Resource allocation over multirate wireless networks: a Network Utility Maximization perspective

Andrés Ferragut*, Fernando Paganini

Facultad de Ingeniería, Universidad ORT Uruguay

Abstract

Wireless local area networks, in particular those based in the IEEE 802.11 standard, are becoming increasingly prevalent to deliver Internet access. One of the features these networks introduce is the use of multiple transmission rates in the physical layer; in this regard, a crucial performance issue is how this capability interacts with higher layer protocols such as TCP to determine resource allocation among competing users.

In this paper, we use the Network Utility Maximization framework to characterize the cross-layer interaction between TCP and an underlying MAC layer with multirate capabilities; the result, for current wireless networks, shows a significant bias against users with high modulation rates, to the point of overriding the high-speed feature. Based on this recognition, we propose an alternative resource allocation that overcomes this bias, and simple mechanisms to impose these more efficient equilibria in single cell scenarios and wireless access systems.

We implement these proposals at the packet level in ns2, and present simulations of these mechanisms in action in concrete IEEE 802.11 networks.

Key words: IEEE 802.11, Resource Allocation, Performance Evaluation.

1. Introduction

Wireless local area networks (WLANs) based on IEEE 802.11 [1] are nowadays present in most networking deployments around the world. WLAN hotspots are shared by multiple users at a time through Medium Access Control (MAC) protocols, most prominently the IEEE 802.11 Distributed Coordination Function. The interaction of this standard with those in other layers determines the efficiency and fairness between stations in the use of bandwidth resources.

The pioneering work by Bianchi [2] analyzed IEEE 802.11 performance through a careful modelling of collisions and the backoff response process stipulated by the standard. This led to accurate means of predicting the effective long term rates of a set of stations sharing the medium, for the case of a single modulation rate in the physical (PHY) layer. Recently in [3] this analysis was extended to consider multiple physical data rates present in a single cell. What has been mostly idealized in these references are higher layer protocols: it is assumed that stations always have packets to send. Some simple ways to account for the transport layer were suggested in [3, 4], but there is no full consideration of the TCP congestion control mechanisms that interact with the MAC layer to determine user performance.

Congestion control is commonly modelled through the Network Utility Maximization (NUM) approach of Kelly [5], which was extended in recent years to cross-layer optimization in wireless networks, (see [6, 7] and references therein). A significant portion of this literature refers to a *scheduled* MAC, for which the cross-layer optimization can be tackled through dual decomposition, although the scheduling component is difficult. In the case of a random MAC, most of the work has been theoretical, with an idealized model of collisions: [6] studies a change of variables to handle the non-convexity in these models, while a recent line of work [8, 9] shows how a fine-grained control of medium access probabilities allows in principle to approach the performance of the scheduled case. While this research has deep theoretical implications, it has limited impact in current practice since it does not represent prevailing protocols such as IEEE 802.11.

*Corresponding author

Email addresses: ferragut@ort.edu.uy (Andrés Ferragut), paganini@ort.edu.uy (Fernando Paganini)

We argue also that for wireless LAN technologies, the emphasis on collisions is misplaced. In recent versions of the IEEE 802.11 standard, the loss of performance due to collisions is not as important as one might expect, due to two main reasons: on one hand, most of the traffic is downlink, and the Access Point (AP) does not collide with itself; secondly, with the 802.11 default parameters, collision probabilities and resolution time are low when compared with data transmission, whenever the number of stations is not too large.

Far more relevant to user performance is the interaction between TCP and the multiple physical data rates. A first contribution of our paper is to model in the NUM framework this cross-layer interaction involving TCP flows that respond to losses in an access point buffer, itself served through multiple PHY rates. We show the inefficiency that can result from this arrangement, where users with high PHY rates are severely penalized, with minimal benefit for the slow users. This issue had already been identified in [3] without a full model of the TCP behavior. In our NUM model, the bias against fast users appears as an inverse scaling in the corresponding utilities; as a result one can make more general performance predictions, in particular study the effect of the bias on non-homogeneous TCP flows.

This model suggests that the desirable resource allocation should remove the utility bias; this alternative is studied in Section 3. We find that a suitable price scaling applied to the TCP sources achieves global convergence to the unbiased NUM problem, and explore ways to implement this scaling through an appropriate Active Queue Management scheme, termed Multirate RED. We also extend the analysis to more general networks than a single cell, involving possibly wired and wireless links. We show that as long as each link participates in a single "contention set", the problem admits a decentralized solution via price scaling.

Higher up in the protocol hierarchy, in Section 4 we study this resource allocation subject to randomly arriving connections. In wired networks it is known [10, 11] that the bandwidth sharing of TCP congestion control is able to stabilize any load compatible with the capacity constraints. We show the multi-rate counterpart of this condition, which involves a Discriminatory Processor Sharing (DPS) queue, and geometric models are developed to determine the stability region of a cell under spatially distributed loads. On the performance side, the DPS model allows for a study of connection level throughput (equivalently, workload completion time). This exhibits once more the inefficiency of the current allocation and the enhancement obtained through the unbiased version of Section 3.

Finally, in Section 5 we apply the above analysis to 802.11. We begin by analyzing the effective rates the TCP layer can achieve when operating above the 802.11 MAC layer, taking into account the impact of overheads. This provides us with a characterization of the effective data rates which were the basis of the preceding analysis. We then implement a multirate cell in the ns2 simulator, together with our Multirate RED algorithm. A series of tests are carried out to validate the conclusions of the NUM models, in particular the inefficiencies of current networks, as well as to exhibit the improvements that can be obtained through our proposals.

Conclusions are given in Section 6. Some preliminary versions of this work were presented in [12, 13].

2. TCP resource allocation in a multirate wireless environment

We begin by considering a single cell scenario where N users indexed by $i = 1, \ldots, N$ are downloading data from a single AP. Each station establishes a downlink TCP connection, with sending rate x_i. Our main purpose is to analyze the behavior of the x_i when the wireless medium offers different transmission rates to each user, and the higher layer control is performed by TCP.

The packets of each connection will be stored in the interface queue of the AP, before being put into the shared medium to reach their destination. The physical layer offers each destination a different modulation rate, which combined with other factors such as protocol overheads results in an *effective rate* C_i offered to the connection. We assume for now these rates are given, and postpone to Section 5.1 the discussion on how to calculate them in the case of 802.11 networks.

First, we analyze the service rate y_i attained by each user in such a queue. Assume that the Head of Line (HOL) probability for a packet of user i is proportional to the input rates x_i. Then, we can write the relationship

$$y_i = \frac{p_{HOL,i}L}{\sum_j p_{HOL,j}L/C_j} = \frac{x_i}{\sum_j x_j/C_j}. \qquad (1)$$

This relationship is similar to eq. (5) in [3], established through a renewal reward argument; here we omit the collision terms since we are focusing in the downlink case.

To complete the loop, we now model the TCP behavior, that determines the input rates x_i. Recall (c.f. [14]) that TCP congestion control algorithms can be modelled as

$$\dot{x}_i = k(x_i)(U_i'(x_i) - p_i),$$

where $U(x)$ is an increasing and strictly concave utility function, p_i is the link loss rate (interpreted as a price) and $k(x_i) > 0$ a scaling factor. In the analysis, we will restrict ourselves to the $\alpha-$fair family of utility functions introduced in [15], which verify $U'(x) = Kx^{-\alpha}$ with $\alpha > 0$ a parameter which determines the compromise between efficiency and fairness of the allocation.

We can model the link loss rate as:

$$p_i = \left(\frac{x_i - y_i}{x_i}\right)^+ = \left(1 - \frac{1}{\sum_j x_j/C_j}\right)^+ = p$$

which is simply the proportion of packets that exceed the current service rate, where $(\cdot)^+ = \max(\cdot, 0)$ as usual. With this model the packet loss rate of each flow is the same and the complete dynamics follow:

$$\dot{x}_i = k(x_i)(U_i'(x_i) - p), \tag{2a}$$

$$p = \left(1 - \frac{1}{\sum_j x_j/C_j}\right)^+. \tag{2b}$$

We would like to characterize the equilibrium of this dynamics in terms of a NUM problem. For this purpose, consider the following function:

$$\Phi(x) = \sum_i \frac{x_i}{C_i} - 1 - \log\left(\sum_i \frac{x_i}{C_i}\right),$$

whenever $\sum_i \frac{x_i}{C_i} > 1$ and 0 otherwise.

We have the following Lemma, proved in the Appendix:

Lemma 1. *Φ is a convex function of x.*

Consider now the following convex optimization:

Problem 1.

$$\max_x \sum_i \frac{1}{C_i} U_i(x_i) - \Phi(x). \tag{3}$$

Theorem 1. *The equilibrium of the dynamics* (2) *is the unique optimum of Problem 1. Moreover this equilibrium is globally asymptotically stable.*

Let $V(x)$ denote the objective function in equation (3). The following properties of $V(x)$ are proved in the Appendix:

Lemma 2. *V has compact upper level sets $\{x : V(x) \geqslant \gamma\}$, and is radially unbounded:* $\lim_{\|x\|\to\infty} V(x) = -\infty$.

Proof of Theorem 1. By Lemma 2 and the strict concavity of the objective function, there is a unique optimum for Problem 1, and it must satisfy the optimality conditions, namely:

$$\frac{1}{C_i} U_i'(x_i) - \frac{\partial}{\partial x_i}\Phi(x) = 0 \quad \forall i$$

By substituting Φ we have:

$$\frac{1}{C_i}\left[U_i'(x_i) - \left(1 - \frac{1}{\sum_j x_j/C_j}\right)^+\right] = 0$$

Identifying the last term as $p = p(x)$ in (2), the optimality conditions become:

$$U_i'(x_i) - p = 0 \quad \forall i$$

which is the equilibrium condition of (2).

We consider now $V(x)$ as a Lyapunov function of the system. Differentiating along trajectories:

$$\dot{V} = \nabla V \cdot \dot{x} = \sum_i \frac{k(x_i)}{C_i}\left(U_i'(x_i) - p\right)^2 \geqslant 0$$

So V is increasing along the trajectories. Moreover, $\dot{V} = 0$ only when $x = x^*$ the solution of Problem 1. Invoking Theorem 3.2 in [16], the equilibrium is globally asymptotically stable. □

The function Φ in Problem 1 plays the role of a penalty function, and thus Problem 1 can be seen as an approximation of the following problem:

Problem 2 (Modified Network Problem)**.**

$$\max_x \sum_i \frac{1}{C_i} U_i(x_i)$$

subject to the constraint:

$$\sum_i \frac{x_i}{C_i} \leqslant 1.$$

This NUM problem has two variations with respect to the standard one of [5]. The first is that the constraint is rewritten in terms of x_i/C_i, the "time proportion" the shared medium is used by connection i. The sum of the allocated time proportions must be less than one, this is a natural way to express capacity constraints in a shared multirate substrate.

The second difference with [5] is the scaling factor C_i^{-1} for the user utility. This is *not* a natural feature of the problem, it reflects a bias in the resource allocation against users of higher physical rates, which are given less weight in the net utility. The effect of this bias is a radical equalization of rates dominated by the slowest stations, as shown in the following example.

Example 1. *Assume 3 users are downloading data from a single AP, and they have equal utilities $U(x) = -\frac{1}{\tau^2 x}$ which models the TCP/Reno response, while their lower-layer effective rates are $C_i = 10$. In this case*

Problem 1 gives $x_i^* = 3.333$ *for all three users. Now, if for instance user 3 changes its radio conditions to* $C_3 = 1$*, the new allocation is*

$$x_1^* = x_2^* = x_3^* = 0.8333.$$

So we see that the faster destinations are heavily penalized due to the user 3 inefficiency[1]*. This problem has been observed in practice in 802.11 environments, and we will exhibit it by simulation in Section 5.*

Remark 1. *In the case where all users share a common utility function, the solution to Problem 2 is*

$$x_i^* = \frac{1}{\sum_j 1/C_j}; \tag{4}$$

the TCP rates are all equal to the harmonic mean of the effective data rates. This is in a accordance with results obtained in [3] for multiple-rate 802.11 networks, where collisions are considered, and the rate (4) appears as an upper bound on the realistic rate of permanent connections. In [3] the TCP layer is not modelled; rather, it is assumed that the AP has equal probability of servicing all users. Here we have modeled TCP, and we find that (4) holds only under the assumption of equal utilities (independent of the TCP flavor). However, if users have different utility functions, as is the case when TCP-Reno connections have different RTTs, this solution is no longer valid, and the allocation must be calculated as the solution of Problem 2.

The previous remarks and examples suggest that ways of removing the bias from Problem 2 should be explored. This is the subject of the following Section.

3. A more efficient resource allocation

We will study unbiased versions of Problem 2, starting with the case of a single wireless cell.

3.1. The case of a single cell

Consider the following optimization problem:

Problem 3 (Wireless Multirate Network Problem)**.**

$$\max_x \sum_i U_i(x_i)$$

subject to the constraint:

$$\sum_i \frac{x_i}{C_i} \leqslant 1, \tag{5}$$

As formulated, the above NUM problem becomes a special case of those in the literature on *scheduled* wireless networks (see e.g. [7] and references therein), where the set of feasible rates is taken to be the convex hull of rates achievable by independent (non-interfering) sets of links. In the current scenario, only individual links are schedulable without interference, at rate C_i, so the convex hull becomes (5). In contrast to these references, we will seek a solution to Problem 3 compatible that does not require a complicated scheduling mechanism in the AP, which would imply a significant departure from current 802.11 networks.

The Lagrangian for Problem 3 is

$$\mathcal{L}(x, p) = \sum_i U_i(x_i) - p\left(\sum_i \frac{x_i}{C_i} - 1\right).$$

To see the difference with Problem 2, consider the case of identical utilities $U_i(\cdot) = U(\cdot)$ of the α-fair family. The Karush-Kuhn-Tucker (KKT) [17] conditions for this problem give in particular

$$x_i^{-\alpha} = U'(x_i) = \frac{p}{C_i};$$

imposing equality in (5) we obtain

$$p^{-\frac{1}{\alpha}} \sum_i C_i^{\frac{1}{\alpha}-1} = 1,$$

from where we obtain the optimal rates

$$x_i^* = \frac{C_i^{\frac{1}{\alpha}}}{\sum_j C_j^{\frac{1}{\alpha}-1}}. \tag{6}$$

We see that, in contrast to (4), rates are no longer equalized in the solution, users with larger C_i will receive a larger share of resources[2]. We look at some important special cases.

Proposition 1. *In the case of proportional fairness,* $U(x) = K\log(x)$ *for all connections (*$\alpha = 1$*) the equilibrium of Problem 3 is* $x_i^* = \frac{C_i}{n}$*. In particular, the allocated rate for user i depends only on its own effective rate and the total number of users.*

Equivalently: under proportional fairness time is shared equally among users, but those with a more efficient use of time can obtain a proportionally greater rate. This protects the fastest users from the lower rate

[1] The result from Problem 1 would be $x_1^* = x_2^* = x_3^* = 0.89$, which shows that the barrier function approximation is very close.

[2] Except for the case $\alpha \to \infty$ (max-min fair allocation) where we recover (4).

ones. For instance, in Example 1, high rate users would be unaffected by the change in C_3.

When TCP Reno is in use, we do not have complete protection but the situation is nevertheless improved, as the following example shows.

Example 2. *Consider the same situation of Example 1. When all three users have $C_i = 10$, the equilibirium of Problem 3 is the same as before, $x_i^* = 3.33$. When user 3 changes its radio conditions and $C_i = 1$, the equilibrium changes to $x_1^* = x_2^* = 1.93$, $x_3^* = 0.61$. The total network throughput increases by $\approx 80\%$ with respect to Example 1, and fastest users are not as heavily penalized.*

We would like to drive the network to the equilibrium of Problem 3. A simple primal-dual gradient algorithm to solve this optimization problem is:

$$\dot{x}_i = k(x_i)\left(U'(x_i) - \frac{p}{C_i}\right), \tag{7a}$$

$$\dot{p} = \left(\sum_i \frac{x_i}{C_i} - 1\right)_p^+, \tag{7b}$$

where $k(x_i) > 0$ as before and $(\cdot)_p^+$ is the usual positive projection. It is well known [18, 19] that for strictly concave utility the trajectories of (7) converge globally to the optimum of Problem 3.

Note that, in order to implement (7), the congestion control must react to a scaled version of the congestion price generated at the links. To achieve this in a decentralized way, we must convey to the sources this scaled price. In the following section we analyze a practical algorithm to achieve this goal.

3.2. The Multirate RED algorithm

In the wired case, dual algorithms have interpreted the price variable as the queueing delay [20, 21]. This is also the case here in this modified version. By integrating $\dot{p}$ in equation (7b) we see that p tracks the amount of time the shared medium is not capable of coping with the demands, and thus accumulating as delay in the queue.

More formally, let b_i denote the amount of data of connection i in the buffer (assume it is non empty). Then

$$\dot{b}_i = x_i - y_i$$

and the delay d is given by:

$$d = \sum_i \frac{b_i}{C_i}.$$

Therefore, recalling equation (1) we have:

$$\dot{d} = \sum_i \frac{\dot{b}_i}{C_i} = \sum_i \frac{x_i}{C_i} - 1.$$

Observe further that when all capacities are equal $C_i = C$ we recover the delay based model of [20, 21].

From equations (7) we see that in order to appropriately solve Problem 3, we need to scale the price to which the user reacts by the effective rate C_i. This makes sense since connections with higher rates use the medium more efficiently, and thus should be charged less whenever this resource is scarce. Note however that this poses problems on implementation, because it prevents from using directly the queueing delay as the price. Moreover, the user source would have to be notified of the correct MAC level rate, which is infeasible. We now discuss a practical method to overcome these limitations without resorting to a complicated scheduling mechanism. In order to drive the system to the optimum of Problem 3, we propose to use a simple Active Queue Management policy which we call the Multirate RED algorithm (MRED).

Instead of using queueing delay as the price, we propose to use as a proxy the buffer length b, and to generate the price, the AP discards packets randomly with probability p_i proportional to $\frac{b}{C_i}$ for connection i. This gives a linear Random Early Detection (RED) algorithm, but with probabilities related to the effective data rates. Note that less packets will be dropped for connections with higher MAC rates. Moreover, this mechanism can be implemented in the AP resorting only to local information, such as destination address and current effective rate for this destination.

The closed loop dynamics for the proposed system is:

$$\dot{x}_i = k(x_i)\left(U'(x_i) - \kappa b/C_i\right), \tag{8a}$$

$$\dot{b} = \left(\sum_i x_i - y_i\right)_b^+ = \left(\sum_i y_i\right)\left(\sum_i \frac{x_i}{C_i} - 1\right)_b^+ \tag{8b}$$

where $\kappa > 0$ is the proportionality constant of RED. These equations are similar to (7). In particular, in equilibrium, the x_i and $p = \kappa b$ will satisfy the KKT conditions of Problem 3. Stability results for these equations are harder to obtain, in Section 5 we explore its behavior by simulation.

3.3. Extension to wireless access networks

The analysis of Section 3.1 is valid in a single cell setting. We would like to generalize it to more complicated networks. In particular, we are interested in the case of

wireless access systems, i.e. a wired backbone which has non-interfering wireless cells as stub networks.

In such networks, one obtains a combination of classical capacity constraints for wired links (that do not interfere with any other links), and constraints of the type (5) for links of the same stub cell interfering with one another, potentially with different effective rates. For such networks, we would like to develop a price scaling method that enables decentralized users to allocate resources to maximize utility.

Of course, one could further consider more general interference models, such as different wireless cells interfering with each other, or more arbitrary interference patterns as has been considered in the scheduling literature [7]. These, in addition to the complexity of scheduling, lead to optimization problems that are difficult to decentralize. For this reason we choose to focus on a narrower setting which nevertheless covers scenarios of practical importance, and which can be addressed through a smaller departure from current practice, in particular using currently deployed MAC layers.

Consider then a network composed of links $l = 1, \ldots, L$. These links can be wired or wireless, and have an effective transmission rate C_l. In the case of wired links, C_l is the link capacity. In the wireless case, it is the effecive data rate. Let $i = 1, \ldots, n$ represent the connections, with rate x_i, and R the classical routing matrix: $R_{li} = 1$ if connection i traverses link l and 0 otherwise.

To represent the contention inherent to the case, we group the links l in contention sets: two links belong to the same contention set if they cannot be transmitting simultaneously and define $G_{kl} = 1$ if link l belongs to contention set k and 0 otherwise; we call G the contention matrix. If a link is wired, its contention set is a singleton. In the case of a wireless cell, the contention set is composed of all links that depend on the same AP. Each link belongs to only one contention set: this is the restriction imposed on the interference model.

The network capacity constraints can then be written as $Hx \leqslant \mathbf{1}$ where H is given by $H = GC^{-1}R$, with G, R previously defined, $C = diag(C_l)$ and $\mathbf{1}$ a column vector of ones.

To see how this framework enables us to model different situations, consider the following examples:

Example 3 (Wired network)**.** *If all links are wired, the contention matrix G is the Identity matrix. By taking R and C as before we recover the classical wired constraints* $\sum_{i \in l} x_i \leqslant C_l$.

Example 4 (Single wireless cell)**.** *If there is only one wireless AP with N users in the cell, we can take R as the identity matrix, C as the wireless effective capacities and* $G = \mathbf{1}^T$ *(there is only one contention region where all links participate). We then recover the constraints discussed in Section 3.*

Example 5 (Wireless Distribution System)**.** *To see a more complete example, consider the network composed of wired and wireless links shown in Figure 1. This topology appears in outdoor wireless distribution scenarios. We can model the capacity constraints of this network with the above framework by taking:*

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

$$C = diag(c, c_{AP_1}, c_{AP_2}, c_1, c_2, c_3, c_4)$$

$$R = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

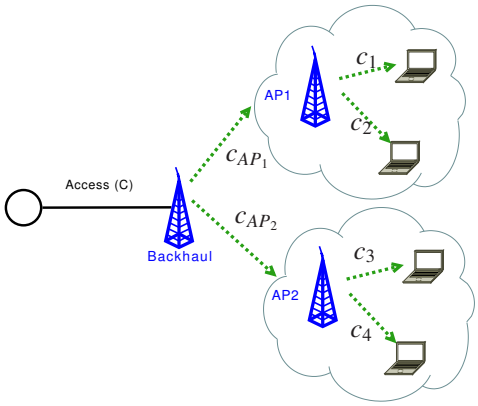

Figure 1: Topology of a mixed wired-wireless distribution system with 4 end-users.

We can now pose the general Network Utility Maximization problem, which is:

Problem 4 (Wired-Wireless Network Problem)**.**

$$\max_x \sum_i U_i(x_i)$$

subject to:

$$Hx \leqslant \mathbf{1}.$$

The previous problem seeks an optimal allocation within the natural constraints of the network, expressed in terms of allocated time proportions. These constraints are equivalent to the ones used in the scheduling literature [7]. We will show that for the special structure under consideration, a decentralized solution can be obtained that involves FIFO buffers and standard congestion control, provided a suitable price scaling is applied.

Consider the Lagrangian of Problem 4:

$$L(x,p) = \sum_i U_i(x_i) - p^T(Hx - \mathbf{1}) \tag{9}$$

where $p = (p_1, \ldots, p_K)^T$ is the vector of prices. We see therefore that we have one price for each contention set.

By denoting $q = H^T p$, the KKT conditions of Problem 4 are $U_i'(x_i) = q_i$ where q_i is given by:

$$q_i = \sum_{l:i\in l} \sum_{k:l\in k} \frac{p_k}{C_l} \tag{10}$$

Therefore, the connection must react to a price which is the sum of the prices of the the contention sets it traverses, divided by the link capacities it uses within each contention set.

Again, to solve Problem 4, we can use a primal-dual algorithm with the following dynamics:

$$\dot{x}_i = k(x_i)\left(U'(x_i) - q_i\right) \tag{11a}$$
$$\dot{p} = (Hx - \mathbf{1})_p^+, \tag{11b}$$
$$q = H^T p. \tag{11c}$$

These dynamics are globally asymptotically stable [18, 19] and its equilibrium is the solution of Problem 4. In this context, the prices track again the queueing delays at each FIFO queue.

The remaining issue is whether this prices can be correctly generated and transmitted to the sources via the MRED implementation discussed in Section 3.2. The following examples show how this can be done for wireless access networks.

Example 6 (Mixed wired-wireless access network)**.** *A typical configuration for wireless access coverage is to distribute access points in non overlapping channels across the region to cover, and wire them to the Internet access node. This produces the tree topology of Figure 2. There, the APs are connected to a central switch, which is also connected to the router handling the Internet connection. End users are then connected to the APs via 802.11 for example. In this case, each user traverses three contention sets, one per link.*

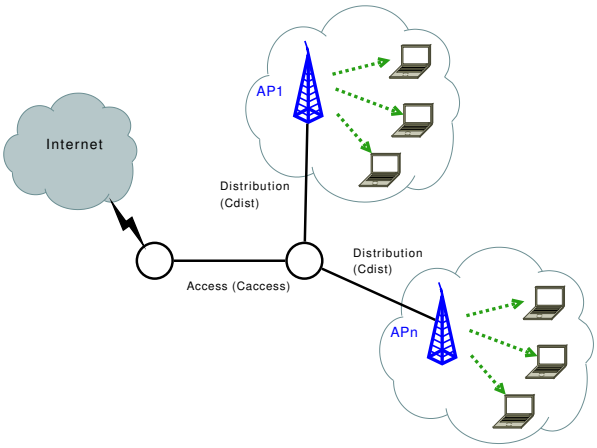

Figure 2: Topology of a mixed wired-wireless access network.

Assuming the link capacities and user distributions shown in Figure 2, the corresponding price for user i is calculated according to equation 10 as:

$$q_i = \frac{p_{access}}{C_{access}} + \frac{p_{dist_j}}{C_{dist_j}} + \frac{p_{AP_j}}{C_i}$$

where p_{access}, and p_{dist_j} are the queueing delays of the wired links traversed by packet of user i, whereas and p_{AP_j} is the queueing delay of the FIFO queue at the AP used by connection i. All are scaled by the corresponding link capacities.

By using the Multirate RED algorithm in each link in the network, we can therefore transmit this price to the source, and impose the notion of fairness of Problem 4 by emulating the dynamics of (11)*.*

Example 7 (Wireless distribution system)**.** *A variation of the above example occurs when the area to cover is large, for instance, large outdoor deployements. In this case, the distribution links that connect each AP with the wired network are replaced by a wireless cell that backhauls all the APs and which is directly connected to the Internet router, as in Figure 1. The APs have 2 radio interfaces: one to connect to the backhaul link and one for local connectivity.*

In this case, the corresponding price for user i can again be calculated according to 10 as:

$$q_i = \frac{p_{access}}{C_{access}} + \frac{p_{BH}}{C_{AP_j}} + \frac{p_{AP}}{C_i}.$$

The main difference with respect to the wired backhaul case is that p_{BH} is common to all distribution links, it reflects the FIFO queueing delay at the backhaul node. c_{AP_i} is the effective rate at which the AP of user i connect

to the backhaul node. Again, by using Multirate RED in this tree topology we can impose the notion of fairness of Problem 4.

In Section 5 we will analyze the performance of such systems via simulation.

What is the structure in these examples that enables a simple distributed solution? The essence is that each link belongs to a single contention set, and each such set is served by a common FIFO queue. Under these assumptions, the inner sum of (10) consists of a single term, i.e. we can write

$$q_i = \sum_{l:i\in l} \frac{1}{C_l} p_{k(l)},$$

where $k(l)$ is the contention set associated with l. The price $p_{k(l)}$ is the queueing delay of the corresponding FIFO queue. By performing the correct capacity scaling, the price q_i can be appropriately relayed to the sources.

4. Connection level stability and performance.

In this section we turn to the analysis of multirate wireless networks at the connection level, focusing on the case of a single cell. We develop a stochastic model for the evolution of the number of connections present in the cell, that tries to capture the time and spatial behavior of a connection arrivals, as well as the resource allocation that the lower layers (TCP, multirate MAC) impose on the rate of these connections.

Assume that the AP is located at the origin of the plane $\mathbb{R}^2$. Let $R_j \subset \mathbb{R}^2$ be the region of the plane where users can achieve a transmission rate C_j. As an example, in an outdoor setting the R_j could be concentric discs around the origin, with decreasing physical layer rates as depicted in Figure 3. Our model, however, requires no assumptions on the shape of the different rate regions.

We describe first the arrival of connections. Stations can be anywhere in the cell and connections may arrive randomly in time. We model this by assuming that connections follow a spatial birth and death process [22]. New connections appear in the cell as a Poisson process of intensity $\lambda(x), x \in \mathbb{R}^2$ which represents the frequency per unit of area. In particular, users arrive to region R_j with intensity:

$$\Lambda_j = \int_{R_j} \lambda(x)\, dx.$$

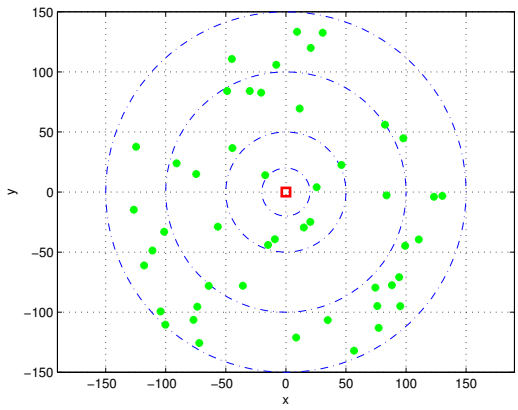

Figure 3: A typical 802.11 cell with geometrically random connection demands and circular rate regions.

Each connection demands some random amount of workload which we assume exponential[3] with mean $1/\mu$. For simplicity we assume that these workloads have the same mean in all regions.

To complete the model, and thus derive stability conditions and performance metrics at the connection level timescale, we must specify the service rate of each connection. In this regard, we will analyze both the biased resource allocation discussed in Section 2, and the alternative proposed in Section 3.

4.1. Connection level performance of current TCP-multirate environments.

Supppose that rates are allocated to ongoing connections following the TCP resource allocation of Problem 2; as argued in Section 2, this approximately models the behavior of the prevailing loss-based TCP congestion control when connections share a FIFO queue, serviced with multiple lower-layer rates.

For simplicity, assume that all connections share the same utility function, for which the solution of Problem 2 is given by equation (4). In this case, given that at a certain moment of time there are n_j connections of rate C_j, the rate for every connection is given by

$$x(\mathbf{n}) = \frac{1}{\sum_j n_j/C_j}; \tag{12}$$

here we denote by $\mathbf{n}$ the vector of n_j.

[3]The use of exponential distributions for connection size is questionable, we do this in order to obtain a tractable model. See [11] for a discussion.

Putting together all the previous considerations we get the following Markov model for the vector $\mathbf{n}(t)$.

$$\mathbf{n} \mapsto \mathbf{n} + e_j \quad \text{with intensity} \quad \Lambda_j \tag{13a}$$

$$\mathbf{n} \mapsto \mathbf{n} - e_j \quad \text{with intensity} \quad \mu n_j x(\mathbf{n}) \tag{13b}$$

where e_j denotes the vector with a 1 in the $j-$coordinate and 0 elsewhere.

Remark 2. *The Markov model holds irrespectively of whether new connections are from different stations or belong to the same station (in the same position in space). The only important parameter is the total arrival rate Λ_j of connections from class j.*

The first question of interest for this model is the stochastic stability region, i.e. the region of arrival intensities that produce an ergodic Markov process, and hence a stationary distribution. We are also interested in calculating the throughput of connections. Both issues will be studied by identifying the above model with a well known queue.

Substituting equation (12) in (13) we can rewrite the death rates as

$$\mu n_j x(\mathbf{n}) = \mu C_j \frac{g_j n_j}{\sum_k g_k n_k},$$

where $g_j := \frac{1}{C_j}$. With this notation, we can identify the transition rates of (13) with those of a *Discriminatory Processor Sharing* (DPS) queue [23], with total capacity 1, and where for each class j the arrival rate is Λ_j, the mean job size $\nu_j = \mu C_j$ and the DPS weight is g_j.

With this identification, the following result follows directly from the stability condition of the DPS queue:

Proposition 2. *The Markov process describing the number of connections is ergodic if and only if:*

$$\varrho = \sum_j \frac{\Lambda_j}{\mu C_j} < 1. \tag{14}$$

It is worth specializing equation (14) to the important case in which connections arrive uniformly distributed in the cell, i.e. $\lambda(x) = \lambda$, a constant. This can represent a situation where users do not know where the AP is. In that case, if A_j is the area of the region R_j, $\Lambda_j = \lambda A_j$ and the stability condition becomes

$$\frac{\lambda}{\mu} < \frac{1}{\sum_j \frac{A_j}{C_j}}. \tag{15}$$

This is of the form $\rho < C^*$ where ρ is the traffic intensity in $bits/(s \cdot m^2)$ and C^* can be thought as a cell traffic capacity, which captures the geometry, and is a weighted harmonic mean of the effective rates.

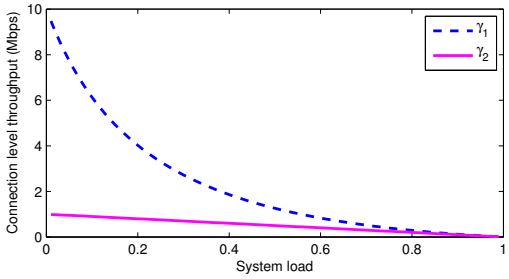

Figure 4: Connection level throughputs, 2 classes, $C_1 = 10$, $C_2 = 1$ Mbps. Arrival rates are proportional to coverage areas.

The second issue we are interested in is performance, measured by connection level throughput, whenever the system is stable (i.e. $\varrho < 1$). This can be evaluated by calculating the expected time in the system for jobs in a DPS queue, using the results of [24, 25]. Let τ_j be the time in the system for a job of class j. For the case of two classes there is an explicit formula:

$$E(\tau_1) = \frac{1}{\mu C_1(1-\varrho)}\left(1 + \frac{\Lambda_2(C_1 - C_2)}{\mu C_2^2(2-\varrho)}\right)$$

$$E(\tau_2) = \frac{1}{\mu C_2(1-\varrho)}\left(1 + \frac{\Lambda_1(C_2 - C_1)}{\mu C_1^2(2-\varrho)}\right)$$

where ϱ is the system load as in equation (14).

Observing that a user can send an average $1/\mu$ bits during time τ_1 we can measure the connection level throughput as:

$$\gamma_j = \frac{1}{\mu E(\tau_j)}$$

Applying the previous formula, we calculated the connection level throughput for a cell with two allowed rates as in Example 1, namely $C_1 = 10$ and $C_2 = 1$ Mbps. We assume that connections arrive uniformly in space, so the proportion of slow connections is greater. Results are shown in Figure 4, that shows connection level throughput under varying cell load. Note that when the load increases, the connection level throughput of both classes equalizes to the detriment of faster users, with no appreciable gain for the slower ones; this is a consequence of the allocation that equalizes per-connection rates.

For the general case with multiple classes simple explicit formulas for the τ_j are not available. Rather, these values can be obtained by solving a system of linear equations:

$$E(\tau_j) = B_{j0} + \sum_{k=1}^{m} E(\tau_k)\Lambda_k B_{jk} \tag{16}$$

where the B_{jk} depend on the system parameters (we refer the reader to [23] for the expressions). Therefore, we can calculate the connection level throughputs γ_j by numerically solving this system for the effective data rates of, for example, IEEE 802.11. A case study is given in Section 5.

Results similar to the above example were given in [26], where a connection level model is also analyzed. Once again, however, in this work the lower layers are not modeled, and a simple time-sharing mechanism is assumed for the medium. Here we have found that the "downward" equalization of connection-level throughputs will occur with any $\alpha-$fair congestion control, in particular current implementations, provided utilities are the same for all sources.

4.2. *Connection level performance of the unbiased allocation*

We now analyze the connection level performance of the resource allocation of Problem 3, which removes the bias in utility against fast users, and as discussed in Section 3 can be implemented through a price scaling mechanism.

Again, we will assume that all users share the same utility function $U(x) = x^{1-\alpha}/(1-\alpha)$ of the $\alpha-$fair family. In this case the solution of Problem 3 is given by (6); given that at a certain moment of time there are n_j connections of rate C_j, we obtain the connection rates

$$x_i(\mathbf{n}) = \frac{C_i^{1/\alpha}}{\sum_j n_j C_j^{1/\alpha-1}}. \qquad (17)$$

The connection level process behaves again as a DPS queue, with arrival rates Λ_j, job sizes μC_j and weights depending on α as:

$$g_j = C_j^{1/\alpha-1}.$$

Therefore, for any α, the system will be stable when the loads verify the stability condition (14), and the expected job service times and connection level throughput can be calculated by the same method we described above.

It is worth noting that as $\alpha \to \infty$, we recover the weights of the current allocation analyzed in the previous section, which can be highly inefficient, penalizing the highest rates.

For the case of $\alpha = 1$ corresponding to proportional fairness, the weights associated to each class become equal and the performance metrics can be explicitly calculated as:

$$E(\tau_j) = \frac{1}{\mu C_j(1-\varrho)}, \quad \gamma_j = C_j(1-\varrho).$$

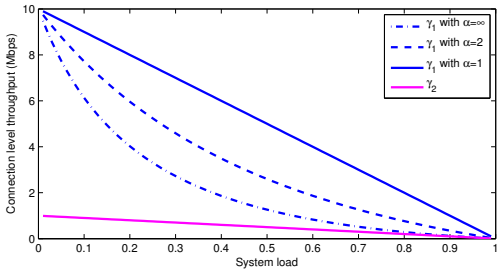

Figure 5: Connection level throughputs, 2 classes, $C_1 = 10$, $C_2 = 1$ Mbps. for different fairness notions under price scaling. Class 2 throughput is only plotted once since results are similar in all three cases.

This has the nice property that connection level throughputs become proportional to the lower layer offered rate, the proportionality constant being the *slowdown* of the processor sharing queue, which only depends on the cell *total* load.

The case of current TCP Reno-like algorithms will be an intermediate one, corresponding to $\alpha = 2$. In Figure 5 we compare the connection level throughputs of a cell with two allowed rates as in Example 1, namely $C_1 = 10$ and $C_2 = 1$ Mbps, in the case of max-min, proportional fair and Reno-like allocations.

5. Application to IEEE 802.11

In this Section, we apply the previous models to quantify the behavior of TCP over an IEEE 802.11 MAC layer. We begin by calculating the effective data rates 802.11 provides to TCP connections, and then we proceed to analyze through simulation several examples.

5.1. *Calculating the effective rates: the impact of overheads.*

In the models derived in the preceding sections, an important quantity is the C_i, which is the effective data rate at which TCP packets from a given user i are served by the underlying layers. When the AP wants to trasmit a packet to user i of length L at a physical layer modulation rate PHY_i using 802.11, it must comply with a series of backoff and waiting times, as well as headers included by the PHY layer.

This means that the MAC layer offers a service to the upper layer consisting of a tranmission rate $C_i \leqslant PHY_i$. This has been analyzed before [2, 3, 12] and we will recall and extend this analysis here.

The time it takes to send this packet has a fixed component given by

$$T_i^0 := DIFS + H + \frac{L}{PHY_i} + SIFS + MAC\_ACK_i, \quad (18)$$

that includes the time in the air and all overheads, plus a random number of time slots $K\sigma$, where $K \sim U\{0, \ldots, CW\}$. In Table 1 we show typical values of these parameters for 802.11g.

Parameter	Value
Slot time σ	$9\mu s$
$SIFS$	$10\mu s$
$DIFS$	$28\mu s$
PLCP Header H	$28\mu s$
PHY_i	6Mbps . . . 54Mbps
CW_{min}	15 slots
MAC_ACK	$50\mu s$

Table 1: IEEE 802.11g parameters

We are interested in the average rate obtained by a station to study the upper layer effective rate. Observing that each packet is treated independently, the transmission times of successive packets form a renewal process, and the renewal reward theorem [27] tells us that in the long range the average rate is:

$$C_i^0 = \frac{L}{EK\sigma + T_i^0} = \frac{L}{\frac{CW_{min}}{2}\sigma + T_i^0}, \quad (19)$$

where we substituted K for its mean. We also took $CW = CW_{min}$ since we are modeling downlink traffic from the AP, which does not collide with itself. We also assume the appropriate PHY_i has been used so that one can neglect packet transmission errors. The denominator of the preceding expression (mean total time) is denoted by T_i.

In Table 2 we show the corresponding MAC level rates C_i^0 for the different PHY rates allowed in 802.11g with parameters as in Table 1. Note the impact of overheads in the highest modulation rates: this is due mainly to the fact that physical and MAC layer overheads are fixed in time, independent of the modulation rate PHY_i chosen for the data. This implies that higher modulation rates can finish the data part of the packet more quickly, but they still have to send the fixed length headers and wait for the backoff slots.

When TCP connections are taken into account, another overhead must be considered: the TCP ACK packet. These packets were designed to have low impact on the reverse path, by having a length of 40 bytes. However, due to the overheads added by the MAC layer, the TCP ACK becomes non negligible, in particular at high modulation speeds. We assume that one TCP ACK is sent in the uplink direction for every TCP packet sent downlink. We will also assume that collision probabilities are low between downlink packets and the TCP ACKs. Under these assumptions, the TCP ACK packet introduces another overhead time in the system. The effective data rate then becomes:

$$C_i = \frac{L}{T_i + TCP\_ACK_i} \quad (20)$$

where TCP_ACK_i is the average time to transmit a TCP_ACK packet and is given by:

$$TCP\_ACK_i := DIFS + \frac{CW_{min}}{2}\sigma + H + \frac{L_{ack}}{PHY_i} + SIFS + MAC\_ACK_i \quad (21)$$

where L_{ack} is typically 40 bytes. These effective data rates C_i are also shown in Table 2. Again, note the strong impact of the TCP ACKs in the performance of the protocol at high modulation rates, due to the fact that the lower layer protocol overheads are fixed in time.

PHY rates	MAC rate (C_i^0)	Eff. rate (C_i)	Measured rate
54	28.6	19.5	19.7
48	26.8	18.6	18.6
36	22.4	16.3	16.2
24	16.9	13.1	12.8
18	13.6	11.0	10.6
12	9.78	8.25	8.0
6	5.30	4.74	4.6

Table 2: MAC rates for the corresponding PHY rates of 802.11g for a packet size of $L = 1500$ bytes. Values are in Mbps. The measured rates are estimated within $0.1 Mbps$ of error.

To validate the above expressions, we simulated in ns-2 [28] several independent replications of a long TCP connection in a single wireless hop scenario. The average throughput for each PHY rate is reported in the last column of Table 2, showing good fit with the predicted values.

In the following, we shall not try to modify the impact of overheads and consider them given, since they are included in the standards. For the purpose of modelling, we will use the C_i values of Table 2, as the effective data rates provided by 802.11 to TCP connections, and use this values for the algorithm implementations.

5.2. Multirate RED implementation

As we discussed in section 3, the price to which a TCP connection should react in order to attain the equilibrium of Problem 3 is the queueing delay. However, this price should be scaled by the effective data rate C_i the connection experiments in each link. Clearly, this is difficult to implement without resorting to scheduling. Moreover, typical TCP connections use loss based congestion control mechanims, such as TCP Reno. Therefore, we propose to use the MRED algorithm developed in Section 3.2 at each node to attain the optimum of Problem 3.

To test the proposal in a real environment, we implemented this algorithm in the Network Simulator `ns-2` [28]. Our implementation is based on the library `dei80211mr` [29]. Two important extensions were made to the library: the existing ARF mechanism was updated to cope with the possibility of a single node having different modulation rates for different destinations, which reflects the real behavior of current APs. The second modification was to implement the Multirate RED (MRED) queue, where the described early packet discard takes place.

Note that the cross-layer information needed for implementation of the mechanism is minimal: whenever a packet for next-hop j is received, it is discarded with probability $p_j = \kappa b/C_j$ where κ acts as a scaling parameter, b is the current queue length, and C_j is the corresponding effective rate for the current modulation rate the AP maintains with destination i (as in Table 2). In the case of wired links, the link capacity is used to scale this drop probability. The non-dropped packets are served then on a FIFO basis.

We now present several simulation scenarios to illustrate the behavior of the proposed algorithm.

5.3. Simulation examples

5.3.1. Single-cell scenario

We simulate the topology shown in Figure 6, which consists of a single cell 802.11g scenario in which 3 users are connected with a modulation rate $PHY_i = 54Mbps$, and some time later, a fourth user is added at the lowest possible modulation $PHY_4 = 6Mbps$. All four connections use TCP/Newreno and share equal Round Trip Times (RTTs), then having similar utility functions.

Results are shown in Figure 7. We simulated 50 independent replications of the experiment, and the average throughput for each connection type as well as 95% confidence intervals are shown. For these modulation rates, the effective data rates according to Table 2 are $C_i = 19.5Mbps$, $i = 1, 2, 3$ and $C_4 = 4.74Mbps$. In the first graph of Figure 7, we see that all connections converge to the same throughput, which is approximately $x^* = 2.74Mbps$, the harmonic mean discussed in Remark 1. In the second graph, we show the behavior of the system under the MRED algorithm. In this case the allocation converges approximately to $x_i^* = 4.2Mbps$, $i = 1, 2, 3$ and $x_4^* = 2.1Mbps$, which is the exact solution of Problem 3. Note that the total throughput in the network is increased by more than 30%.

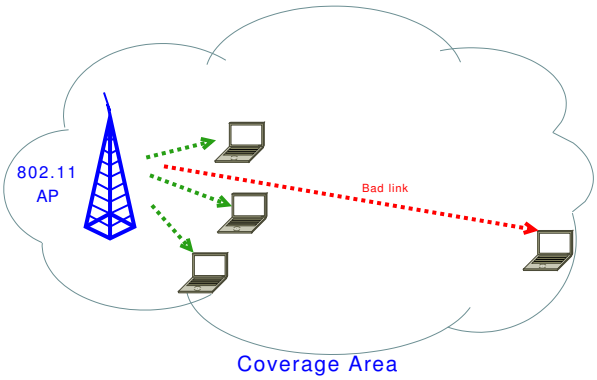

Figure 6: Topology of a single-cell scenario.

5.3.2. Different RTTs scenario

The purpose of this example is to show that Problem 2 captures the behavior of the system when the TCP connections have different RTTs, and thus different utilities, and to show how efficiency can also be improved in this case with the MRED algorithm.

We consider the topology of Figure 8, where two connections with different RTTs share a wireless bottleneck link. In this example, connection 1 has a longer RTT than that of connection 2, and its station is closer to the AP, having a modulation rate $PHY_1 = 54Mbps$. The second connection has a modulation rate of $PHY_2 = 6Mbps$. Both connections use TCP/Newreno, which we model by the utility function $U(x) = -1/(\tau^2 x)$ with τ the connection RTT.

Plugging these values into Problem 2 using the effective data rates of Table 2, the allocation results $x_1^* = 2.43Mbps$ and $x_2^* = 4.14Mbps$. In the first graph of Figure 9 we show the results of 50 independent replications of the experiment, which shows that indeed the connection throughputs converge approximately to the values predicted by Problem 2.

By using MRED in the AP we can change the allo-

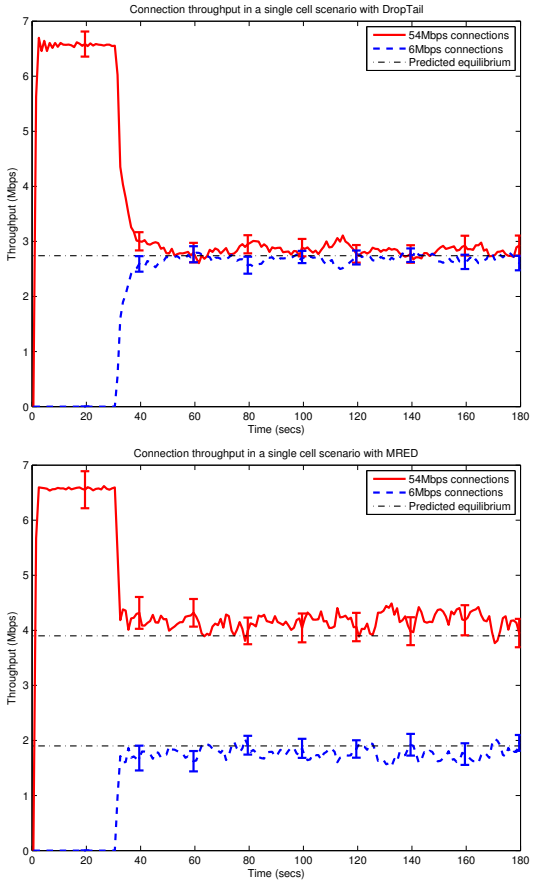

Figure 7: Comparison between throughputs: without MRED (above), with MRED (below).

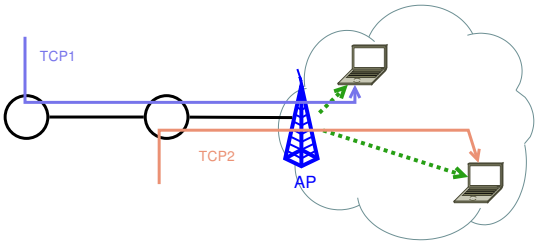

Figure 8: Wired-wireless topology.

cation to the one proposed in Problem 3, removing the bias of Problem 2. The resulting allocation is $x_1^* = 4.38$ and $x_2^* = 3.67$.In the second graph of 9 we show the corresponding simulation results. We see that the MRED algorithm approximately drives the system to the new equilibrium. Note that this new equilibrium is also more efficient.

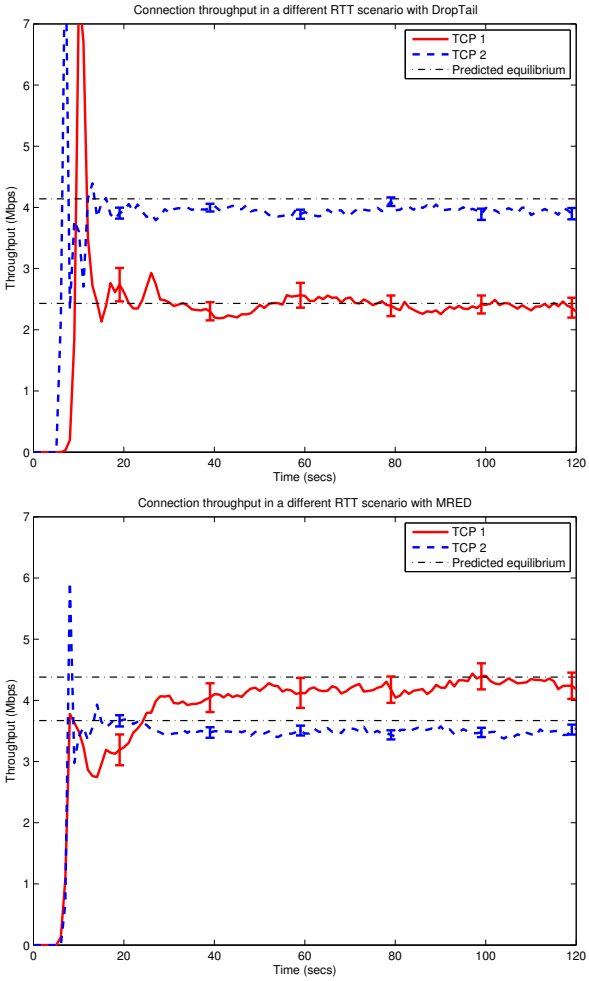

Figure 9: Wired-wireless topology simulation. Above: original allocation. Below: MRED algorithm.

5.3.3. IEEE 802.11 connection level throughputs

We now apply the results of Section 4 to evaluate the TCP connection level performance when working with a random workload and an underlying IEEE 802.11 MAC layer. As before, we focus on a downlink scenario where all TCP connections arrive at the coverage zone at a random point, with total arrival rate Λ_j for the rate C_j. These rates are chosen as in Table 2, which are valid for 802.11g.

As a first example, we consider a single cell scenario with two *PHY* rates. Users near the cell establish connecions at the highest possible modulation rate of $54Mbps$ and the remaining users use the lowest possible modulation of $6Mbps$. We simulated the random arrival of connections in ns-2 and measured the connection level throughput for different values of the cell load.

To take into account the fact that the low modulation rate has a greater coverage area, the arrival intensities were chosen proportional to the size of the coverage areas.

In Figure 10 we plot the connection level throughputs obtained by simulation and the predicted throughputs using the results of Section 4 for different values of the total cell load ϱ. The first graph shows the connection level throughputs when the proposed Multirate RED is not in use, and the second one shows the results for a cell using Multirate RED. Note that in both cases results show a good fit with the model.

In each case, the connection level throughputs start at the effective data rate for each class, C_i, corresponding to the case where each connection arrives to an empty network, and thus is able to obtain its full rate. When the offered load ϱ begins to grow the throughputs go to zero, as expected. We observe that in the case where MRED is not in use, the high modulation rate users are more heavily penalized, and throughputs tend to equalize.

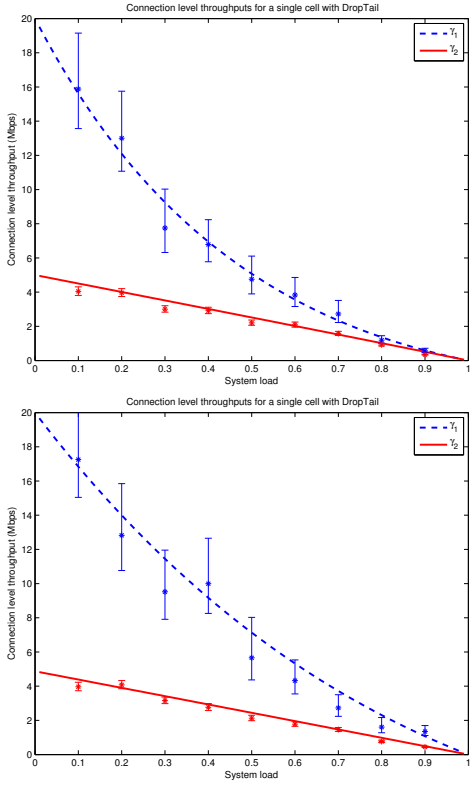

Figure 10: Connection level throughputs for an IEEE 802.11g cell with two modulation rates. Above: without MRED, Below: with MRED in use.

When all the data rates are allowed, the connection level throughputs can be calculated by solving the linear system of equations discussed in Section 4. In Figure 11 we plot the predicted connection level throughputs for such a setting, with increasing cell load. Again, the arrival rates for each class are chosen proportional to the estimated coverage areas. In the first graph, we show the results for current 802.11g cells, where price scaling is not used, and thus penalizing the higher rates. In the second graph we show the connection level throughputs under the price scaling mechanism, removing the bias. We can see that the higher rates get better throughput in all cell loads, while the lower ones are mostly unaffected by the change.

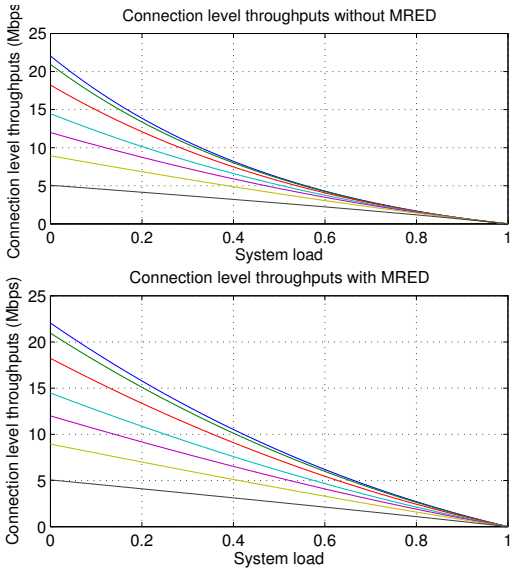

Figure 11: Connection level throughputs for an IEEE 802.11g cell. Above: without MRED, Below: with MRED. Each line corresponds to a different *PHY* rate in decreasing order.

To evaluate the difference between the two mechanisms, in Figure 12 we plot the ratio between the higher and lower throughputs for three different situations: without MRED, with MRED and proportional fair. We can see that applying MRED with current TCP implementations gives an intermediate situation, improving on the higher throughputs with respect to the current situation.

5.3.4. An application to a mixed wired-wireless tree topology

In this example, we simulate the topology of Figure 2 with two distribution APs and two users in each AP. The access link capacity is $c_{access} = 20Mbps$ representing a typical access capacity (e.g. a DSL line). The distribu-

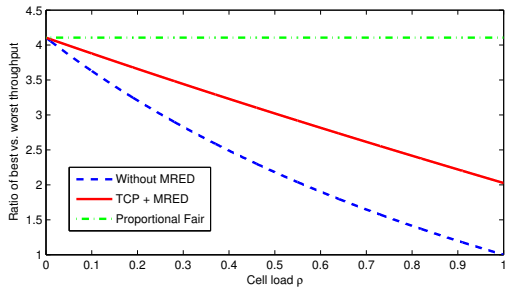

Figure 12: Ratio of best vs. worst throughput for an IEEE 802.11g cell.

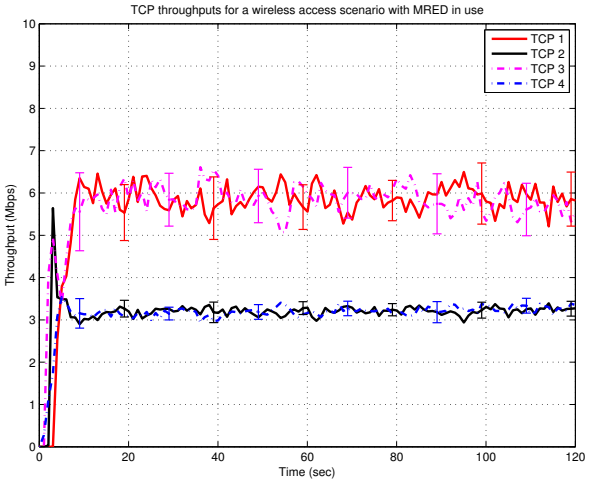

Figure 13: Throughputs of TCP connections for a wireless access scenario with 4 users. MRED is in use.

tion links have $c_{dist} = 100Mbps$ and thus are overprovisioned. The wireless cells are identical and have each one two users, with modulation rates $PHY_1 = PHY_3 = 54Mbps$ and $PHY_2 = PHY_4 = 6Mbps$. Each user has a single TCP connection and all connections have equal RTTs.

Plugging these values in Problem 3 gives the following allocation:

$$x_1^* = x_3^* = 6.4Mbps \quad x_2^* = x_4^* = 3.2Mbps$$

Note in particular that both the access link and the wireless cells are saturated in the resulting allocation. This is a difference with typical wired-only models with tree topologies. In particular, in this case, there is a positive price (queueing delay) both at the APs and the wired access link.

By using the MRED algorithm as discussed in Section 3.3, we can drive the system to this allocation. Results are shown in Figure 13, where again 50 independent replications were performed, and the average throughputs as well as 95% confidence intervals are shown. We see that the throughputs appoximately converge to the above equilibrium.

Note also that, if we choose not to use the MRED algorithm, the allocation will be given by the solution of Problem 2, which is $x_i^* \approx 3.8Mbps$ for each user. In that case, the full access capacity will not be used.

6. Conclusions

In this paper we applied the Network Utility Maximization framework to characterize the cross-layer interaction between the TCP transport protocol with an underlying MAC where multiple modulation rates coexist. This situation is present in typical IEEE 802.11 deployment scenarios. We described the resource allocation imposed by current wireless networks in this framework, showing that a bias is imposed against users of high modulation rates.

We then proposed an alternative resource allocation that generalizes the fairness and efficiency notions of TCP in wired networks to this context, and overcomes the inefficiencies of current protocols. We developed a simple mechanism to impose these more efficient equilibria in single cell scenarios and generalizations of this procedure to more complex topologies.

We also showed how the connection level dynamics can be analyzed through a Markov process, which can be identified in some cases with a well known queue. This enables us to characterize the stability regions and connection-level throughput obtained by the current resource allocation and our proposed alternative.

Finally we applied the previous results to the IEEE 802.11 MAC layer, establishing the effective data rates and validating the results by simulations.

In future work, we plan to extend the proposed mechanisms to the new 802.11 additions, where these issues may become more important due to the higher data rates involved, and due to packet aggregation mechanisms. We also would like to study the performance of a system where the uplink traffic is not negligible, as is the case in several of today usage models, where collisions have to be taken into account.

A. Appendix

Proof of Lemma 1. $\Phi(x)$ can be written as $\Phi(x) = g(f(x))$ where $f(x) = \sum_i x_i/C_i$ is a linear function of

x and $g(u) = (u - 1 - \log(u))\mathbf{1}_{\{u>1\}}$, where $\mathbf{1}$ represents the indicator function. It is easy to see that, for $u > 0$, $g'(u) = \max\{0, 1 - 1/u\}$ which is nonnegative and increasing function of u. Therefore, g is increasing and convex and thus Φ is convex [17]. □

Proof of Lemma 2. Denote by $V(x) = \sum_i \frac{1}{C_i} U_i(x_i) - \Phi(x)$ the objective function of Problem 1. We analyze the case $0 < \alpha < 1$, where we have the following bound:

$$\begin{aligned}
\sum_i \frac{1}{C_i} U_i(x_i) &= \sum_i \frac{K_i x_i^{1-\alpha}}{C_i(1-\alpha)} \\
&= \sum_i \frac{K_i C_i^{-\alpha}}{1-\alpha} \left(\frac{x_i}{C_i}\right)^{1-\alpha} \\
&\leqslant K \left(\max_i \frac{x_i}{C_i}\right)^{1-\alpha} \\
&\leqslant K \left(\sum_i \frac{x_i}{C_i}\right)^{1-\alpha}.
\end{aligned}$$

Here $K = \max_i \left\{\frac{K_i C_i^{-\alpha}}{1-\alpha}\right\}$. Let $y = \sum_i x_i/C_i$ and assume $y > 1$. Using the previous bound we have:

$$V(x) \leqslant K y^{1-\alpha} - y + 1 + \log(y) = h(y) \qquad (22)$$

Now, $h(y) \to -\infty$ when $y \to \infty$, since $\alpha > 0$. Therefore $V(x) \to -\infty$ when $y \to \infty$; since y is a norm over the positive orthant, the same conclusion holds when $\|x\| \to \infty$ for any (equivalent) norm. Also, $\{x : V(x) \geq \gamma\} \subset \{x : h(y) \geq \gamma\}$ which is compact since $h(y) \to -\infty$.

For $\alpha = 1$, we can use the bound $x_i \leq c_i y$ to get $\sum_i \frac{1}{C_i} U_i(x_i) \leq \sum_i \frac{1}{C_i} U_i(C_i) + K \log(y)$ which gives a bound analogous to (22) and the rest follows. If $\alpha > 1$, the bound is even simpler since $\sum_i \frac{1}{C_i} U_i(x_i) \leq 0$. □

Acknowledgments

This work was partially supported by LATU/Ceibal, CSIC I+D (Universidad de la Repblica), ANII-FCE and AFOSR-US. We thank José García for assistance with the ns2 implementation.

References

[1] IEEE 802.11-2007, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, `http://www.ieee802.org/11/`, 2007.

[2] G. Bianchi, Performance analysis of the IEEE 802.11 distributed coordination function., IEEE Journal on Selected Areas in Communications 18 (2000) 535–547.

[3] A. Kumar, E. Altman, D. Miorandi, M. Goyal, New insights from a fixed-point analysis of single cell IEEE 802.11 WLANs, IEEE/ACM Transactions on Networking 15 (2007) 588–601.

[4] F. Lebeugle, A. Proutiere, User-level performance of WLAN hotspots, in: Proceedings of the 19th International Teletraffic Conference.

[5] F. Kelly, A. Maulloo, D. Tan, Rate control in communication networks: shadow prices, proportional fairness and stability, Journal of the Operational Research Society 39 (1998) 237–252.

[6] M. Chiang, S. H. Low, A. R. Calderbank, J. C. Doyle, Layering as optimization decomposition: A mathematical theory of network architectures, in: Proceedings of the IEEE, volume 95, pp. 255–312.

[7] X. Lin, N. B. Shroff, R. Srikant, A tutorial on cross-layer optimization in wireless networks, IEEE Journal on Selected Areas in Communication (2006) 1452–1463.

[8] L. Jiang, J. Walrand, A distributed CSMA algorithm for throughput and utility maximization in wireless networks, in: Proceedings of the Forty-Sixth Annual Allerton Conference on Communication, Control, and Computing.

[9] A. Proutire, Y. Yi, M. Chiang, Throughput of random access without message passing, in: Proceedings of the 44th Conference on Information Science and Systems (CISS 08).

[10] G. de Veciana, T.-J. Lee, T. Konstantopoulos, Stability and performance analysis of networks supporting services with rate control - could the internet be unstable?, in: IEEE/Infocom, pp. 802–810.

[11] T. Bonald, L. Massoulié, Impact of fairness on internet performance, in: ACM SIGMETRICS/Performance, pp. 82–91.

[12] A. Ferragut, F. Paganini, A connection level model of IEEE 802.11 cells, in: IFIP/ACM Latin American Networking Conference (LANC'09).

[13] A. Ferragut, J. Garcia, F. Paganini, Network utility maximization for overcoming inefficiency in multirate wireless networks, in: 8th Intl. Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt'10).

[14] R. Srikant, The Mathematics of Internet Congestion Control, Birkhuser, Boston, MA, 2004.

[15] J. Mo, J. Walrand, Fair end-to-end window based congestion control., IEEE/ACM Transactions on Networking 8 (2000) 556–567.

[16] H. K. Khalil, Nonlinear Systems, Prentice-Hall, Upper Saddle River, NJ, 1996.

[17] S. Boyd, L. Vanderberghe, Convex Optimization, Cambridge University Press, Cambridge, UK, 2004.

[18] K. Arrow, L. Urwicz, H. Uzawa, Studies in Linear and Nonlinear Programming, Stanford University Press, Stanford, CA, 1958.

[19] D. Feijer, F. Paganini, Krasovskii's method in the stability of network control, in: Proceedings of the American Control Conference.

[20] S. H. Low, D. Lapsley, Optimization flow control, I: basic algorithm and convergence, IEEE/ACM Transactions on Networking 7 (1999) 861–874.

[21] S. H. Low, F. Paganini, J. C. Doyle, Internet congestion control, IEEE Control Systems Magazine 22 (2002) 28–43.

[22] F. Baccelli, S. Zuyev, Stochastic geometry models of mobile communication networks, in: Frontiers in Queueing, CRC, Boca Raton, FL, 1997, pp. 227–243.

[23] E. Altman, K. Avrachenkov, U. Ayesta, A survey on discriminatory processor sharing, Queueing Systems: Theory and Applications 53 (2006) 53–63.

[24] G. Fayolle, I. Mitrani, R. Iasnogorodski, Sharing a processor among many job classes, Journal of the ACM 27 (1980) 519–532.

[25] M. Haviv, J. van der Wal, Mean waiting times for phase-type discriminatory processor sharing, European Journal of Operations Research 189 (2008) 375–386.

[26] T. Bonald, L. Massoulié, A. Proutiere, J. Virtamo, A queueing analysis of max-min fairness, proportional fairness and balanced fairness, Queueing Systems: Theory and Applications 53 (2006) 65–84.
[27] W. Feller, An Introduction to Probability Theory and Its Applications, John Wiley & Sons, New York, 1965.
[28] S. McCanne, S. Floyd, ns Network Simulator, `http://www.isi.edu/nsnam/ns/`, 2000.
[29] dei80211mr: a new implementation of 802.11 for ns2, `http://www.dei.unipd.it/wdyn/?IDsezione=5090`, 2007.