MAC design for WiFi infrastructure networks: a game-theoretic approach

Ilenia Tinnirello, Laura Giarré and Giovanni Neglia

***Abstract*—In WiFi networks, mobile nodes compete for accessing a shared channel by means of a random access protocol called Distributed Coordination Function (DCF). Although this protocol is in principle fair, since all the stations have the same probability to transmit on the channel, it has been shown that unfair behaviors may emerge in actual networking scenarios because of non-standard configurations of the nodes. Due to the proliferation of open source drivers and programmable cards, enabling an easy customization of the channel access policies, we propose a game-theoretic analysis of random access schemes. We show that even when stations are selfish, efficient equilibria conditions can be reached when they are interested in both uploading and downloading traffic. We explore the utilization of the Access Point as an arbitrator for improving the global network performance. Finally, we propose and evaluate some simple DCF extensions for practically implementing our theoretical findings.**

I. Introduction

The problem of resource sharing in WiFi networks [1], [2], is addressed by the Distributed Coordination Function (DCF), which is a Medium Access Control (MAC) protocol based on the paradigm of Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). The basic idea of the protocol is very simple: sensing the channel before transmitting, and waiting for a random backoff time when the channel is sensed busy. This random delay, introduced for preventing collisions among waiting stations, is slotted for efficiency reason and extracted in a range called contention window. Standard DCF assumes that the contention window is set to a minimum value (CW_{min}) at the first transmission attempt and is doubled up to a maximum value (CW_{max}) after each transmission failure.

The distributed DCF protocol is in principle fair, because the contention window settings CW_{min} and CW_{max} are homogeneous among the stations, thus ensuring that each node receives in the long term the same number of access opportunities. Nevertheless, some unexpected behaviors have been recognized as a consequence of non-standard settings of the contention windows. The stations employing lower contention windows gain probabilistically a higher number of transmission opportunities, at the expense of compliant stations. These settings can be changed by the card manufacturers, as recognized in [3], or by the end users thanks to the availability of open-source drivers.

I. Tinnirello is with DIEET, Università di Palermo, 90128 Palermo, Italy ilenia.tinnirello@tti.unipa.it

L. Giarré is with DIETT, Università di Palermo, 90128 Palermo, Italy giarre@unipa.it

G. Neglia is with Équipe Projet INRIA Maestro, 2004 route des Lucioles, 06902 Sophia Antipolis, France, giovanni.neglia@inria.fr

This work has been partially supported by the European Union under project FP7-257263 (FLAVIA).

Another problem specific to infrastructure networks is given by the repartition between uplink and downlink resources. Infrastructure networks are characterized by a star topology, which connects multiple mobile nodes to a common station called Access Point (AP). On one side, mobile stations can upload traffic to the AP, which is connected to external networks (e.g. to the Internet); on the other side, they can download traffic from the external networks through the AP. Since the AP contends as a normal station to the channel, its channel access probability is the same of the other mobile stations. This implies that the AP aggregated throughput, i.e. the downlink bandwidth, is equal to the throughput perceived by any other stations, thus resulting in a per-station downlink bandwidth much lower than the uplink one [4]. Indeed, recent extensions of DCF [5] (namely, the EDCA protocol) allow the AP to set heterogeneous contention windows among the stations to give priority to downlink throughput or to delay-sensitive traffic. Thus, nowadays nodes can adapt their contention windows according to the values signaled by the AP for each traffic class. However, there is the risk to exploit this adaptation in a selfish manner, for example by using a contention window value of a higher priority class [6].

These considerations motivate a game theoretical analysis of DCF, in order to propose some protocol extensions able to cope with the resource sharing problems. The problem can be formulated as a non cooperative game, whose players are n contending stations. When stations work in saturation conditions, i.e. they always have a packet available in the transmission buffer, DCF can be modeled as a slotted access protocol, while station behavior can be summarized in terms of per-slot access probability [7]. Therefore, we consider that the strategy of a generic station i at each time slot is its access probability, say it τ_i. A vector of station payoffs $(J_1, J_2, \ldots, J_n)$ can be defined according to the network and application scenario [8]. Previous studies have mainly considered that each node utility is given by the node saturation throughput performance [9]. In [10], [11], it has been shown that a utility function equal to the node upload throughput may lead to an inefficient Nash equilibrium in which stations transmit in every channel slot (i.e. play $\tau = 1$). This situation creates a resource collapse, because all stations transmit simultaneously thus destroying all packet transmissions. More complex utility functions combining upload throughput and costs related to collision rates [10], [12], [13] or to energy consumptions [14], [15] lead to different equilibria, but they appear less natural and implicitly assume that all the nodes have the same energy constraints or collision costs. In some cases [16], the utility function does not correspond to any performance metric and so appears completely arbitrary.

In this paper, we show that efficient equilibria can be naturally reached in infrastructure networks *when stations are interested in both uploading and downloading traffic*. Since the utility of each station depends not only on its throughput but also on the AP throughput, no station is motivated to transmit continuously. Extending our preliminary results in [17] and in [18], we derive Nash equilibria and Pareto optimal conditions as a function of the network scenario. We also define a mechanism design scheme, in which the AP plays the role of arbitrator to improve the global performance of the network, by forcing desired equilibria conditions.

The rest of the paper is organized as follows. In Sec. I-A we briefly review some research papers related or complementary to ours; in Sec. II we carry the game theoretic analysis and we find the Nash equilibria and the Pareto Optimal solutions; in Sec. III we analyze the use of the AP for performing some mechanism design schemes; in Sec. IV we show the MAC scheme implementation and the performance evaluation through simulations; finally we drew some conclusive remarks in Sec. V.

A. Our Scenario and Related Work

In recent years, the proliferation of open-source drivers for WiFi cards has motivated several game-theoretical analysis of different selfish behaviors of 802.11 nodes. In particular, great attention has been dedicated to the backoff attacks [9]-[18] (i.e. to the presence of selfish nodes changing the contention window value in order to increase their throughput), which are also the focus of our work. However, our scenario differs from previous ones because we consider an infrastructure network, where each station is involved in bidirectional traffic flows. This assumption introduces an intrinsic self-regulatory mechanism in the contention process, since each station needs to constrain its transmission rate at least to leave space to its own downloading traffic. Indeed, we believe that this assumption is pretty realistic. In most cases, nodes are interested in a bilateral information exchange. Even applications like file uploading or downloading that look unidirectional, in reality require some signaling traffic in the opposite direction. Moreover, in most recent P2P systems, peers interested in the same file are incentivized to barter their chunks, so that each peer downloads and uploads the file at the same time.

We also assume that the stations work with saturated transmission buffers. This is quite natural if we think about large data transfers and it does not necessarily imply that a station should try to transmit at the highest possible rate. In fact in our scenario the station is also interested to its download rate.

Therefore, in this paper we define a new utility function able to simultaneously account for a generic bidirectional traffic scenario and a desired uplink/downlink bandwidth repartition. When such a ratio goes to infinity, we asymptotically recover the case when a node is only interested in its upload rate [10].

Moreover, in an infrastructure network, the Access Point (AP) is a central element that can try to implement fair resources repartition and punish misbehaving nodes when needed. In particular, although we prove that stations' interest in bidirectional traffic is sufficient to lead to efficient equilibria, we also suggest how to use the AP for dealing with the presence of stations interested in upload traffic only. The upload traffic scenario has been largely analyzed in literature and our solution is derived by the approach introduced in [10] (based on a MAC-layer artificial throughput control) that has been adapted to an infrastructure network. Specifically, while [10] proposes a distributed jamming mechanism for destroying transmissions of too greedy stations, here we centrally control the unidirectional upload flows, by selectively dropping the ACK frames of greedy stations at the AP. Obviously, different punishment schemes can be implemented at different layers. In [9], [19] the authors describe two driver-level approaches, called CRISP and SPELL, based on software modules to be installed at each contending node, for piloting the MAC settings of the cards according to some monitored parameters. Stations are discouraged from installing different driver modules because their bandwidth share is asymptotically inferior to what they would receive playing CRISP or SPELL. In [20] both routing-layer and application-layer punishments are considered. At the routing layer, each node can stop the forwarding of traffic packets sent by stations recognized as selfish, while at the application layer it is possible to shape or deny the traffic incoming from selfish nodes. Similarly, rather than dropping the ACK frames, the AP can stop forwarding the traffic sent by selfish nodes to the application layer or to the wired network. However, since we are assuming that stations work in saturation conditions, we do not model the interaction between the packet generation process at higher layers and the MAC-layer queue. Therefore, we limit our analysis on MAC-layer traffic control, because in the current framework the higher layer discard process cannot have effects on the MAC-layer queues and contention process.

Finally, while our work is focused on analyzing and/or correcting selfish node behaviors, some other related work [21], [22] has addressed the issue of how to identify a misbehaving wireless node. These papers consider different misbehaviors, including those aiming to increase the upload rate and those aiming to increase the download rate. They present DOMINO, a software to be installed in or near the Access Point, to detect and identify greedy stations. It is important to note that DOMINO does not address the problem of how to control or punish the identified greedy stations. Similarly to DOMINO, DREAM [23] is a solution to detect and contrast a specific attack (in this sense it is a malicious behavior rather than a selfish one): a host could maliciously modify the protocol timeout mechanism (e.g., by changing SIFS parameter in 802.11) and cause MAC frames to be dropped at well-behaved nodes. Both these works are orthogonal to our purposes and could be integrated in our framework.

II. Contention-based channel access: a game theoretic analysis

We assume that all the stations try permanently to transmit on the channel because their transmission queues are never empty, i.e. they work in saturation conditions. We have verified that non-saturated stations affect the performance of saturated stations only marginally and regardless of their contention windows. When all stations are saturated, it has been shown [24] that DCF can be accurately approximated as a persistent slotted access protocol, because packet transmissions can be originated only at given time instants. Figure 1 shows an example of DCF as a slotted protocol. After a busy time, stations A and B defer their transmissions by extracting a random slotted delay (respectively, 4 and 8 slots). Since the timer of station A expires first, station A acquires the right to transmit on the channel. The next transmission, which results in a collision, is performed again after an integer number of backoff slots from the end of the previous channel activity. Therefore, the channel time can be divided into slots of uneven duration delimited by potential transmission instants. In a generic channel slot, each station i has approximately a fixed probability τ_i to transmit, which depends on the average backoff values.

A. Station strategies

Let n be the number of saturated contending stations. We assume that each station i is rational, and can arbitrarily choose its channel access probability τ_i in $[0,1]$. This choice can be readily implemented by tuning opportunistically the minimum and the maximum values of the contention windows. By observing that $\tau_i = 1/(1+E[W]/2)$, where $E[W]$ is the average contention window used by station, a solution is to set $CW^i_{min} = CW^i_{max} = 2/\tau_i - 2$[1]. The set of all the strategies in the network is then $[0,1]^n$. We define as *outcome* of the game a specific set of strategies taken by the players, then a vector $\boldsymbol{\tau} = (\tau_1, \tau_2, \cdots, \tau_n) \in [0,1]^n$. We call it *homogeneous* outcome whenever all the stations play the same strategy, i.e. $\boldsymbol{\tau} = (\tau, \tau, ...\tau)$.

Performance perceived by a given station i not only depends on the probability τ_i to access the channel, but also on the probability that no other station transmits in the same slot. Therefore, from the point of view of station i, the vector strategy $\boldsymbol{\tau}$ can be represented by the couple of values (τ_i, p_i), where $p_i = 1 - \prod_{j \neq i}(1-\tau_j)$, the probability that at least another station transmits, summarizes the interactions with all the other mobile stations. In the presence of downlink traffic, we also assume, unless otherwise specified, that the AP contends for the channel as a legacy DCF station with saturated downlink traffic. Thus, the overall collision probability suffered by station i results to be $1-(1-p_i)(1-\tau_{AP})$, where τ_{AP} is the channel access probability employed by the AP. Since the AP is a legacy station, its transmission probability is not chosen by the AP, but is function of the perceived collision probability p_{AP}, $\tau_{AP} = f(p_{AP})$. The function $f()$ has been derived in [7]:

$$\tau = f(p) = \frac{1}{1+E[W]/2}$$
$$\begin{cases} \frac{2(1-p^{R+1})}{1-p^{R+1}+(1-p)\sum_{i=0}^{R} p^i W(i)} & 0 \leq p < 1 \\ \frac{2(R+1)}{R+1+\sum_{i=0}^{R} W(i)} & p = 1 \end{cases} \quad (1)$$

[1]Obviously, this solution introduces some quantization effects on the actual τ_i values, since the contention window can assume only integer values. A discussion about these effects is provided in [26].

where R is the retry limit employed in the network (i.e. the maximum number of times the station tries to retransmit a packet as consequence of collisions) and $W(i)$ is the contention window at the i-th retry stage (i.e. $W(i) = \min\{2^i CW_{min}, CW_{max}\}$). We can evaluate the AP collision probability as a function of the vector strategy $\boldsymbol{\tau}$ or as a function of a generic couple (τ_i, p_i):

$$p_{AP}(\boldsymbol{\tau}) = 1 - \prod_{i=1}^{n}(1-\tau_i) = 1-(1-p_i)(1-\tau_i)$$

B. Station Utility

According to the slotted channel model, the random access process can be described as a sequence of slots resulting in a successful transmission (when only one station accesses the channel), in a collision (when two or more stations access the channel), or in an idle slot (when no station accesses the channel). By observing that each slot boundary represents a regeneration instant [25] for the access process, the throughput of each station can be readily evaluated as the ratio between the average number of bits transmitted in each slot and the average duration of each slot [24].

In our study we consider that the AP could allocate a different downlink throughput to each station by implementing a specific scheduling mechanism, as described in Sec. II-D. For now on we consider that the scheduling rule is given and we denote x_i the fraction of the AP's throughput (S_{AP}) given to station i (clearly $\sum_i x_i = 1$). We can express the uplink throughput S^i_u and the downlink throughput S^i_d for the i-th station as [24]:

$$S^i_u(\tau_i, p_i) = \frac{\tau_i(1-p_i)(1-\tau_{AP})P}{P_{idle}\sigma + [1-P_{idle}]T} \quad (2)$$

$$S^i_d(\tau_i, p_i) = x_i S_{AP}(p_{AP}) = x_i \frac{f(p_{AP})(1-p_{AP})P}{P_{idle}\sigma + [1-P_{idle}]T} \quad (3)$$

where P is the frame payload which is assumed to be fixed, σ and T are, respectively, the empty and the busy slot duration[2], and P_{idle} is the probability that neither the stations, nor the AP transmit on the channel, i.e. $P_{idle} = (1-p_{AP})(1-\tau_{AP})$.

We define the utility function J_i for the mobile station i as:

$$J_i = \min\{S^i_u, k_i S^i_d\} \quad (4)$$

The rationale of this definition is the assumption that the station applications require bandwidth on both directions. The coefficient $k_i \in (0,\infty)$ takes into account the desired ratio between the uplink and the downlink throughput required by station i and we call it the *application requirement* at station i. If $k_i = 1$, station i requires the same throughput in both directions. The limit case $k_i = 0$ corresponds to a user i only interested in the downloading rate S^i_d. In this case it is trivial to determine the user's dominant strategy, that is to not transmit at all in order to avoid any collision with the AP. For this reason, in this paper we exclude the case $k_i = 0$. Conversely, the limit case $k_i = \infty$ corresponds to a user i only interested in

[2]We are implicitly considering a basic access scheme, with EIFS=ACK_Timeout+DIFS, which corresponds to have a fixed busy slot duration in both the cases of successful transmission and collision.

the uploading rate S_u^i (as assumed in most previous literature). Apart from the mechanism design analysis, we briefly treat this case, since most limit results have been discussed in [10], [11]. When all the coefficients k_i are equal to a fixed value k, we talk about uniform application requirements.

Figure 2 plots the utility of a given station i, in the case of uniform application requirements with $k = 1$, 802.11b physical (PHY) layer, $P = 1500$ bytes, a data rate equal to 11 Mbps, and an acknowledgment rate of 1 Mbps. In such a scenario, by including physical preambles, acknowledgment transmissions, MAC headers and interframe times, the T duration is equal to 1667 μs. Different network conditions, summarized by different values of the p_i probability, have been considered. The collision probability p_i takes into account only the competing mobile stations, so that the actual collision probability is given by $1-(1-p_i)(1-\tau_{AP})$. From the figure, it is evident that, for each p_i, the utility is maximized for a given best response value (e.g. about 0.01 for $p = 0.15$), which slightly increases as p_i grows. For uniform application requirements, we consider that the AP equally shares its throughput among the contending stations (i.e. $x_i = 1/n \ \forall i$). In this case, it is useful to define the single variable functions

$$\begin{aligned} S_u^{hom}(\tau) &= S_u(\tau, 1-(1-\tau)^{n-1}) \\ S_d^{hom}(\tau) &= \tfrac{1}{n} S_{AP}(1-(1-\tau)^n) \end{aligned} \tag{5}$$

representing, respectively, the uplink and downlink throughput perceived by each station in the case of homogeneous outcomes $(\boldsymbol{\tau}|\tau_i = \tau, \forall i)$.

Figure 3 plots the utility of a given station in the case of homogeneous outcomes for $n = 2$ and $n = 10$, and for different uniform k values. In these curves $p_i = 1-(1-\tau)^{n-1}$ is not fixed, because the strategy changes are not unilateral. The optimal strategy, which maximizes the station utility, is a function of both n and k.

C. Nash Equilibria

We are interested in characterizing Nash Equilibria (NE) of our game model where stations achieve a non-null utility. The inefficient equilibria in which all stations achieve an utility value equal to 0 can be easily found by observing that:

Remark 2.1: In general, station i utility is a function of the whole set of strategies $(\boldsymbol{\tau})$, but it is constant and equal to 0 if a) $p_i = 1$, i.e. if at least one of the other players is transmitting with probability 1 $(\exists j \neq i \mid \tau_j = 1)$, or if b) $\tau_i = 0$. We also observe that the AP access probability τ_{AP} depends on τ_i and p_i according to (1) and cannot be equal to 1 for standard contention window values.

Proposition 2.1: The vectors of strategies $\boldsymbol{\tau}$, such that $\exists\ j, l \in 1, 2, \cdots n \mid \tau_j = 1,\ \tau_l = 1$ are NE of the distributed access game in which all stations achieve an utility value that is constant and equals 0.

Proof: The result is an immediate consequence of Remark 2.1. If there are at least two stations transmitting with probability 1, then the channel is entirely wasted because of collisions and $S_u^i = S_d^i = 0, \forall i$. In these conditions, $J_i = 0 \ \forall i$ and stations are not motivated in changing their strategies. ■

The following remark will be useful for characterizing more efficient NE.

Remark 2.2: Consider a generic station i and the collision probability $p_i \in (0, 1)$ suffered because of the other station strategies. By derivation, it can be easily proved that $S_d^i(\tau_i, p_i)$ is a monotonic decreasing function of τ_i, starting from $S_d^i(0, p_i) > 0$, and that $S_u^i(\tau_i, p_i)$ is a monotonic increasing function of τ_i, starting from $S_u^i(0, p_i) = 0$.

Let us denote a *best response* strategy of a station i as $\tau_i^{(br)}$. For $k_i = \infty$, the station utility function is equal only to $S_u^i(\tau_i, p_i)$. From Remarks 2.1 and 2.2, it results that the utility is maximized for $\tau_i^{(br)} = 1$ when $p_i < 1$ (then there is a unique best response), and it is constant to 0 when $p_i = 1$ (then any strategy is the best response). For $k_i \neq \infty$ and $p_i < 1$, from Remark 2.2 we can state that the utility J_i is maximized for $\tau_i^{(br)} \in (0, 1)$ such that $S_u^i(\tau_i^{(br)}, p_i) = k_i S_d^i(\tau_i^{(br)}, p_i)$. It follows that, for $p_i < 1$, $\tau_i^{(br)}$ is the solution of the following implicit equation:

$$\begin{aligned} \tau_i^{(br)} &= \tfrac{k_i x_i \tau_{AP}}{1-(1-k_i x_i)\tau_{AP}} = \\ &\tfrac{k_i x_i f\left(1-(1-p_i)\left(1-\tau_i^{(br)}\right)\right)}{1-(1-k_i x_i) f\left(1-(1-p_i)\left(1-\tau_i^{(br)}\right)\right)} \end{aligned} \tag{6}$$

The previous equation has a single solution τ_i^* in the range $(0, 1)$. In fact, the left side $l(\tau_i^{(br)})$ of (6) is a continuous strictly increasing function of $\tau_i^{(br)}$ with values in $[0, 1]$. For $p_i \neq 1$, the right side $r(\tau_i^{(br)})$ is a continuous strictly decreasing function with values in the same interval (we are going to show it below), and with $r(0) > l(0) = 0$ and $r(1) < l(1) = 1$. Then, there is necessarily a unique solution for $p_i \neq 1$. In order to check our statement about the function on the right side of (6), we can express it as the composition of three functions $h(y) = k_i x_i y/(1-(1-k_i x_i)y)$, $f(x)$, $g(\tau_i^{(br)}) = 1-(1-p_i)(1-\tau_i^{(br)})$. Now $g()$ is strictly increasing for $p_i \neq 1$ and has value in $[0, 1]$. $f()$ is strictly decreasing and has value in $[0, 1]$ (this is evident if we remind that $f(x)$ is the probability to access the channel for a legacy station that experience a collision probability x). $h()$ is strictly increasing in the interval $[0, 1]$ (for all the possible values of $k_i x_i$). Then, the composition $h \circ f \circ g$ is strictly decreasing for $p_i \neq 1$. The solution τ_i^* of (6) can be found numerically in a few fixed point iterations.

Note that, as originally proved in literature and revisited in [8], if there are stations with only uplink traffic flows, the NE of the distributed access game with non-null utility values are all and only the vector of strategies $\boldsymbol{\tau}$, such that $\exists!\ i \in \{1, 2, \cdots n\} \mid \tau_i = 1$ and $k_i = \infty$. In this particular case our general utility function leads to the same results of [10], [11]. Conversely, when $k_i \neq \infty \ \forall i$, the next proposition shows that there is a non trivial NE where all players obtain non null utility.

Proposition 2.2: For a given vector $\boldsymbol{k}$ of application requirements $(k_1, k_2, \cdots k_n)$ in $(0, \infty)^n$, and a given vector of downlink throughput coefficients $(x_1, x_2, \cdots x_n)$, it exists a unique NE $\boldsymbol{\tau}$ with non-null utility values.

Proof: We already know that all the vectors of strategies such that at least two stations transmit with probability 1 are NE with zero utility. Moreover, an outcome with only one station, say it i, transmitting with $\tau_i = 1$ cannot be a NE because the station would find convenient to unilaterally reduce τ_i to increase its downloading rate. Then we can conclude that a NE with non-null utility values can only exist for $\tau \in [0,1)^n$, or equivalently $p_j < 1$ for all j, so in what follows we consider this case. A NE is an outcome $\boldsymbol{\tau}^*$ of mutual best responses, that can be expressed by (6), being that $p_i < 1$ for all i, i.e. an outcome such that for each i, $\tau_i^* = \frac{k_i x_i \tau_{AP}}{1-(1-k_i x_i)\tau_{AP}}$, with $\tau_{AP} = f(1 - \prod_{i=1}^n (1-\tau_i^*))$. Although the above equations characterize the best responses only for $\boldsymbol{\tau} \in [0,1)^n$, we will first look for solutions with $\boldsymbol{\tau} \in [0,1]^n$, knowing that solutions with one transmission probability equal to 1 are not NE. The conditions can be geometrically represented in the $n+1$ dimensional hypercube $[0,1]^{n+1}$, where the first n dimensions are the strategies $\tau_1, \tau_2, \cdots, \tau_n$ and the last dimension is the AP access probability τ_{AP}. We denote with $\boldsymbol{\theta} = (\tau_1, \tau_2, \cdots \tau_n, \tau_{AP})$ a generic vector in this hypercube. Moreover, $\mathbf{0}^m$ and $\mathbf{1}^m$ is the m-dimensional vectors whose elements are respectively all equal to 0 and to 1.

A solution of the set of equations, if any, corresponds to the intersection of the n-dimensional hypersurface S identified by the equation $\tau_{AP} = f(1 - \prod_{i=1}^n (1-\tau_i))$ with $(\tau_1, \tau_2, \cdots, \tau_n) \in [0,1]^n$, and the one-dimensional curve C, identified by the set of n equations $\tau_i = \frac{k_i x_i \tau_{AP}}{1-(1-k_i x_i)\tau_{AP}}$ with $\tau_{AP} \in [0,1]$.

We observe that S is continuous, and it divides the hypercube in three regions: the surface S itself, the region R_b of the points "below the surface", i.e. $R_b = \{\boldsymbol{\theta} | \tau_{AP} < f(1-\prod_{i=1}^n (1-\tau_i))\}$, and the region R_a of the points "above" it, i.e. $R_a = \{\boldsymbol{\theta} | \tau_{AP} > f(1 - \prod_{i=1}^n (1-\tau_i))\}$. Note that the point $\mathbf{0}^{n+1}$ belongs to R_b, because $f(p_{AP}(\mathbf{0}^n)) > 0$, and the point $\mathbf{1}^{n+1}$ belongs to R_a because $f(p_{AP}(\mathbf{1}^n)) < 1$. The one-dimensional curve is also continuous and it connects $\mathbf{0}^{n+1}$ (for $\tau_{AP} = 0$) and $\mathbf{1}^{n+1}$ (for $\tau_{AP} = 1$), then it necessarily intersects the surface. This proves that it exists an intersection point.

Moreover, it is easy to check that, for each i, $\frac{\partial \tau_{AP}}{\partial \tau_i}|_{\boldsymbol{\tau} \in S} < 0$ and $\frac{\partial \tau_i}{\partial \tau_{AP}}|_{\boldsymbol{\tau} \in C} > 0$. Then, there must be a unique intersection point.

Finally, we observe that this intersection point need to belong to $(0,1)^{n+1}$, because the sign of the derivatives for the point in C imply that all the points of C lie in $(0,1)^{n+1}$ but $\mathbf{0}^{n+1}$ and $\mathbf{1}^{n+1}$, neither of which could be the intersection point because we have shown that they do not belong to S. Then, the intersection point is indeed a NE and moreover the corresponding nodes' utilities are all non-null. ∎

Figure 4 shows some examples of equilibrium conditions in terms of surface and parametric curve intersections for two stations (hence in a 3-dimensional space) and for different k_1 and k_2 values.

D. Downlink Scheduling Scheme

For evaluating the ratio x_i of the downlink throughput to be assigned to each station, the AP can employ different policies. If the AP is not aware of the application requirements of each station, a possible solution is to equally share the downlink throughput among the stations (i.e. $x_i = 1/n \ \forall i$). Under this policy, since each station i tries to get an uplink throughput equal to $k_i S_d^i = k_i/nS_{AP}$, the total uplink and downlink throughput perceived by each station at the NE is $(1+k_i)/nS_{AP}$. This implies that stations requiring large k_i values will consume a large fraction of the network resources.

Whenever the AP is able to estimate the application requirement of each station (by monitoring the ratio between the uplink and downlink throughput perceived by each station), it can implement a different downlink scheduling policy devised to improve the network fairness. For example, by imposing that the total per-node bandwidth $S_u^i + S_d^i = (1+k_i)x_i S_{AP}$ is equal for each station, with the constraint $\sum_i x_i = 1$, it results:

$$x_i = \frac{\frac{1}{k_i+1}}{\sum_{j=1}^n \frac{1}{k_j+1}} \tag{7}$$

When multiple stations have the same application requirements, we can group these stations into applications classes and represent each class i with a single k_i value. Stations belonging to the same classes will also receive the same downlink ration x_i.

Note that the first scheduling policy guarantees a uniform utility for all the stations, while the second *application-aware* scheduling policy equalizes the total per-station bandwidth, thus resulting in heterogeneous utilities. Therefore, we could argue that a different utility definition, based on the total per-station bandwidth, could be considered. However, such a definition does not capture the bidirectional nature of the considered applications and could lead to situations in which the uplink or downlink bandwidth is null.

E. Social utility

In this section, we try to identify desirable outcomes from a global point of view. A natural choice is to look at outcomes that maximize a social utility function, such as the minimum utility $J_S(\boldsymbol{\tau})$ perceived in the network: $J_S(\boldsymbol{\tau}) = \min_{i=1\cdots n} J_i$. This global utility is often referred to as social utility[3]. The following remark will be useful for such a characterization.

Remark 2.3: The uplink throughput $S_u^{hom}(\tau)$ given in (5) and perceived in the case of homogeneous outcomes is a non-monotonic function in τ, with a single maximum value $S_u^{hom}(\tau_x)$, for $\tau_x \in (0,1)$.

Proposition 2.3: The social utility is maximized for a unique homogeneous outcome $(\tau', \tau', \cdots, \tau')$ and such outcome is Pareto Optimal.

Proof: From the utility definition, we have that the minimum utility perceived in the network is given by $J_S(\boldsymbol{\tau}) = \min_{i=1,\cdots n} \{\min\{S_u^i, x_i k_i S_{AP}\}\}$. Let us consider m such that $x_m k_m \leq x_i k_i$ for all i. It is evident that the social utility can be expressed in a simpler way as $J_S(\boldsymbol{\tau}) = \min\{\min_{i=1,\cdots n}\{S_u^i\}, x_m k_m S_{AP}\}$. Therefore, the minimum utility is due to the minimum uplink throughput among all the stations or to the downlink throughput of station m.

[3]In this application scenario, it does not seem that is meaningful to consider as global utility the sum of all the utilities. Consider for example that for $k_i = \infty$ for all i, according to this definition, the optimal social outcome would be the extremely unfair one where a single node accesses the channel with probability 1 and all the others do not transmit.

Let us consider an outcome maximizing the social utility such that $\min_{i=1,\cdots n}\{S_u^i\} < x_m k_m S_{AP}$. We prove that this outcome has to be homogeneous. In fact let us consider, without loss of generality, a non-homogeneous ordered vector $\boldsymbol{\tau}$ with $0 \le \tau_1 = \cdots \tau_j < \tau_{j+1} \le \cdots \tau_n \le 1$, then $S_u^h(\tau_h, p_h) = \min_i S_u^i(\tau_i, p_i)$ for $h = 1, 2, \cdots, j$. Let τ'_{j+1} be a new strategy for the $(j+1)$-th station, such that $\tau'_{j+1} = \tau_j$. For the new outcome $(\tau_1, \cdots, \tau_j, \tau'_{j+1} \cdots \tau_n)$, the social utility is still determined by the minimum uplink throughput that is now the throughput perceived by stations from 1 to $j+1$. This throughput is higher than the previous one, since $S_u^i(\tau_i, p_i)$ is monotonic decreasing in τ_{j+1} for $i \ne j+1$. Then the new outcome has strictly higher social utility, this proves that an outcome has to be homogeneous in order to maximize the minimum uplink throughput.

Let us then consider the other case, i.e. when an outcome $\boldsymbol{\tau}$ maximizing the social utility is such that $\min_{i=1,\cdots n}\{S_u^i\} \ge x_m k_m S_{AP}$. It has to be $S_u^m = x_m k_m S_{AP}$, because if it were $S_u^m > x_m k_m S_{AP}$, then station m could increase its downlink throughput and (then increase the social utility) by reducing τ_m. Then it has to be $S_u^i \ge S_u^m$ for all i, i.e. $\tau_i \ge \tau_m$ for all i. In particular all the τ_i have to be equal, because otherwise we could reduce the largest access probability to τ_m and improve the social utility. The conclusion is that also in this case an outcome maximizing the social utility has to be homogeneous.

Now we prove that there is a unique outcome maximizing the social utility. In fact $J_S((\tau, \tau, \cdots, \tau))$ has a unique maximum because S_u^{hom} has a unique maximum (see Remark 2.3) and S_{AP}^{hom} is non-increasing. We denote the outcome maximizing the social utility $\boldsymbol{\tau'} = (\tau', \tau' \cdots, \tau')$.

Finally, we prove the Pareto optimality. We recall that a Pareto optimal outcome is one such that no one could be made better off by changing the vector of strategies without making someone else worse off. Now, if we take any outcome different from $\boldsymbol{\tau'}$ the corresponding social utility is strictly smaller, this means that there is at least one station whose utility has decreased. ■

It is easy to check when the social utility is limited by the uplink throughput or by the downlink throughput:

Remark 2.4: Being τ^* the value for which $S_u^{hom}(\tau) = x_m k_m S_{AP}^{hom}(\tau)$, and τ_x the homogeneous strategy defined in Remark 2.3, the optimal social outcome $\boldsymbol{\tau'}$ is such that $\tau' = \tau^*$ if $\tau^* \le \tau_x$, or $\tau' = \tau_x$ when $\tau_x < \tau^*$.

As an example, Figure 3 plots $J_S(\boldsymbol{\tau}) = \min\{S_u^{hom}(\tau), x_m k_m S_{AP}^{hom}(\tau)\}$ for $x_m = 1/n$ and different $k_m = k$ values, showing cases where the maximum utility value J_S is limited by the uplink throughput (i.e. $\tau^* > \tau_x$) or by the downlink one (i.e. $\tau^* \le \tau_x$). Note that the intersection between the curves corresponding to $S_u^{hom}(\tau)$ and $x_m k_m S_{AP}^{hom}(\tau)$ depends on the scheduling policy and on the application requirements. When $x_i = 1/n\ \forall i$, or when x_i is given by (7), the index of the station perceiving the lower utility at the NE is that of the station with the smallest k_i value, i.e. $m = \arg\min_i k_i$.

It is interesting to note that for $k = \infty$ the optimal social outcome $\boldsymbol{\tau'}$ coincides with the Nash bargaining solution of the game that is studied in [10], [11]. Such solution corresponds to maximize $\prod_{i=1,\cdots,n} S_u^i$, and the homogeneous outcome is the one maximizing each S_u^i. For $k = \infty$, this is equivalent to maximize the defined social utility. This is an unexpected link between two different mathematical formulations.

An immediate consequence of Remark 2.4 when $x_i k_i$ is constant is the following result:

Corollary 2.4: If $x_i k_i$ is constant for all i and the solution τ^* of (6) for $p_i = 1 - (1-\tau^*)^{n-1}$ is lower or equal to τ_x, then the NE $(\tau^*, \tau^*, \cdots \tau^*)$ is Pareto optimal.

Proof: When $x_i k_i$ is constant for all i, τ^* is also the strategy at the (homogeneous) NE identified in Prop. 2.2, and if $\tau^* < \tau_x$ then $\tau' = \tau^*$. ■

Figure 3 shows that the limit condition $\tau^* = \tau_x$ is approximately reached for $k_m = 20$ in the case of $n = 2$, and for $k_m = 11$ in the case of $n = 10$. For smaller k_m values, the homogeneous NE $\boldsymbol{\tau^*}$ is Pareto optimal. For larger k_m values, including the unidirectional traffic case $k_m = \infty$, the Pareto optimal outcome $\boldsymbol{\tau'}$ is not an equilibrium point and the NE $\boldsymbol{\tau^*}$ gives poor performance (i.e. performance much worse than $J_S^{hom}(\tau')$).

Note that Prop. 2.1 implies that the *price of anarchy*[4] is infinite. In fact the global utility at the NE described by Prop. 2.1 is 0, because no user can transmit.

III. Channel access mechanism design

In this section, we explore the possibility of using the Access Point to change the S_d^i or S_u^i functions, in order to force desired equilibrium outcomes. Indeed, since the AP plays the role of gateway to external networks, it can also play the role of arbitrator for optimizing the global performance of its access network.

A. Tuning of the AP channel access probability

In order to improve the downlink short-term fairness and the overall network performance, we can use the AP channel access probability τ_{AP} as a tuning parameter. In this case, τ_{AP} does not depend on $\boldsymbol{\tau}$ according to (1), but it is equal to a fixed value c, which can be tuned by the AP. The best response (6) for each station i is equal to

$$\tau_i^+ = \frac{k_i x_i \cdot c}{1 - (1 - k_i x_i)c} \tag{8}$$

and the NE in $(0,1)^n$ becomes the intersection between an hyperplane $\tau_{AP} = c$ and the parametric curve C identified by the best response equations. Let $J_i^{NE}(c)$ and $S_{AP}^{NE}(c)$, respectively, the station i utility and the AP throughput perceived at the NE for each different c value selected by the AP. When $x_i \ne 0\ \forall i$, the utility value J_i^{NE} of each station is proportional to the AP throughput. Therefore, all the utilities can be maximized by maximizing the same function S_{AP}^{NE}:

$$\max_c J_i^{NE}(c) = k_i x_i \cdot \max_c S_{AP}^{NE}(c).$$

Figure 5 shows the effects of the c tuning on the total bandwidth perceived by n_1 and n_2 contending stations belonging

[4]Remind that the price of anarchy is defined as the ratio between the optimal global utility and the global utility at the worst Nash Equilibrium.

to two different application classes, under the application-aware scheduling policy (7). The figure refers to a scenario in which one station requires an uplink/downlink throughput ratio $k_1 = 1$, and all n_2 stations require a k_2 ratio equal to 1, 2 or 10 (as indicated in the figure legend). The packet size has been set to 1500 bytes, with an 802.11b PHY and a data rate equal to 11 Mbps. The figure enlightens that the per-station bandwidth is maximized for a given $c = \tau_{AP_o}$ value. For example, for $n_1 = 1$, $n_2 = 10$ and $k_2 = 10$, a maximum bandwidth of 0.57 Mbps can be obtained when c is set to 0.02. For comparison, the figure also plots some black points, corresponding to the bandwidth received at the NE under a legacy AP. Despite the fact that the curves have a large flat region, in which the throughput is close to the optimal one, the bandwidth obtained under a legacy AP can be much lower than the maximum value (e.g. 0.46 Mbps for the previous $n_1 = 1$, $n_2 = 10$, and $k_2 = 10$ case).

In order to tune the c parameter maximizing the function $S_{AP}^{NE}(c)$, it is convenient to express each channel access probability τ_j^+ as a function of the channel access probability experienced by a reference station i at the NE:

$$\tau_j^+ = \frac{\tau_i^+}{\tau_i^+ + \frac{k_i x_i}{k_j x_j}(1-\tau_i^+)},$$

where we have inverted the best response expression given in (8) for the reference station i, and substituted $c = \frac{\tau_i^+}{\tau_i^+ + k_i x_i (1-\tau_i^+)}$ in the best response equation of any other station $j \neq i$. It results:

$$\begin{aligned} &S_{AP}^{NE}(c) = S_{AP}^{NE}(c(\tau_i^+)) \\ &= \frac{1}{k_i x_i} \frac{\tau_i^+(1-\tau_i^+)^n P}{T\left[1+\tau_i^+(\frac{1}{k_i x_i}-1)\right]\prod_{j=1}^{n}\left[1+\tau_i^+(\frac{k_j x_j}{k_i x_i}-1)\right]-(1-\tau_i^+)^{n+1}(T-\sigma)} \end{aligned} \quad (9)$$

By deriving (9), it can be shown that (for $k_j \neq 0 \ \forall j$) the function S_{AP}^{NE} has a unique maximum in $\tau_{i_o} \in (0,1)$. Such desired maximum can be obtained by setting a specific $\tau_{AP_o} = c(\tau_{i_o})$ value. Although a closed form expression for such a maximum is not trivial, we verified that an excellent approximation for $k_j > 1 \ \forall j$ is given by:

$$\hat{\tau}_{AP_o} = \frac{1}{(1+\sum_j k_j x_j)\sqrt{T/2\sigma}} \quad (10)$$

which leads to $\hat{\tau}_{i_o} = \frac{k_i x_i}{(1+\sum_j k_j x_j)\sqrt{T/2\sigma}-(1-k_i x_i)}$. The approximation is based on the result shown in [7], according to which the optimal channel access probability for a network with n competing stations is given by $\frac{1}{n\sqrt{T/2\sigma}}$. In our scenario, at the NE outcome, the AP behaves as a single contending station, while all the others require an uplink throughput equal to $k_j x_j$ times the AP one.

Figure 6 plots some examples of the NE utilities J_1^{NE} perceived by station 1, competing with $n_2 = 5$ stations whose application requirement is $k_2 = 1$, for $P = 1500$ bytes and different k_1 values. The figure has been obtained in the case of an 802.11b PHY at 11 Mbps and an application-aware scheduling policy. Although this mechanism design scheme cannot be performed when it exists $k_i = \infty$ (since in this case station i is not interested in the download traffic and c cannot be used as a tuning parameter for τ_i^+), Figure 6 also plots the limit curve obtained when $k_1 \to \infty$. In both Figures 5 and 6 the bandwidth perceived when c is tuned to the approximated value (10) is enlightened by empty boxes. The points are quite close to the actual maximum values (as we also verified numerically).

We observe that this scheme could also be presented as a Stackelberg game with the AP as leader and the users as followers. In this case the AP would also be a player, with the same set of strategies, but with a different utility function (the social utility). The resulting Stackelberg equilibrium corresponds to the one obtained by maximizing (9), i.e. to our desired NE.

B. *ACK suppression*

When a contending station i is not interested in downloading, the mechanism design based only the τ_{AP} tuning is not effective for obtaining efficient NE, because in this case station i will maximize its utility by playing $\tau_i^{(br)} = 1$ and all the other stations will receive a null utility.

A solution for controlling the resource repartition in infrastructure networks with stations not requiring downlink throughput is adding a selective discard of the ACK transmissions at the AP side. Since the AP is the common receiver for all stations, suppressing the ACKs at the AP side corresponds to triggering ACK timeouts at the station side, which are interpreted as collisions. Therefore, ACK dropping can act as a punishment strategy devised to limit the uplink throughput of too aggressive stations. We propose the following threshold scheme: if a generic station i has an access probability τ_i higher than a given value γ, the AP drops an ACK frame with probability $\min\{\alpha(\tau_i - \gamma), 1\}$.

In this case, for station i with $k_i = \infty$ the utility function J_i is given by the uplink throughput and can be expressed as:

$$J_i(\tau_i, p_i) = \begin{cases} \frac{\tau_i(1-p_i)(1-\tau_{AP})}{P_{idle}\sigma+[1-P_{idle}]T} & 0 < \tau_i < \gamma \\ \frac{\tau_i(1-p_i)(1-\tau_{AP})[1-\alpha(\tau_i-\gamma)]}{P_{idle}\sigma+[1-P_{idle}]T} & \gamma \leq \tau_i < \gamma + 1/\alpha \\ 0 & \gamma + 1/\alpha \leq \tau_i \leq 1 \end{cases} \quad (11)$$

where we recall that $P_{idle} = (1-\tau_i)(1-p_i)(1-\tau_{AP})$ and τ_{AP} can be zero if $k_i = \infty$ for all i. According to the previous expression, for $\tau_i < \gamma$ the utility function J_i is an increasing function of τ_i, while for $\tau_i \geq \gamma$ its slope depends on the α setting. By selecting an α value which corresponds to a negative derivative of J_i with respect to τ_i, for $\gamma < \tau_i < \gamma + 1/\alpha$, the utility function is maximized for $\tau_i^{(br)} = \gamma$.

We observe that this approach has some similarities with that proposed in [10], [11]. There, the authors consider that a penalty mechanism should be deployed in a distributed way through jamming, and they show that a simple linear control is sufficient to lead the system to work at a desired operation point (as we are going to show for our ACK suppression scheme). We believe that our solution is more appealing from a practical point of view. In fact, a distributed jamming would require that all WiFi cards support this mechanism. On the contrary our ACK suppression scheme requires only some changes to the AP and WiFi cards do not need any change.

Figure 7 plots the station utility perceived in the case of 10 stations requiring uplink traffic only (i.e. $k = \infty$) under the ACK suppression scheme, for different α values. For $\alpha = 0$, the utility is an increasing function of the channel access probability and the best response of station i is $\tau_i^{(br)} = 1$. For $\alpha > 0$, the utility function is maximized for $\tau_i^{(br)} < 1$. Such a maximum corresponds to γ for large enough values of α (in the figure, $\alpha = 80$).

Let the station be ordered for decreasing k_i values and let n_u be the number of stations with only upload traffic (i.e. $k_i = \infty$). We can then prove the following result.

Proposition 3.1: The outcome $\tilde{\boldsymbol{\tau}}$ such that $\tau_i = \gamma$ for $i \in [1, n_u]$ and $\tau_i = k_i x_i \tau_{AP}/(1-(1-k_i x_i)\tau_{AP})$ for $i \in [n_u+1, n]$ is a Nash equilibrium of the game, when the ACK suppression scheme indicated above is implemented with

$$\alpha \geq \frac{1}{\gamma\left(1+\gamma\frac{(1-\gamma)^{n_u-1}(1-\tau_{AP})\prod_{i=n_u+1}^{n}(1-\tau_j)(T-\sigma)}{T-(1-\gamma)^{n_u-1}(1-\tau_{AP})\prod_{i=n_u+1}^{n}(1-\tau_j)(T-\sigma)}\right)} \tag{12}$$

Moreover, when $k_i = \infty \ \forall i$ and $\gamma \leq \tau_x$ (the value in Remark 2.3) the NE is also Pareto Optimal.

Proof: First we observe that $\tilde{\boldsymbol{\tau}}$ is a NE. Indeed, whatever player i we consider with $i \leq n_u$, Remark 2.1 guarantees that for $\tau_i < \gamma$ J_i decreases as τ_i decreases. For $\gamma < \tau_i < \gamma + 1/\alpha$ inequality (12) guarantees that J_i decreases as τ_i increases until it reaches the value 0. For $\tau > \gamma + 1/\alpha$, the punishment strategy implies $J_i = 0$. Then deviating from $\tilde{\boldsymbol{\tau}}$ is not convenient for player i. For each other station j with $j > n_u$, $\tau_j = k_j x_j \tau_{AP}/(1-(1-k_j x_j)\tau_{AP})$ is the station best response (8), which is fixed for a given τ_{AP} setting. Then, also for these stations it is not convenient deviating from $\tilde{\boldsymbol{\tau}}$.

Second, when $n_u = n$, the NE is $\boldsymbol{\gamma} = (\gamma, \gamma, \ldots \gamma)$. Considering the subset of outcomes $[0, \gamma]^n$, we can reason as in Prop. 2.3 and show that the social utility (defined as the minimum of stations' utilities as in Sec. II-E) is maximized in this subset at a unique homogeneous outcome. Moreover, for every outcome in $[0,1]^n - [0,\gamma]^n$, there is at least one station with $\tau_i > \gamma$ and this station gets a smaller utility than at the NE $\boldsymbol{\gamma}$. We can then conclude that there is a unique homogeneous outcome maximizing the social utility and that it lies in $[0,\gamma]^n$. Considering the value τ_x introduced in Remark 2.3, if $\tau_x < \gamma$ the homogeneous outcome is $(\tau_x, \tau_x, \ldots, \tau_x)$, while if $\tau_x >= \gamma$ it is $\boldsymbol{\gamma}$. Then, under the assumed hypotheses, $\boldsymbol{\gamma}$ maximizes the social utility and is Pareto optimal. ■

Note that the utility perceived at the NE point depends on the settings of both τ_{AP} and γ. Under the scheduling policy given in (7), the per-station bandwidth maximization given in (10) can be written as:

$$\hat{\tau}_{AP_o} = \frac{\sum_{j=n_u+1}^{n}\frac{1}{(k_j+1)}}{n\sqrt{T/2\sigma}} \tag{13}$$

which corresponds to a per-station best response

$$\hat{\tau}_{i_o} = \frac{\frac{k_i}{k_i+1}}{n\sqrt{T/2\sigma}+1-\sum_{j=n_u+1}^{n}\frac{1}{k_j+1}} \tag{14}$$

Since for the stations employing $k_i = \infty$ the uplink bandwidth is equal to the total perceived bandwidth (i.e. $k_i/(k_i+1) \to 1$), a possible tuning strategy (maximizing the per-station total bandwidth) is tuning τ_{AP} to (13) and γ to $\frac{1}{n\sqrt{T/2\sigma}+1-\sum_{j=n_u+1}^{n}\frac{1}{k_j+1}}$. When $n_u = n$, the previous expression becomes $\gamma = \frac{1}{n\sqrt{T/2\sigma}+1}$, which is similar to the approximation proposed in [7].

IV. Game-based MAC scheme: Implementation and Evaluation

On the basis of the results discussed in the previous sections, we propose some simple DCF extensions devised to i) enable each contending station to dynamically tune its channel access probability according to a best response strategy; ii) enable the AP to act as a game designer to induce some desired equilibrium conditions. Being n the number of stations associated to the AP, we assume that the AP maintains n independent downlink queues. For each station i, uplink and downlink transmission queues are always saturated, apart from the case $k_i = \infty$ when the i-th downlink queue is empty. We also assume that each station is aware of its application requirements k_i, while the AP is aware of the number of associated stations n involved in the contention process.

A. Estimators at the Station and AP side

In actual networks, for implementing a best response strategy, each station needs to estimate the AP channel access probability τ_{AP}. Moreover, for implementing the mechanism design and scheduling policies described in the previous sections, the AP needs to estimate the channel access probability τ_i employed by each station and the per-station application requirements k_i[5]. All these parameters can be estimated on the basis of channel observations.

Considering the slotted channel model due to saturation conditions, a channel observation corresponds to the channel outcome observed into a given slot. Such outcome is given by an idle slot when no station transmits, by a successful slot when a single station transmits, by a collision slot when two or more stations transmit simultaneously. In order to perform run-time estimators, the channel observations can be grouped in observation intervals at which new measurement samples are available. We express the measurement intervals in terms of an integer number B of channel slots. Since the slot size is uneven (because successful slots and collisions last for a T time, while idle slots last only for σ), the actual time required for a new measurement sample is not fixed.

In each interval, a monitoring station cannot count the total number of transmissions performed by the access point, because in a collision slot it is not possible to detect the identity of the stations involved in the collision. Therefore, we implemented an access probability estimator based on the counting of the number of idle slots s and the number of successful transmissions tx_{AP} performed the AP. Let $P_{s_{AP}}$

[5]Although stations could in principle notify their application requirements, we prefer to consider an independent estimate carried out by the AP for avoiding malicious false notifications.

the probability to have a successful AP transmission on the channel. Since $P_{s_{AP}} = \tau_{AP}(1 - p_{AP}) = \frac{\tau_{AP}}{1-\tau_{AP}} P_{idle}$, we have that the actual τ_{AP} value can be expressed as $P_{s_{AP}}/(P_{s_{AP}} + P_{idle})$ and a $\tau^m_{AP}(t)$ measure in the t-th time interval can be evaluated as $\frac{tx_{AP}(t)}{tx_{AP}(t)+s(t)}$. Similarly, during the observation interval B, the AP can separately count the successful transmissions tx_i performed by each station i, for measuring $\tau^m_i(t)$ as $\frac{tx_i(t)}{tx_i(t)+s(t)}$. Being tx^i_{AP} the number of successful transmission performed by the AP for station i, a measurement of the downlink ratio $x^m_i(t)$ is simply given by tx^i_{AP}/tx_{AP}.

As far as concerns the k_i estimates, when the i-th downlink queue is not-empty, the AP can perform an estimation of the application requirements by considering that at the NE k_i is equal to the throughput ratio $\frac{S^i_{up}}{S^i_{down}}$. In the assumption of fixed packet size, such a ratio can be expressed as the ratio between the successful transmissions tx_i performed by station i and the successful transmissions tx^i_{AP} performed by the AP for station i, i.e. $k^m_i(t) = tx_i(t)/tx^i_{AP}(t)$. Obviously, the number tx^i_{AP} depends on the scheduling policy implemented at the AP side, which in turns might depend on the current k_i estimates. It follows that also the x_i coefficients might be time-dependent and updated at each estimation interval B. When the i-th downlink queue is empty, the AP can immediately understand that the station application does not require downlink bandwidth, i.e. $k_i = \infty$.

Finally, the estimates $\hat{\tau}_{AP}$, $\hat{x}_i$, $\hat{k}_i$, and $\hat{\tau}_i$ are performed by smoothing the measurements τ^m_{AP}, x^m_i, k^m_i and τ^m_i with a filter. In our simulations, we used a first-order autoregressive filters for all the parameters, whose memory coefficient has been set to 0.75.

B. Best response performance under legacy AP

As discussed in Sec. II, in the case of unidirectional upload traffic, the best response strategy leads to very poor throughput performance under legacy AP. Therefore, in this section we consider $k_i \neq \infty$, $\forall i$.

A generic station i may implement a best response strategy, on the basis of the previous estimators and (6), by setting its channel access probability to:

$$\tau^{(br)}(t+1) = \frac{k_i \hat{x}_i(t) \hat{\tau}_{AP}(t)}{1 - (1 - k_i \hat{x}_i(t))\hat{\tau}_{AP}(t).} \qquad (15)$$

Although an analysis of the estimate noise effects on the system and equilibrium performance is also possible (as described in [26]), we have evaluated the effectiveness of the presented scheme (approximating the performance of an ideal best response in which all stations exactly know the τ_{AP} parameters and x_i is evaluated with the actual k_i values) by means of simulations. We have extended the custom-made C++ simulation platform used in [7], for a 802.11g physical rate, with the data rate set to 6Mbps. The contention windows used by the AP have been set to the legacy values $CW_{min} = 16$ and $CW_{max} = 1024$. All the simulation results have been obtained by averaging 10 different simulation experiments lasting 10s, leading to a confidence interval lower than 3%. Unless otherwise specified, the measurement interval B has been set to 500 channel slots.

Figure 8 compares the behavior of our scheme with standard DCF. Each point refers to a network scenario in which n stations (indicated in the x axis), with uniform application requirements $k_i = k$ $\forall i$, compete on the channel with a legacy AP. The aggregated uplink throughput (i.e. the sum of the throughput perceived by all the mobile stations) and k times the aggregated downlink throughput, (i.e. the AP throughput) are indicated by the y axis, respectively by white and black points. From the figure, it is evident that, as the number of contending stations increases, standard DCF gives very poor performance to the downlink throughput. Conversely, for $k = 1$ our scheme is able to equalize uplink and downlink throughput for each n, and even in congested network conditions. Moreover, it is also able to maintain the overall network throughput (i.e. the sum of the aggregated uplink and downlink throughput) almost independent on the network load. For example, for $n = 20$ the sum of the uplink and downlink throughout is about 3.8 Mbps for standard DCF and about 5 Mbps for our scheme. The figure also shows our scheme effectiveness for different application requirements (i.e. $k = 0.5$). The figure clearly visualizes that $\sum_i S^i_u = knS^i_d$, as expected. Note also that our scheme is different from a classical prioritization scheme, such as the schemes defined in the EDCA extensions [5]. Indeed, by giving lower contention windows to the AP (i.e. a higher EDCA priority class to the AP), it is not possible to perform a desired resource repartition between uplink and downlink which is also load independent.

We have also checked our scheme performance when application requirements are time-varying, by running several simulation experiments in which the k_i coefficients dynamically change during the simulation time. Figure 9 shows a simulation example lasting 750 seconds, in which two contending stations (station 1 and station 2) have initially the same application requirements $k_1 = k_2 = 1$, and station 2 changes temporarily these requirements to $k_2 = 5$ in the time interval [250s, 500s]. In the figure we plot the uplink and downlink throughput perceived by each station (labeled as Sta1 and Sta2) under the application-aware scheduling policy, when the AP estimates the k_i coefficients according to the estimators introduced in (IV-A) (in the interval [0s, 500s]) and when the AP knows the exact k_i values (in the interval [500s, 750s]). Since we initialized our estimation process with $k_1(0) = k_2(0) = 0$, it also follows that the scheduling starts with $x_1 = x_2 = 1/2$, i.e. by equally sharing the downlink throughput among the stations. When the k_2 value changes to 5, after a transient phase of a few tens of seconds, station 2 downlink throughput is reduced, in order to provide an equal aggregated bandwidth to both the stations. When k_2 comes back to 1, the downlink throughput is again equally shared among the stations and the transient phase is much quicker because in this case we assumed that the AP knows the exact k_i values. Note that the throughput fluctuations in the range [0s, 250s] and [500s, 750s] are comparable, thus proving that the noise on the k_i estimates does not critically affect the network bandwidth repartition.

C. Best response performance under AP mechanism design

The implementation of the optimal tuning of the τ_{AP} probability can be easily supported in actual networks, by using the approximated optimal value given given in (10). Such a value depends only on the k_i estimates, which we have previously introduced to enable application-aware scheduling policies.

Figure 10 plots the overall bandwidth (i.e. $\sum_{i=1}^{n} S_u^i + S_d^i$) available in the network under the application-aware scheduling policy, in case of two application classes ($k_1 = 1$ and $k_2 = 10$), as a function of the per-class number of stations $n_1 = n_2$. The figure compares the performance obtained when the AP behaves as a legacy station and when the AP adaptively tunes its τ_{AP} parameter according to the approximated optimal value given in (10). For $n_1 = n_2 = 20$, the bandwidth available under legacy AP is 10% lower than the one available in case of adaptive τ_{AP} tuning. Indeed, as evident in Figure 5, there is a wide range of τ_{AP} values which provide performance quite close to the optimal one.

As far as concerns the implementation of the ACK suppression scheme (to be considered when $\exists k_i = \infty$), it is necessary to configure: i) the γ threshold, which depends on the k_i estimates and on the known parameter n; ii) the α coefficient, which is simply related to γ and to the PHY parameters T and σ; iii) the per-station channel access probability τ_i, $i = 1, 2, \cdots n_u$ estimates. All these configurations may rely on the estimators already introduced in (IV-A). The implementation of the ACK suppression scheme at the AP side has important implications for preventing users and card manufacturers from using non-standard contention window values. As proved in [3], currently there is an impressive proliferation of cheating cards, i.e. cards which implement lower contention windows to gain advantage during the contention with other cards.

Figure 11 shows a simulation example (reproducing one of the realistic scenarios documented in [3]), in which a cheater card with a contention window equal to 8 compete with one legacy card. Both the stations are interested in upload traffic only. The figure refers to a simulation experiment lasting 105 seconds, after a transient phase of 10 seconds. Despite of the temporal fluctuations, it is evident that the cheating card obtains a throughput (dashed line) higher than two times the throughput (bold line) perceived by the legacy card. Also, Figure 11 plots the throughput performance of the two stations when the AP implement the ACK suppression scheme. In this case, the cheater is no longer motivated to use a channel access probability higher than the contending station, since its throughput is maximized tuning the channel access probability to the threshold value $1/(2\sqrt{T/2\sigma}+1)$ (i.e. implementing a best response strategy). The figure shows the throughput performance of the two stations implementing the best response (bold and dashed lines labeled as best response), and the throughput degradation perceived by the cheater station by still using a contention window equal to 8. The ACK suppression scheme works properly, even if the AP relies on the channel access probability estimators, rather than on the actual values.

In order to asses the effectiveness of the ACK suppression implementation as the number of competing stations grows, Figure 12 compares the aggregated network throughput of our scheme with the standard DCF one, for different n values. Each point refers to a network scenario in which n stations (indicated in the x axis) are aware of the ACK suppression risk and employ a consequent best response strategy. Although the variance of the τ_i estimators could imply that in some intervals such a probability passes the threshold value (i.e. there is a non null probability of unnecessary ACK dropping), the figure shows that the aggregated throughput is almost constant regardless of the number of competing stations. This behavior is very different from standard DCF, whose efficiency depends on the number of contending stations and degrades for high load conditions. Therefore, our scheme is able not only to discourage cheating card behaviors, but also to optimize the global network performance.

V. Conclusions

The proliferation of MAC-level programmable WiFi cards can potentially create serious coexistence problems, since some stations could implement greedy access policies to increase their bandwidth share at the expenses of compliant users. For this reason, we have proposed a game-theoretic analysis of persistent access schemes for WiFi infrastructure networks, in order to characterize equilibria conditions and to design disincentive mechanisms for inefficient behaviors. We have proved that, when stations are interested in both uploading and downloading traffic, it exists a Nash Equilibrium where all the stations reach a non-null utility. Moreover, we have also explored the utilization of the Access Point as an arbitrator for improving the global network performance. Specifically, we have proposed two different solutions. When all stations require downlink traffic, the AP can tune its channel access probability to control the station best responses and optimizing the overall network capacity. When some stations are interested in uplink traffic only, the AP can selectively discard the acknowledgments of too greedy stations.

We have then proposed some extensions to standard DCF, in order to estimate the network status, in terms of per-station application requirements and channel access probability, and emulate an access scheme based on best response strategies and AP mechanism design. We proved the effectiveness of our solutions in controlling the resource sharing for WiFi networks in various network scenarios. Currently, we are investigating on the prototyping of our solutions in actual WiFi cards and APs. While the estimate and best response modules can be simply implemented at the driver level, the ACK dropping scheme requires a hardware/firmware update.

References

[1] IEEE Standard 802.11 - 1999; Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications; November 1999.

[2] Wi-Fi Alliance, www.wi-fi.org

[3] G. Bianchi, A. Di Stefano, C. Giaconia, L. Scalia, G. Terrazzino, I. Tinnirello, "Experimental assessment of the backoff behavior of commercial IEEE 802.11b network cards", *Proc. of IEEE Infocom 2007,* pp. 1181-1189, 2007.

[4] S.W. Kim, B.S. Kim, Y. Fang, "Downlink and uplink resource allocation in IEEE 802.11 wireless LANs", *Proc. of IEEE Conf. on Vehicular Technology*, vol. 54, pp. 320-327, 2005.

[5] IEEE 802.11e Supplement to Part 11: Wireless Medium Access Control (MAC) and Physical Layer Specification: Medium Access Control (MAC) Enhancements for Quality of Service (QoS)", October 2005.

[6] L. Zhao, L. Cong, H. Zhang, W. Ding, J. Zhang, "Game-Theoretic EDCA in IEEE 802.11e WLANs", *Proc. of IEEE VTC 2008*, pp. 1-5, 2008.

[7] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function", *IEEE Journal of Selected Areas in Communication*, vol. 18, no. 3, pp.535-547, 2000.

[8] L. Giarré, I. Tinnirello, and G. Neglia, "Medium Access in WiFi Networks: Strategies of Selfish Nodes", *IEEE Signal Processing Magazine (SPM)*, vol. 26, issue 5, pp. 124-127, 2009.

[9] J. Konorski, "A Game-Theoretic study of CSMA/CA under a backoff attack", *IEEE/ACM Trans. on Networking*, vol. 14, issue 6, pp. 1167-1178, 2006.

[10] M. Cagalj, S. Ganeriwal, I. Aad, J.P. Hubaux, "On selfish behavior in CSMA/CA networks", *Proc. of IEEE Infocom*, vol. 4, pp. 2513-2524, 2005.

[11] M. Felegyhazi, M. Cagalj, J.P. Hubaux, "Efficient MAC in cognitive radio systems: a game-theoretic approach", *Transactions on Wireless Communcations*, Vol. 8, no. 4, pp. 1984-1995, 2009.

[12] L. Chen, S.H. Low, J. Doyle, "Contention Control: A Game-Theoretic Approach", *Proc. of IEEE Conf. On Decision and Control*, pp. 3428-3434, 2007.

[13] T. Cui, L. Chen, S. Low, "A Game-Theoretic framework for medium access control", *IEEE JSAC*, vol. 26, issue 7, pp. 1116-1127, 2008.

[14] G. Zhang, H. Zhang, "Modeling IEEE 802.11 DCF in wireless LANs as a dynamic game with incompletely information", *Proc. of Conf. on Wireless, Mobile and Multimedia Networks*, pp. 215-218, 2008.

[15] L. Chen, J. Leneutre, "Selfishness, not always a nightmare: modeling selfish MAC behaviors in wireless mobile ad-hoc networks", *Proc. of IEEE ICDCS 2007*, pp. 16-23, 2007.

[16] D.K. Sanyal, M. Chattopadhyay, S. Chattopadhyay, "Improved performance with novel utility functions in a game-theoretic model of medium access control in wireless networks", *Proc. of IEEE TENCON 2008*, pp. 1-6, 2008.

[17] I. Tinnirello, L. Giarré and G. Neglia, "The Role of the Access Point in Wi-Fi Networks with Selfish Nodes", *Proc. of IEEE GameNets*, Instanbul, pp. 632-637, 2009.

[18] L. Giarré, G. Neglia and I. Tinnirello "Resource sharing optimality in WiFi infrastructure networks", *IEEE Proc. of Conference on Decision and Control*, pp. 5877-5882, 2009.

[19] J. Konorski, "A station strategy to deter backoff attacks in IEEE 802.11 LANs", *Elsevier Journal of Discrete Algorithms*, Vol. 5, no. 3, pp. 436-454, 2007.

[20] Lei Guang, Chadi Assi, "Cross-layer cooperation to handle MAC misbehavior in ad hoc networks", *IEEE Proc. of CCECE 2006*, pp. 219-222, 2006.

[21] R. Maxim, J.P. Hubaux, I. Aad, "DOMINO: A System to Detect Greedy Behavior in IEEE 802.11 Hotspots", 2nd international conference on Mobile systems, applications, and services (MobiSys), 2004.

[22] M. Raya, I. Aad, J.P. Hubaux, A. El Fawal,"DOMINO: Detecting MAC layer greedy behavior in IEEE 802.11 hotspots", *Transactions on Mobile Computing*, Vol. 5, no. 12, 2006.

[23] L. Guang, C. Assi, Y. Ye, "DREAM: A system for detection and reaction against MAC layer misbehavior in ad hoc networks", *Computer Communications*, Vol. 30 , no. 8, pp 1841-1853,2007.

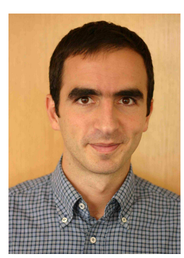
[24] G. Bianchi, I. Tinnirello "Remarks on IEEE 802.11 Performance Analysis", *IEEE Communication Letters*, Vol. 9, no. 8, pp. 765-767, 2005.

[25] D.R. Cox, "Renewal Theory", Methuen and Co., 1970.

[26] I.Tinnirello, L. Giarré, G. Neglia, A game theoretic approach to MAC design for infrastructure networks, *Proc of IEEE CDC*, pp. 1933-1939, 2010.

Ilenia Tinnirello has been Assistant Professor at the University of Palermo since January 2005. She received the Laurea degree in Electronic Engineering and the Ph.D. on Communications, respectively in April 2000 and February 2004. She has also been Visiting Researcher at the Seoul National University, Korea, in 2004, and at the Nanyang Technological University of Singapore in 2006. Her research activity has been mainly focused on wireless networks and in particular on: multiple access algorithms with quality of service provisioning; cross-layer interactions between access solutions and physical layer; mobility management and load balancing in wireless packet networks. I. Tinnirello has been involved in several international and national research projects, among which the bilateral Italian-Korean research project INFINITY and the national research project PRIN MIMOSA, as scientific coordinator for the research unit of Palermo, and the European project FP7 FLAVIA, as technical coordinator.

Laura Giarré received the Laurea degree in Electronic Engineering from the Universitá di Firenze, Italy and the Ph.D. degree in System Engineering from the Universitá di Bologna, Italy, in 1986 and 1992, respectively. She has held visiting positions at the Department of Mechanical Engineering of the University of California at Santa Barbara and at the Laboratory for Information and Decision Systems, MIT, Boston. Since 1993 she was Assistant Professor at Politecnico di Torino, Turin, Italy and since 1998 she has been Associate Professor at the Universitá di Palermo, Italy. She serves as an Associate Editor of Systems and Control Letters, since 1999. In 2010 she was vice-chair of SING6. Her research interests include networked systems, according to a game theoretical approach with applications to WiFi networks, sensor networks,inventory networks. She worked also on identification of LPV and block oriented systems according to the worst-case paradigm with application to networked systems and system biology.



Giovanni Neglia is a researcher in Maestro team at INRIA. He received a master's degree in electronic engineering and a Ph.D. degree in telecommunications from the University of Palermo, Italy, in 2001 and 2005, respectively. In 2005 he was a research scholar at the University of Massachusetts Amherst, visiting the Computer Networks Research Group. Before joining INRIA as a permanent member in September 2008, he was a postdoc at the University of Palermo (from June 2005) and an external scientific advisor in Maestro team (from September 2006). His research focuses on modeling and performance evaluation of computer networks and proposals of new mechanisms to improve their performance. To this end he applied different mathematical tools (Markov processes, control theory, fluid models, game theory, gradient-based and metaheuristic optimization).

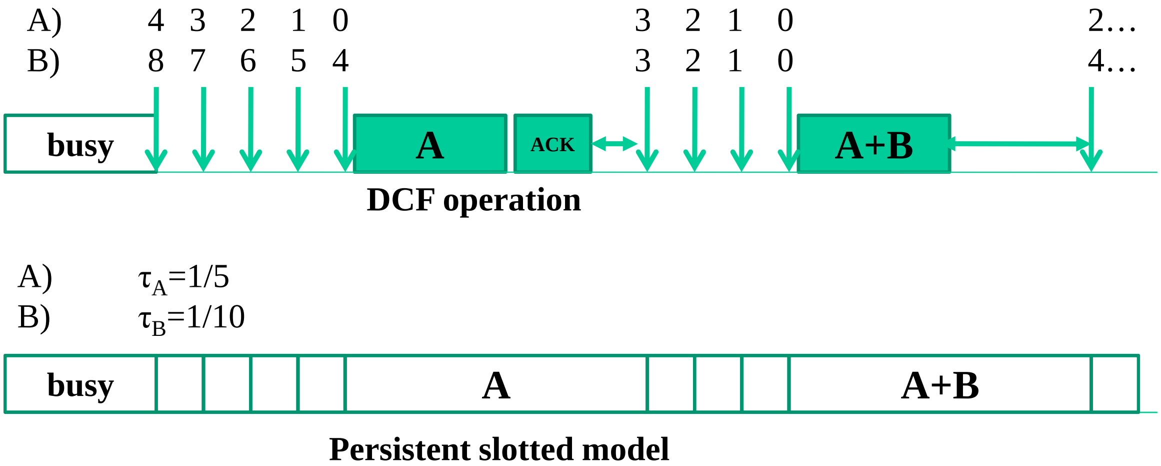


Fig. 1. An example of DCF operation and equivalent slotted model.

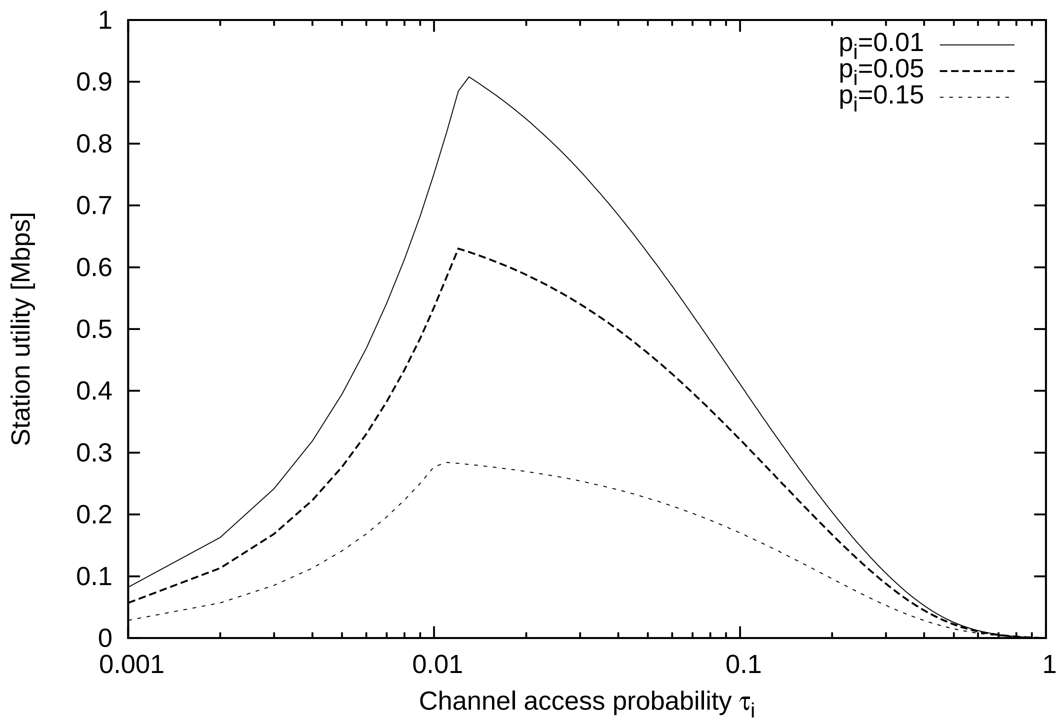


Fig. 2. Utility of a given station i, for different p_i values, as a function of the strategy τ_i ($k = 1$).

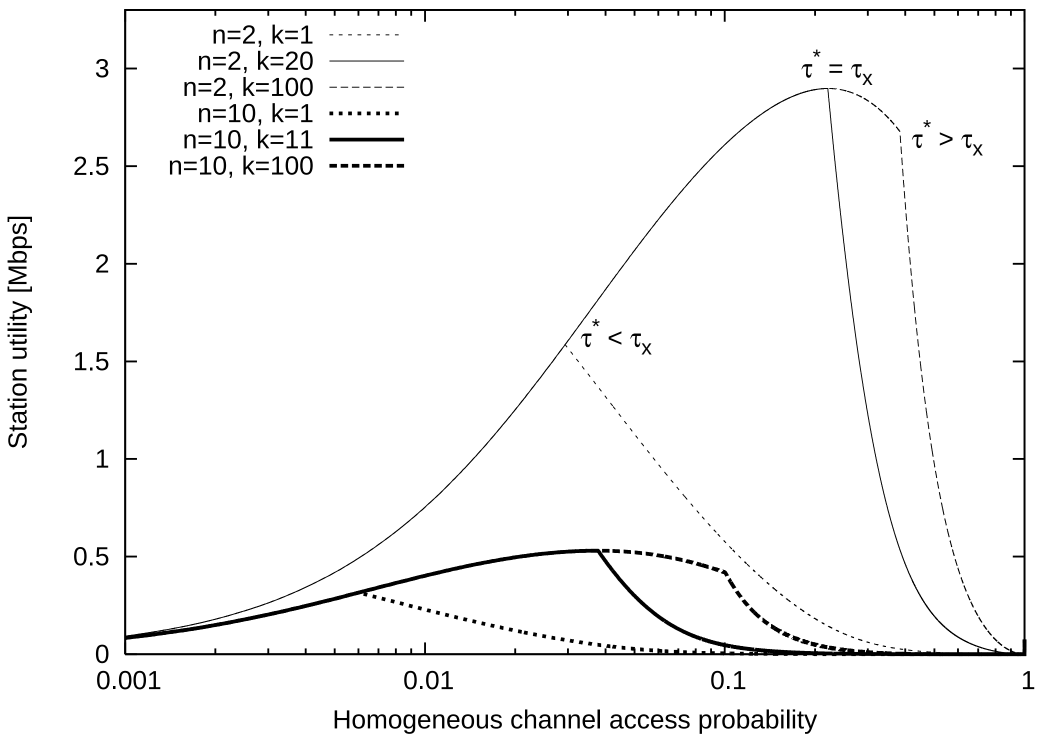


Fig. 3. Station utility in the case of homogeneous access probability employed by all the stations and different k values.

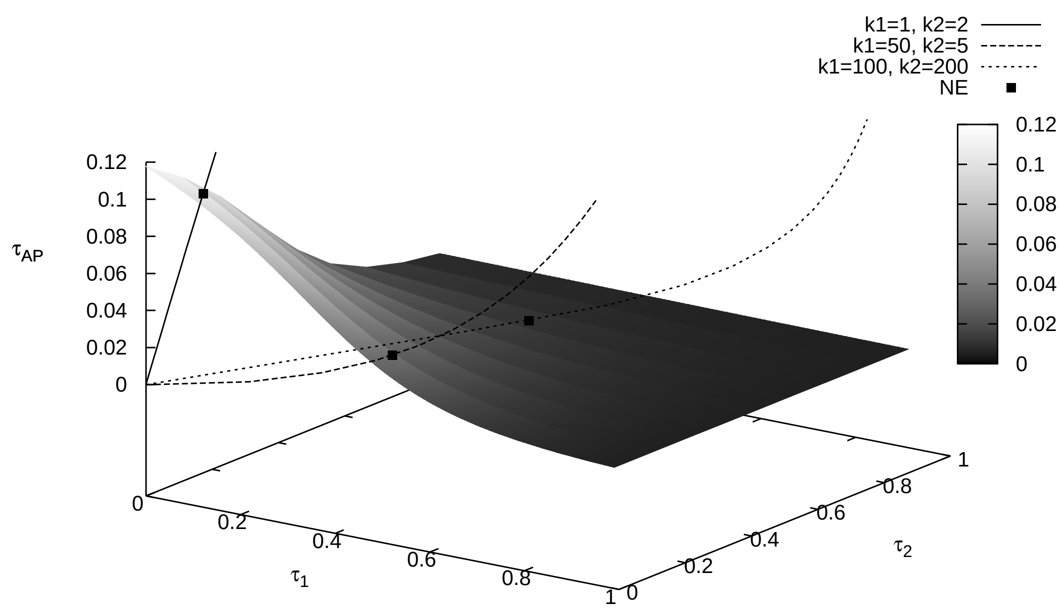


Fig. 4. Geometric interpretation of the Nash equilibrium for heterogeneous application requirements.

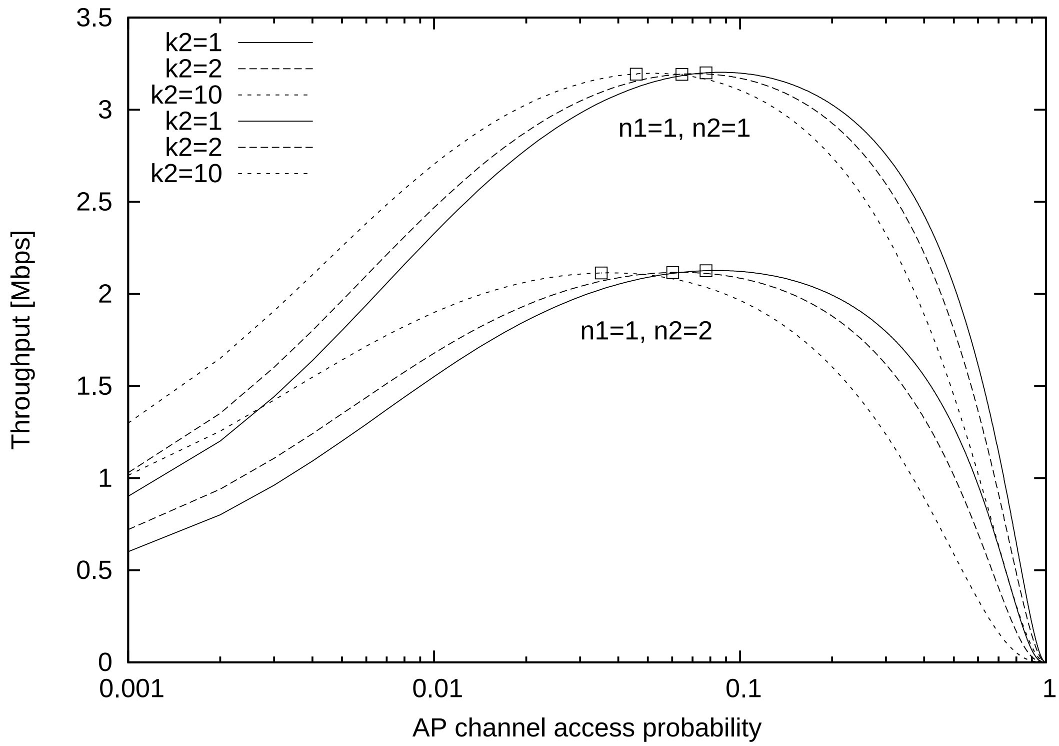


Fig. 5. Per-station total bandwidth, for different application classes (k_1=1, $k_2 = 1, 2, 10$). Comparison between approximated maximum values (empty boxes) and values perceived under a legacy AP (black boxes).

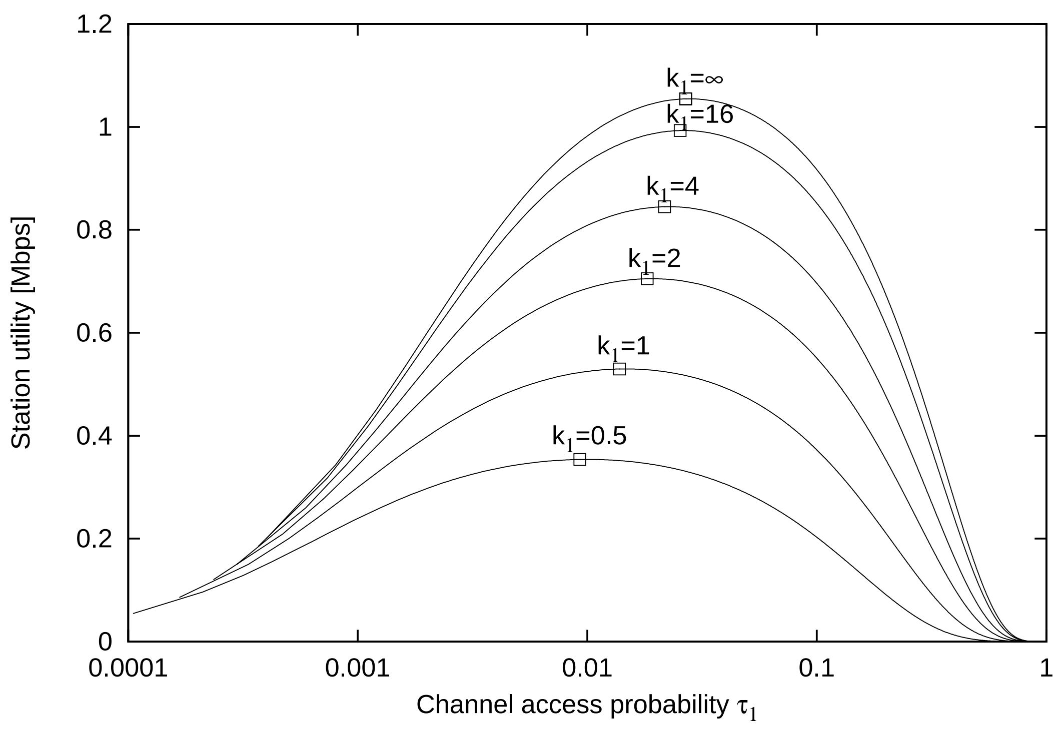


Fig. 6. Station utility at NE as a function of the desired NE outcome, for $n_1 = 1$, $n_2 = 5$, $k_2 = 1$ and different k_1 values. Approximated maximum values are indicated by empty boxes.

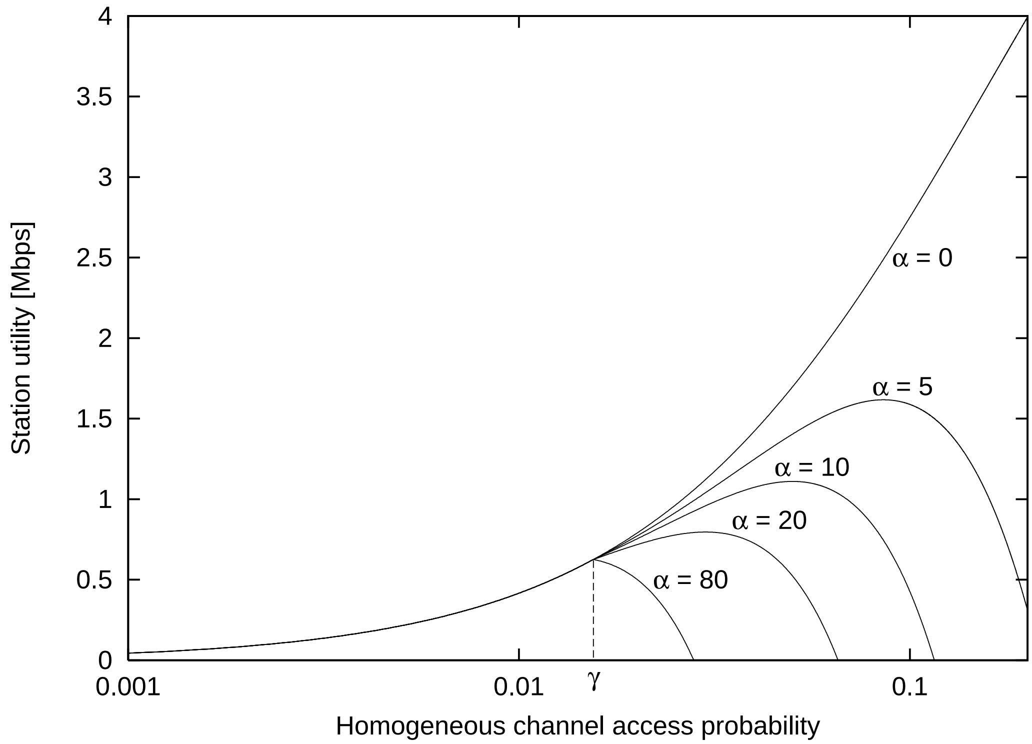


Fig. 7. Station utility in the case of ACK suppression, for n=10, k=∞ and different α values.

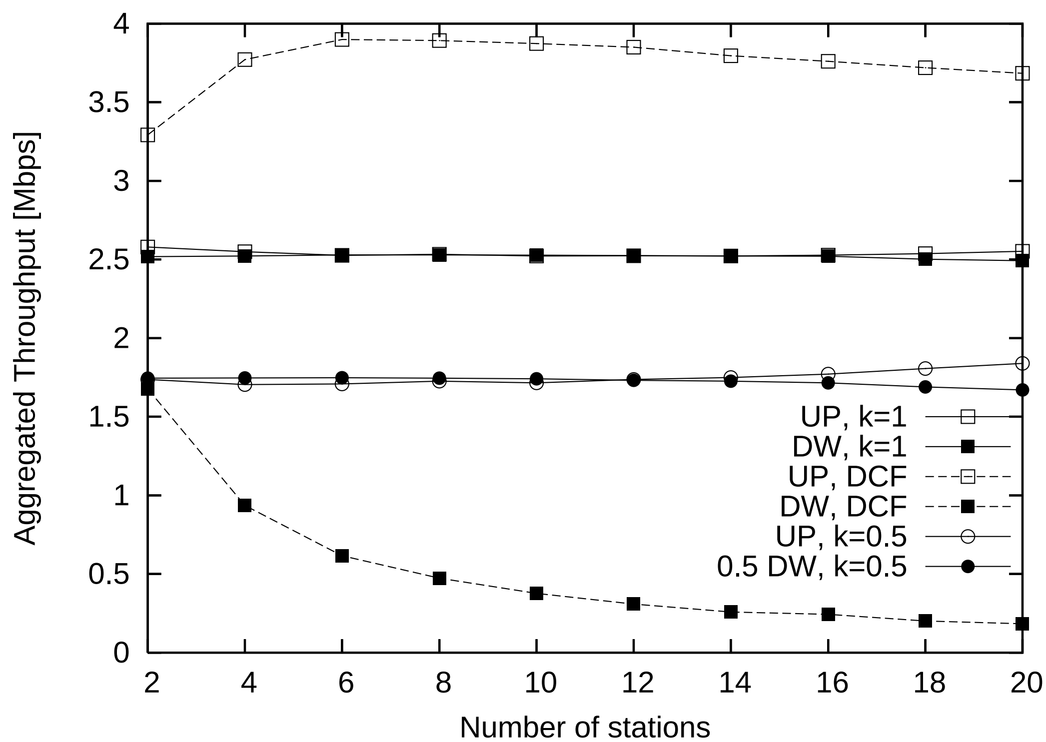


Fig. 8. Aggregated throughput for various number of nodes. Comparison of our scheme ($k = 1$, $k = 0.5$) with standard DCF.

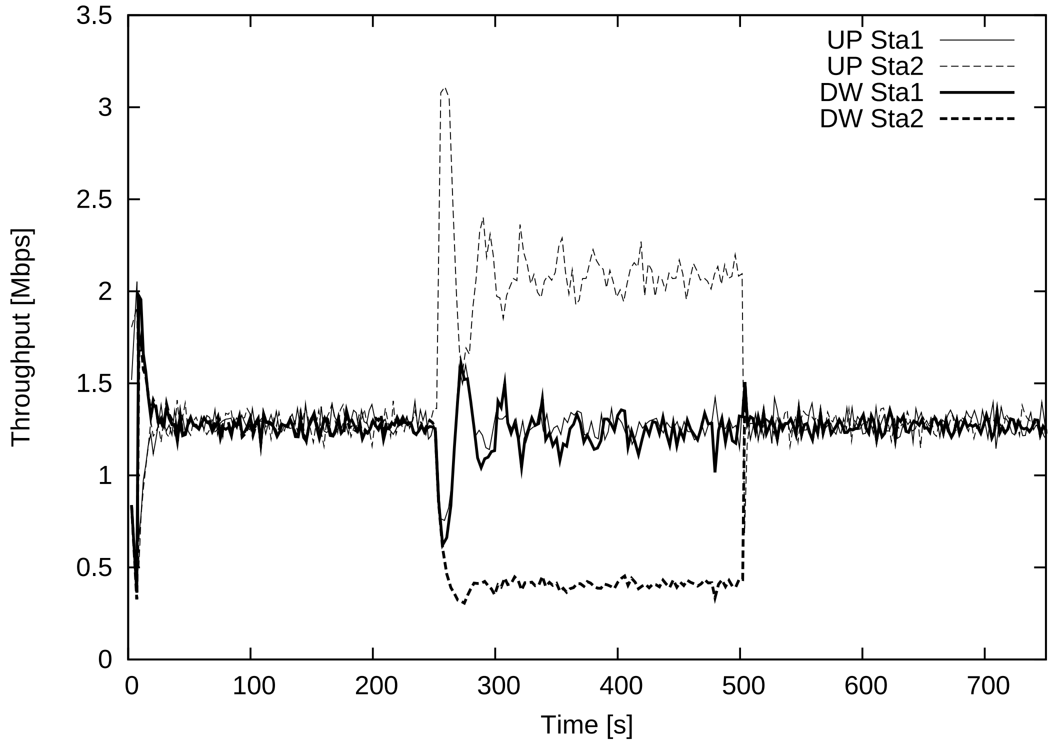


Fig. 9. Effects of best response strategies on the downlink and uplink throughput of two stations employing $k_1 = 1$ in [0s, 750s], $k_2 = 1$ in [0s, 250s] $\cup$ [500s, 750s], and $k_2 = 5$ in [250s, 500s].

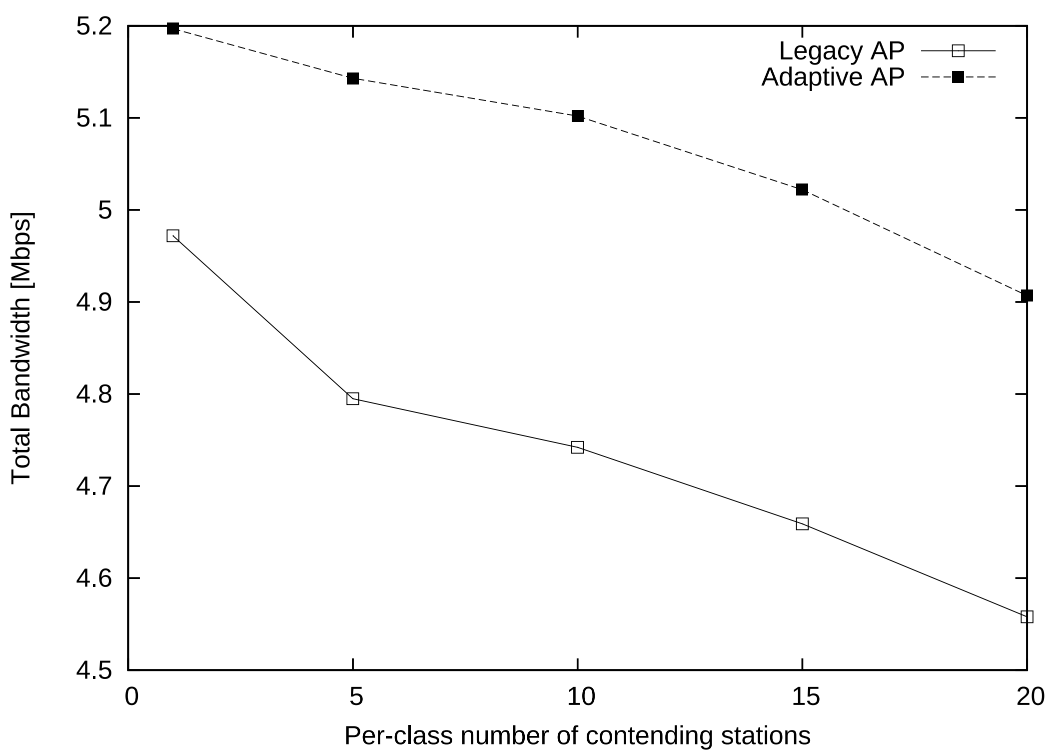


Fig. 10. Total bandwidth available in the network under the application aware scheduling policy, for two application classes ($k_1 = 1$, $k_2 = 10$). Legacy AP (empty boxes) vs. adaptive τ_{AP} optimal tuning (black boxes).

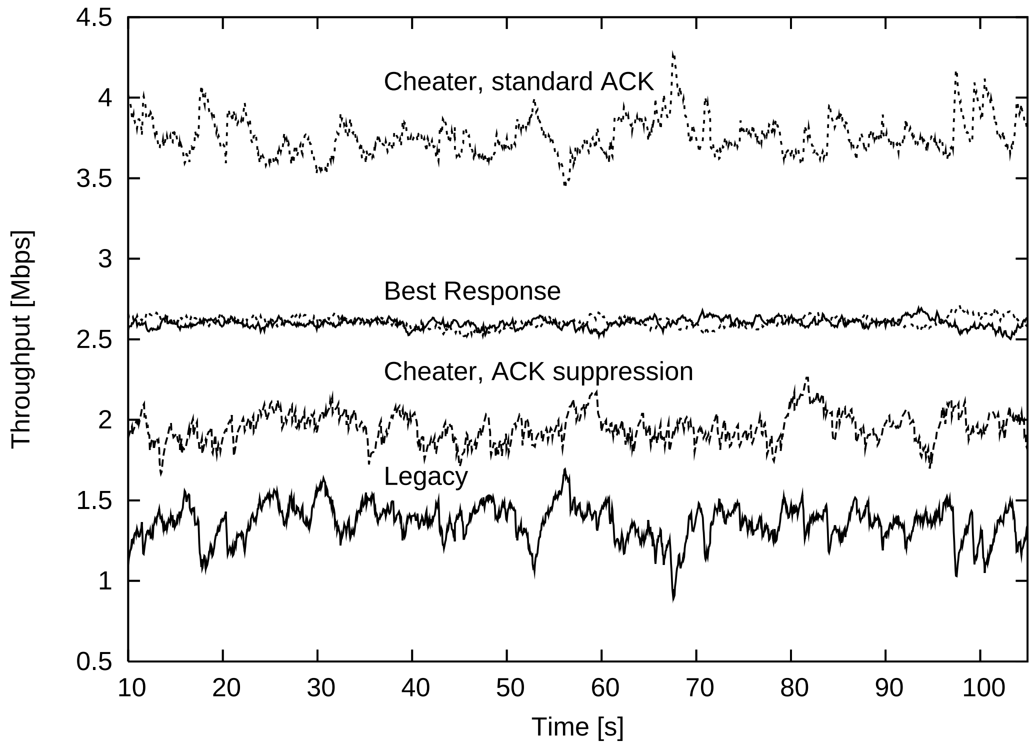


Fig. 11. Effects of the ACK suppression scheme on a cheater station throughput.

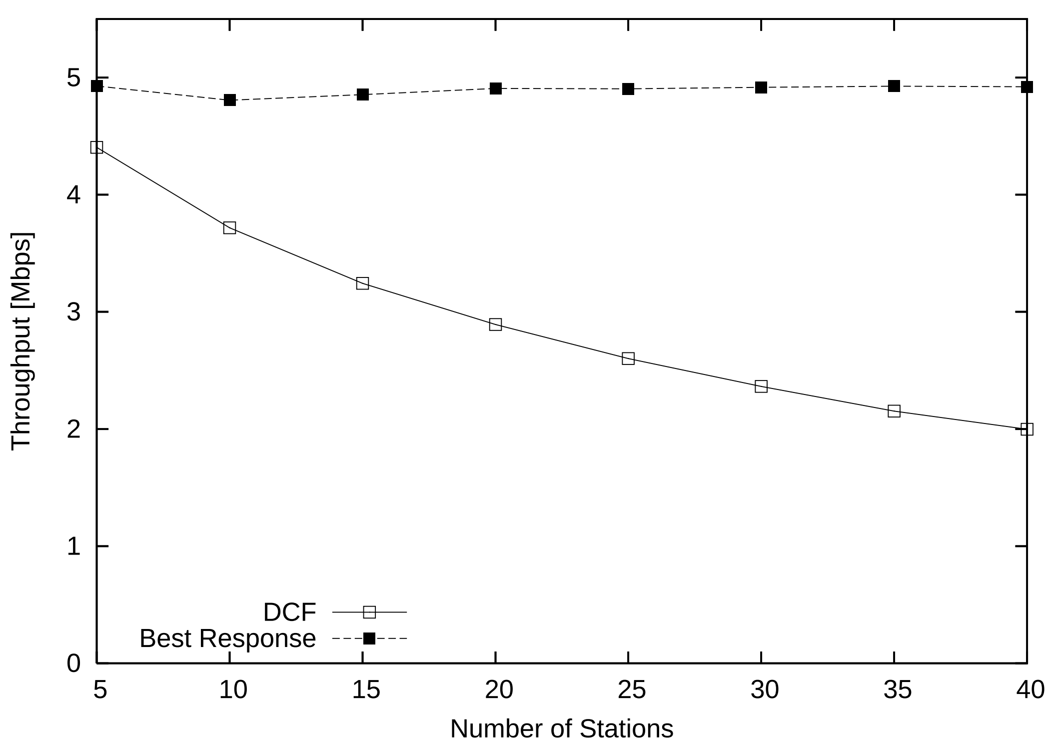


Fig. 12. Unidirectional case ($k = \infty$): aggregated throughput for various number of nodes. Best response is compared with standard DCF.