995, pp. 554–558.

[7] R. Jain, *The Art of Computer Systems Performance Analysis*. John Wiley & Sons New York, 1991.

[8] J. Barcelo, B. Bellalta, A. Sfairopoulou, C. Cano, and M. Oliver, “CSMA with Enhanced Collision Avoidance: a Performance Assessment,” in *IEEE VTC Spring*, 2009.

[9] J. Barcelo, B. Bellalta, C. Cano, A. Sfairopoulou, M. Oliver, and J. Zuidweg, “Traffic Prioritization fo Carrience Sense Multiple Access with Enhanced Collision Avoidance,” in *MACOM (IEEE ICC)*, 2009.

[10] J. Barcelo, B. Bellalta, A. Sfairopoulou, C. Cano, and M. Oliver, “Carrier Sense Multiple Access with Enhanced Collision Avoidance: a Performance Analysis,” in *ACM IWCMC*, 2009.