Fairness in Collision-free WLANs

Luis Sanabria-Russo, Jaume Barcelo, Boris Bellalta

Universitat Pompeu Fabra, Barcelona, Spain
{luis.sanabria, jaume.barcelo, boris.bellalta}@upf.edu

***Abstract*—CSMA/ECA is a contention protocol that makes it possible to construct a collision-free schedule by using a deterministic backoff after successful transmissions. In this paper, we enhance the CSMA/ECA protocol with two properties that make it possible to fairly accommodate a large number of contenders in a collision-free schedule. The first property is called *resilience* and instructs the contenders not to reset their contention window after successful transmissions. Thanks to resilience, the protocol sustains a high throughput regardless of the number of contenders. The second property is called *fair-share*, and preserves fairness when different nodes use different contention windows. We present simulations results that evidence the performance advantages of using CSMA/ECA in combination with resilience and fair-share.**

***Index Terms*—Wireless, MAC, Collision-free, CSMA/ECA.**

I. Introduction

IEEE 802.11 networks use a shared and limited medium to establish communication among nodes. Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) is the protocol in charge of coordinating access to the wireless medium in order to avoid simultaneous transmissions by different nodes. If two or more nodes (or *contenders* for the medium N) attempt transmission at the same time, a *collision* accurs and the resulting transmission is disregarded by receivers.

Time in WLANs is slotted, that means that it is discrete and furthermore, it is divided in three slot types: *empty*, *successful* and *collision*; accounting for no transmission, successful transmission or collision, respectively.

Each contender attempting to transmit a packet chooses a uniformly random *backoff* counter $bo_r \in [0, \ldots, CW_{min} - 1]$, where CW_{min} is referred to as the minimum *contention window* with a typical value of 32. Each passing empty slot decrements bo_r by one; when the backoff counter reaches zero the contender will attempt transmission. The success of the transmission attempt is only confirmed by the reception of an *acknowledgement* (*ack*) from the receiver, otherwise a collision is assumed. If that were the case, each contender involved in the collision doubles its contention window $CW = 2^m CW_{min}, m \in [0, \ldots, 5]$ incrementing the *backoff stage* (m) by one and choosing another uniformly random backoff counter, bo_r. If the transmission is successful, the sender resets its contention window to the minimum value ($CW = CW_{min}$) and chooses another bo_r.

Carrier Sense Multiple Access with Enhanced Collision Avoidance (CSMA/ECA) achieves less collisions and outperforms CSMA/CA in most typical scenarios [1]. The only difference with CSMA/CA is that a deterministic backoff $bo_d = C$ is chosen after each successful transmission. C is defined in Eq. 1 as the *system capacity* and represents the maximum number of host (N) participating in the contend for transmission able to achieve a collision-free state. In Eq. 1, $\lceil \cdot \rceil$ is the ceiling operator, $E[\cdot]$ is the expectation operator, $\mathcal{U}$ is the uniform distribution and CW is the contention window.

$$C = \lceil E[\mathcal{U}[0, CW - 1]] \rceil \tag{1}$$

In a scenario where $N \leq C$, eventually all contenders will be able to pick different transmission slots, therefore achieving a collision-free state.

When the system is overcrowded, $N > C$, CSMA/ECA suffers a decrease in throughput as appreciated in Figure 1. This effect is caused by collisions originated by $N - C$ contenders forced to generate a random backoff counter and attempting transmission on slots previously picked by C nodes using a deterministic backoff.

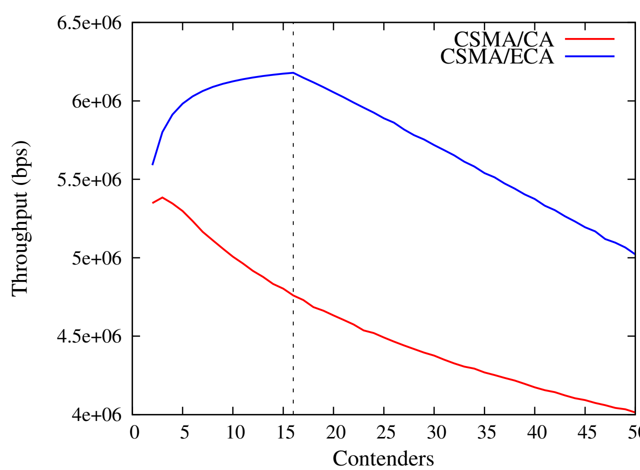

Fig. 1. Throughput and how it is affected when $N > C$

As $N - C$ nodes are unable to successfully transmit, collisions in turn force the C nodes that chose a deterministic backoff, to switch to a random one. The resulting effect is a system where all nodes choose a random backoff (CSMA/CA), which do not take advantage of the higher throughput CSMA/ECA offers.

In this work, a fully-distributed version of CSMA/ECA is presented and the throughput issue when $N > C$ is assessed.

II. A DESCENTRALIZED AND FAIR CSMA/ECA

In an overcrowded CSMA/ECA ($N > C$), colliding nodes will double CW each time and reset it ($CW = CW_{min}$) upon each transmission success, augmenting the collision probability. This behavior accounts for the throughput reduction in Figure 1 (see when $N = 16$). To avoid increasing the number of collisions and achieving a collision-free state for this greater number of contenders, nodes in CSMA/ECA do not reset the CW after successful transmissions. This is called *resiliency* from here forward.

Resiliency forces nodes to *stick* to its backoff stage, m; resulting in a greater system capacity because of the larger CW. This leads to a collision-free state while $N \leq C_m$, where C_m accounts for C in Eq. 1 computed with a CW in a backoff stage m.

Having a greater system capacity ($C_m > C$) means that more nodes are able to achieve a collision-free state. Nevertheless, in a $N \leq C_m$ scenario, contenders may have different deterministic backoff counters (bo_d) which provoke some nodes to access the channel more often than others. This fairness issue is averted with *fair-share*.

Fair-share consist on allowing each contender to send 2^m packets every time its backoff expires ($bo_d = 0$), making sure that contenders with longer backoff are compensated proportionally.

Figure 2, depicts how CSMA/ECA with resiliency and fair-share achieves greater throughput than CSMA/CA, maintaining a collision-free state and being fair (Jain's Fairness Index [2] (JFI) equals 1) for any number of contenders.

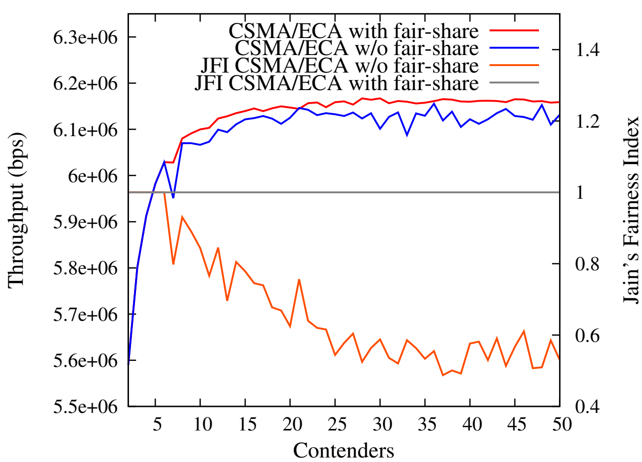

Fig. 2. Throughput and Jain's Fairness Index when implementing fair share in CSMA/ECA

The concept of resiliency and fair-share, was first introduced by Fang et al. in [3]. This work evaluates the performance of CSMA/ECA when implementing the concepts in a customized C++ simulator.

Evaluation

CSMA/ECA preserves backward compatibility with CSMA/CA (details in [1]), which is paramount for the coexistence and progressive adoption of the protocol. Many other performance evaluations, like a semi-analytical framework modelling the enhanced collision avoidance mechanism and comparing it with other access schemes (like Basic Access and RTS/CTS), are provided in [4].

Implementation is performed on a customized version of the COST [5] simulator. The system was set to be under saturation (nodes always have packets to transmit) during a period of a hundred thousand seconds at a maximum throughput of 11Mbps. The number of contenders ranges from 2 to 50. Further MAC-related parameters can be found under *stats/stats.h* in [6]; as well as the code for the whole CSMA/ECA implementation.

Figure 1 and Figure 2 are results derived from the evaluation platform.

III. FUTURE DIRECTIONS

Test

REFERENCES

[1] J. Barcelo, A. Toledo, C. Cano, and M. Oliver, "Fairness and Convergence of CSMA with Enhanced Collision Avoidance (ECA)," in *2010 IEEE International Conference on Communications (ICC)*, may 2010, pp. 1 –6.

[2] R. Jain, D. Chiu, and W. Hawe, *A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer System*. Eastern Research Laboratory, Digital Equipment Corporation, 1984.

[3] M. Fang, D. Malone, K. Duffy, and D. Leith, "Decentralised learning MACs for collision-free access in WLANs," *Wireless Networks*, pp. 1–16, 2010.

[4] G. Martorell, F. Riera-Palou, G. Femenias, J. Barcelo, and B. Bellalta, "On the performance evaluation of CSMA/E2CA protocol with open loop ARF-based adaptive modulation and coding," *European Wireless. 18th European Wireless Conference*, pp. 1 –8, april 2012.

[5] E. Yücesan, C. Chen, J. Snowdon, and J. Charnes, "COST: A component-oriented discrete event simulator," in *Winter Simulation Conference*, 2002.

[6] L. Sanabria-Russo, J. Barcelo, and B. Bellalta. (2012) Implementing CSMA/ECA in COST. [Online]. Available: https://github.com/SanabriaRusso/CSMA-E2CA