Rigorous and Practical Proportional-fair Allocation for Multi-rate Wi-Fi

Paul Patras[1], Andrés Garcia-Saavedra[2], David Malone[3], and Douglas J. Leith[2]

[1]School of Informatics, The University of Edinburgh, UK
[2]School of Computer Science & Statistics, Trinity College Dublin, Ireland
[3]Hamilton Institute, National University of Ireland Maynooth, Ireland

Abstract

Recent experimental studies confirm the prevalence of the widely known *performance anomaly* problem in current Wi-Fi networks, and report on the severe network utility degradation caused by this phenomenon. Although a large body of work addressed this issue, we attribute the refusal of prior solutions to their poor implementation feasibility with off-the-shelf hardware and their imprecise modelling of the 802.11 protocol. Their applicability is further challenged today by very high throughput enhancements (802.11n/ac) whereby link speeds can vary by two orders of magnitude. Unlike earlier approaches, in this paper we introduce the first rigorous analytical model of 802.11 stations' throughput and airtime in multi-rate settings, without sacrificing accuracy for tractability. We use the proportional-fair allocation criterion to formulate network utility maximisation as a convex optimisation problem for which we give a closed-form solution. We present a fully functional light-weight implementation of our scheme on commodity access points and evaluate this extensively via experiments in a real deployment, over a broad range of network conditions. Results demonstrate that our proposal achieves up to 100% utility gains, can double video streaming goodput and reduces TCP download times by 8x.

1 Introduction

Wi-Fi technology has seen remarkable uptake in recent years through leveraging unlicensed spectrum and employing a simple decentralised channel access paradigm [1]. Specifically, client stations that follow the IEEE 802.11 specification use a distributed coordination function (DCF) to schedule their transmission, implementing a medium access control (MAC) protocol that assigns equal transmission opportunities to contenders [2]. By design, this method ensures stations receive similar throughput, irrespective of their individual link qualities, which is desirable when clients experience good channel conditions. In practical deployments, however, stations employ different modulation and coding schemes (MCSs) to preserve transmission robustness as link qualities change. MCSs of lower indexes handle channel errors better, but have inferior spectral efficiency, which yields lower bit rates. Thus in multi-rate scenarios, stations that run DCF and transmit at lower rates will retain access to the channel for longer periods of time. This degrades the overall network utility, since a significant fraction of channel airtime would be used more efficiently if allowing more frequent transmissions of faster clients. This pathological behaviour is widely known as *performance anomaly* and was first identified with the adoption of high-rate 802.11b services [3]. The effect is dramatically exacerbated when stations that make use of very high throughput protocol enhancements are present in the network. Despite being able to transmit at bit rates up to e.g. 780Mb/s (with 802.11ac [4], single stream over 160MHz channel, using 256-QAM), their performance is capped at that of the slowest station in the wireless LAN (WLAN), as we illustrate with dark bars in the example shown in Fig. 1. Also plotted in the figure with light bars is the the network performance when the access point (AP) implements the rigorous proportional-fair (RPF) allocation scheme introduced in this paper, demonstrating its effectiveness in providing faster users with significantly higher throughput, while ensuring all stations receive equal airtime. This contrasts sharply with the legacy DCF, which assigns highly unfair shares of the available channel time to stations that operate with different bit rates (bottom sub-plot).
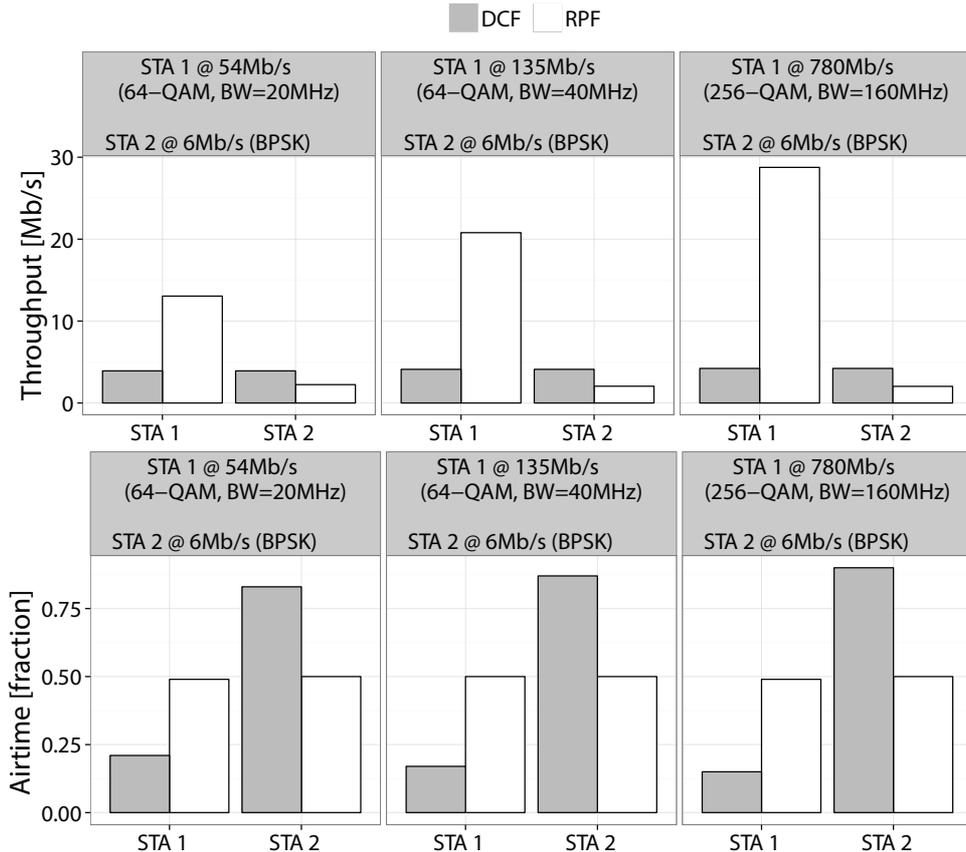
Figure 1: WLAN with 2 backlogged stations sending 1000-byte packets. STA 2 transmits at 6Mb/s (BPSK) on a 20MHz channel. STA 1 transmits at 54Mb/s (20MHz channel, 64-QAM), 135Mb/s (40MHz channel, 64-QAM), and respectively at 780Mb/s (160MHz channel, 256-QAM). Individual throughput (above) and airtime (below) with DCF (dark bars) and respectively with the proposed rigorous proportional-fair (RPF) allocation scheme (light bars). Analytical model.

The proportional-fair allocation criterion is that which maximises the network utility[1] while providing stations with good airtime fairness [5]. Proportional fairness in IEEE 802.11 WLANs has been extensively studied in the context of high- and extended-rate services, i.e. 802.11b and 802.11a/g [7, 8, 9, 10, 11, 12, 13, 14, 15]. Previous proposals that target equal airtime allocation, however, suffer from at least one of the following key limitations: *(i)* their underlying analyses make several simplifying assumptions that do not capture accurately the 802.11 protocol details, *(ii)* assume error-free channel conditions, and *(iii)* cannot be implemented with existing off-the-shelf devices, as they require hardware/firmware modifications or changes to the 802.11 state machine and thus are not standard compliant.

In contrast to earlier works, in this paper we provide, to the best of our knowledge, the first rigorous analysis of 802.11 stations' throughput and airtime, that does not sacrifice accuracy for tractability. Precisely, our model captures accurately the 802.11 protocol operation and all the possible transmission outcomes in multi-rate settings, allowing for different packet lengths, channel bandwidths, and PHY bit rates, while also accounting for dissimilar data link frame error rates, as typically encountered in practice. We argue that considering different packet durations is particularly important in practice, as these directly impact on collisions and thus on the fraction of time the channel can be used for successful transmissions. Capturing all protocol details was often avoided to make 802.11 performance analysis tractable (see e.g. [9, 12]). We show, however, that the contrary holds and based on our analysis we formulate proportional-fair allocation as a convex optimisation problem for which we give a closed-

---

[1]In this paper we use the same definition for the network utility as given by F. Kelly, i.e. the sum of the logarithms of individual throughputs [5]. This is appropriate for the *elastic data* applications we consider herein, e.g. file transfer, electronic mail, web browsing, etc. [6].

form solution. Our approach finds the *optimal contention window* configurations 802.11 stations must employ to maximise utility in multi-rate WLANs.[2] To demonstrate the implementation feasibility of our theoretical results, we develop a practical yet very accurate proportional-fair allocation scheme with open-source drivers on commodity APs. The key advantage of our design is that it requires no modifications to clients' protocol stack, but only the AP to compute basic statistics of correctly received frames and distribute the optimal MAC parameters through *unicast beacons*. To the best of our knowledge, our prototype is the first of its kind to achieve utility figures closed to the theoretical optima, while being standard compliant and amenable to incremental deployment on commodity off-the-shelf (COTS) Unix/Linux based wireless routers and access points.

We evaluate the performance of our solution by conducting extensive experiments in a real 802.11 network over a broad range of multi-rate settings, types of traffic and channel conditions. The results obtained demonstrate that our approach improves network utility by up to 100%, can double the goodput attained by video streaming applications and can reduce the transfer time of TCP downloads by up to 8 times. Further, we demonstrate that our scheme adapts quickly to rate control decisions and can equalise the throughput of stations that transmit at the same bit rate, but experience dissimilar performance due to capture effect.

In the remainder of this paper we briefly discuss the limitations of prior works (§2) and then present a rigorous analytical model of multi-rate 802.11 operation (§3) based on which we formulate and solve explicitly the proportional-fair allocation task as a convex optimisation problem (§4). We detail a practical prototype implementation of our solution (§5) and report the results of the experimental evaluation undertaken (§6), before concluding the paper (§7).

2 Related Work

Recent experimental studies provide substantial evidence of the prevalence and severity of the rate anomaly problem in current Wi-Fi deployments (see e.g. [16, 17]). The issue persists despite the large body of work conducted in this space since Heusse *et al.* first analysed this behaviour [3]. We attribute this to the poor implementation feasibility of prior approaches addressing proportional fairness with off-the-shelf hardware and their inaccurate modelling of the 802.11 protocol, shortcomings that we particularly tackle in this paper. Here we briefly summarise the most relevant research efforts and highlight the key advantages of our proposal.

Analytical Modelling: [10] introduces a general utility maximisation framework for wireless networks, though the analysis does not capture MAC protocol details and is only validated via simulations. The airtime fairness of 802.11 is investigated in [13], but protocol overhead is ignored and the impact of collisions is neglected. Similarly, the model in [12] neglects collisions of more than two stations and uses a coarse approximation of the average slot time. A more detailed analysis is given in [9]; losses due to channel noise, however, are not considered and the authors assume all collisions to be of equal duration. Le *et al.* [7, 8] give a simplified model of 802.11 multi-rate operation, underestimating the impact of slower clients on collisions, and design a distributed but non-compliant contention window adaptation scheme that is not formally proven to converge. A more accurate multi-rate 802.11 model is presented in [11]. The authors attempt to solve a *modified* proportional-fair allocation problem, but do not provide a closed-form solution, as this requires explicit computation of the expected slot time, which proves infeasible. Li *et al.* [14] formulate the utility maximisation in multi-rate WLANs as a client association task and give algorithms that solve a relaxed version of the problem. This, however, only applies to multi-AP deployments managed by a single entity. Proportional fairness is also studied in multi-hop scenarios, but their complexity requires several questionable simplifications, while resource allocation is subject to non-trivial estimation of individual loads [18] or mixed bias strategies [19].

Prototyping & Experimentation: [15] undertakes an empirical study to find the contention window settings that achieve proportional fairness, but lacks analytical support and is limited to static scenarios. Heusse *et al.* propose to control stations' transmission opportunities based on the observed number of idle slots, to tackle airtime fairness [20]; however, the implementation requires precise time synchronisation and is tightly coupled to specific hardware and a proprietary closed-source firmware [21]. Lee *et al.*

---

[2]NB: Employing the transmission opportunity (i.e. frame bursting) feature to equalise *success airtimes* is not effective for proportional fairness, as this requires total airtime, including collisions, to be the same across all stations.

Table 1: Comparison between previous works on proportional-fair allocation and our proposal

	RPF	[10]	[12]	[13]	[7, 8, 9]	[11]	[15]	[20, 21]	[22]	[17, 23]
Models different link error rates	✓	.	.	.	.	✓	.	.	.	.
Models precisely simultaneous collisions	✓	.	.	.	.	✓	.	.	.	.
Accounts for different payload sizes	✓	.	✓	✓	✓	✓	.	.	.	.
Accounts for dissimilar transmission & collision durations	✓	.	✓	.	.	✓	.	.	.	.
Does not require client side changes	✓	.	.	.	.	.	.	.	.	.
Suitable for implementation on the AP without FW/HW modifications	✓	.	✓	✓	.	.	✓	.	DL†	DL†
Prototyped with COTS	✓	.	.	.	.	.	✓	✓	✓	✓
Evaluated experimentally	✓	.	.	.	.	.	✓	✓	✓	✓

†AP-side implementation handles *only downlink* traffic.

implement O-DCF [22], whereby a station's packet rate is adjusted according to the MCS employed, to improve network utility. This approach requires introducing an additional queueing layer between application and driver. Similarly, individual queues are introduced and controlled for each destination in WiFox [17]. ADWISER [23] tackles rate anomaly only in the *downlink*, by introducing a dedicated network entity that performs scheduling before the AP.

Our Contributions: In contrast to the aforementioned works, in this paper we give an accurate model of 802.11 throughput and airtime, accounting for different packet sizes, channel bandwidths, MCSs, and frame error rates. Based on our analytical results we formulate utility maximisation as a convex optimisation problem that we solve explicitly. Further, we provide a practical and accurate implementation of the proposed resource allocation scheme with commodity APs. Unlike other approaches, our solution does not require modification to the clients' protocol stack, which makes it deployable with client equipment of any vendor, and is demonstrably effective over a broad range of network conditions. We summarise the key advantages of our proposal (RPF) as compared to previous work in Table 1, which highlights the comprehensive analytical and practical nature of our contribution.

3 Analytical Model

In this section we give a rigorous analysis of the throughput and airtime attained by wireless stations in multi-rate Wi-Fi networks that operate in the default infrastructure mode, i.e. where all packets are transmitted through the AP. We consider a single-hop 802.11 WLAN with N clients able to select from a fixed set of possible PHY bit rates for transmission, for instance {6, 9, 12, 18, 24, 36, 48 and 54} Mb/s available with the 5 GHz OFDM PHY layer (802.11a) [2]. In our analysis we consider all stations are saturated, i.e. they always have a packet enqueued for transmission, but later also investigate scenarios with non-saturated real-time traffic. Our analysis allows for arbitrary packet sizes and link error probabilities, either due to signal fading and noise or hidden terminal circumstances that may arise in the presence of different BSSs with partially overlapping coverage. We focus on the single AP case, but argue that, with the advent of software defined technologies, our analysis can be easily extended to dense multi-AP networks underpinned by a management scheme, e.g. as in enterprise or university campus deployments [24].

3.1 802.11 Operation

Wireless stations that follow the IEEE 802.11 specification contend for the medium using two key channel access parameters, namely the minimum and the maximum contention windows, CW_{min} and $CW_{max} = 2^m CW_{min}$ (where m is the maximum backoff stage). Precisely, to transmit a packet a station will initialise a backoff counter with a random number uniformly distributed in the [0,CW-1] interval, decrement its value every time slot, and transmit when the counter reaches zero. The contention window is initialised with the CW_{min} value upon the first attempt, doubled up to CW_{max} upon failures (either

due to channel errors or collisions with other simultaneous transmissions), and reset when a frame is delivered successfully. Although the standard defines a set of recommended medium access parameters, it allows the AP to change their values and distribute these periodically to associated stations by means of beacon frames [2].

For the analysis we undertake below, it may prove useful to refer to Table 2 in the Appendix, where we summarise the notation used throughout.

3.2 Throughput Analysis

In our analysis we assume station i contends for the medium with a probability τ_i in a randomly chosen slot. To achieve a particular τ_i in practice, we configure a station with $W_i = \mathrm{CW}_{\min,i} = \mathrm{CW}_{\max,i}$, i.e. $m_i = 0$ where, following [25],[3]

$$W_i = \frac{2 - \tau_i}{\tau_i}. \tag{1}$$

Practical issues of rounding W_i are addressed in §5.

Recall that we account for the fact that nodes may experience different link qualities to the AP and let's denote $p_{n,i}$ the probability that a transmission of station i fails due to channel errors (noise, hidden terminals or interference), which can be estimated practically using e.g. the packet pair technique proposed in [27]. Thus, the we can express the conditional failure probability experienced by a station i as follows,

$$p_{f,i} = 1 - (1 - p_{n,i})(1 - p_i),$$

where p_i denotes the collision probability experienced by a packet transmitted by this station and is given by

$$p_i = 1 - \prod_{j=1, j \neq i}^{N} (1 - \tau_j).$$

The throughput obtained by station i can be expressed as

$$S_i = \frac{p_{s,i} L_i}{P_e T_e + P_s T_s + P_u T_u}, \tag{2}$$

where $p_{s,i}$ is the probability that the station's transmission is successful and L_i denotes the length of the packet payload generated.[4] P_e, P_s and P_u are the expected probabilities that a slot is empty (idle), contains a successful, and respectively an unsuccessful transmission (either due to collision or channel errors), while T_e, T_s and T_u are their corresponding durations. We provide expressions of the above quantities next.

We compute the probability of a successful transmission of a station i as

$$p_{s,i} = \tau_i (1 - p_{f,i}) = \tau_i (1 - p_{n,i}) \prod_{j=1, j \neq i}^{N} (1 - \tau_j) \tag{3}$$

and the probabilities P_e, P_s and P_u as

$$P_e = \prod_{i=1}^{N} (1 - \tau_i), \tag{4}$$

$$P_s = \sum_{i=1}^{N} p_{s,i}, \tag{5}$$

$$P_u = 1 - P_e - P_s. \tag{6}$$

---

[3]While this holds for saturation conditions, different packet arrival rates could be considered, e.g. it would be possible to incorporate the probability q_i that a packet is available for transmission when station i wins a transmission opportunity, as suggested in [26], which gives $W_i = (2q_i - \tau_i)/\tau_i$.

[4]While here we consider stations send a single packet upon channel access, the current model applies unchanged to 802.11n, where multiple packets are allowed with the same attempt (i.e. frame aggregation).

T_e is a PHY layer constant (e.g. 9μs for 802.11a/g). In order to calculate the expected durations of a slot containing a success (T_s) and respectively a failure (T_u), we will index the stations in order of increasing transmission duration. Then

$$T_s = \sum_{i=1}^{N} \frac{p_{s,i}}{P_s} T_{s,i},$$

$$T_u = \sum_{i=1}^{N} \frac{p_{u,i}}{P_u} T_{u,i},$$

where $p_{u,i}$ is the probability that a slot contains an unsuccessful transmission of stations of highest index i, while $T_{s,i}$ and $T_{u,i}$ are the durations of a slot containing a successful and respectively a failed transmission of these stations.

By labelling stations according to their transmission durations and considering the event where a station i is unsuccessful either due to channel errors or due to a collision with a station of lower index, the probability that a slot contains a failure of a station with highest index i is

$$p_{u,i} = \tau_i p_{n,i} \prod_{j=1,j\neq i}^{N} (1-\tau_j) + \tau_i \left(1 - \prod_{j=1}^{i-1}(1-\tau_j) \right) \prod_{j=i+1}^{N} (1-\tau_j). \tag{7}$$

Note that, unlike previous studies, in the above *we do not neglect the probability that more than two stations collide*. Also notice that $P_u = \sum_{i=1}^{N} p_{u,i}$ and the sum of the probabilities of all the possible slot events expressed in (6) is satisfied.[5]

We compute $T_{s,i}$ as

$$T_{s,i} = T_{PLCP} + \frac{H + L_i}{C_i} + SIFS + T_{ack} + DIFS,$$

where T_{PLCP} is the duration of the PLCP (Physical Layer Convergence Protocol) preamble and header, H is the MAC overhead (header and frame check sequence), C_i is the PHY rate employed for transmission (accounting also for wider bandwidth channels), and T_{ack} is the duration of an acknowledgement (ACK). SIFS (Short Inter-frame Space) and DIFS (DCF Inter-frame Space) are PHY layer constants separating a data frame from an ACK and respectively preceding the backoff process (e.g. 16μs and 34μs for 802.11a/g).

Similarly, the duration of a failure involving stations of highest index i is given by

$$T_{u,i} = T_{PLCP} + \frac{H + L_i}{C_i} + EIFS,$$

where EIFS is the Extended Inter-frame Space and is a PHY layer constant that is derived from SIFS, DIFS and the time it takes to transmit an ACK frame at the lowest PHY rate, i.e. EIFS = SIFS + DIFS + $T_{ack}(C_{\min})$ [2]. To reduce notation clutter, we assume that $T_{s,i} \simeq T_{u,i}$, $\forall i$, since the duration of an ACK is dominated by the PLCP preamble and header. This completes our throughput analysis.

3.3 Airtime

Next, we analyse the *total airtime* of a station, i.e. the fraction of time the channel is occupied by the (successful or unsuccessful) transmission of that station. Denoting T_{slot} the average slot duration ($T_{slot} = P_e T_e + P_s T_s + P_u T_u$), the total airtime used by a station i is given by

$$\begin{aligned}
T_i = \quad & \frac{1}{T_{slot}} \left(\tau_i(1-p_{n,i}) \prod_{j=1,j\neq i}^{N} (1-\tau_j)T_{s,i} + \tau_i p_{n,i} \prod_{j=1,j\neq i}^{N} (1-\tau_j)T_{s,i} \right. \\
+ \quad & \left. \tau_i \left(1 - \prod_{j=1}^{i-1}(1-\tau_j) \right) \prod_{j=i+1}^{N} (1-\tau_j)T_{s,i} + \tau_i \sum_{j=i+1}^{N} \tau_j \prod_{k=j+1}^{N} (1-\tau_k)T_{s,j} \right).
\end{aligned}$$

---

[5]The duration of a collision is dominated by the frame with the longest duration involved in that collision and collisions should only be counted once. Though by labelling stations according to their transmission duration, we do not eliminate any of the possible collision scenarios, which is verified by the unit sum of the probabilities of *all* the possible slot events.

Note that we account for all the possible outcomes of the transmission of a station i, i.e. successful reception, failure due to channel errors, collision with at least one station of lower index, and respectively collision with a station of higher index, while we express T_i relatively to the average slot duration T_{slot}. After algebra the airtime expression can be reduced to

$$T_i = \frac{\tau_i}{T_{slot}} \left(\prod_{j=i+1}^{N} (1 - \tau_j) T_{s,i} + \sum_{j=i+1}^{N} \tau_j \prod_{k=j+1}^{N} (1 - \tau_k) T_{s,j} \right). \tag{8}$$

It is interesting to observe in the above that the airtime expression does not depend on the individual link error probabilities. Let us further rewrite T_{slot} as

$$T_{slot} = T_e \prod_{j=1}^{N} (1 - \tau_j) + \sum_{j=1}^{N} T_{s,j} \tau_j \prod_{k=j+1}^{N} (1 - \tau_k).$$

It is now easy to observe that we give T_i as a function of only stations' transmission attempt probabilities τ_i and their corresponding successful transmission durations. As we will show later, this has important practical implications on the design of our proportional-fair allocation prototype.

It will also prove useful to work in terms of the transformed variable $x_i = \tau_i/(1 - \tau_i)$. Then (8) becomes

$$T_i = \frac{x_i}{X} \left(T_{s,i} \prod_{j=1}^{i-1} (1 + x_j) + \sum_{j=i+1}^{N} T_{s,j} x_j \prod_{k=1, k \neq i}^{j-1} (1 + x_k) \right), \tag{9}$$

where

$$X(x) = T_e + \sum_{j=1}^{N} \left(T_{s,j} x_j \prod_{k=1}^{j-1} (1 + x_k) \right).$$

The above completes our airtime analysis. In the remainder of this section we establish convexity properties for the throughput expression derived previously, which will allow us to model proportional-fair allocation as a convex optimisation problem and solve this explicitly.

3.4   Convexity Properties

To guarantee that the utility maximisation problem we formulate and pursue in this work has a solution that is a global optimum, it is first necessary to establish the convexity properties of the logarithmic utility function $U = \sum_i \log S_i$ associated with the proportional fairness criterion.

Let us first rewrite the expression of a station's throughput (2) in terms of the transformed variables x_i. Then

$$S_i = (1 - p_{n,i}) \frac{x_i}{X} L_i \tag{10}$$

and recall that network utility is the sum of the logarithms of individual throughputs, i.e. $U = \sum_{i=1}^{N} \log S_i$.

Lemma 1. *Let $f : \mathbb{R}^n \to \mathbb{R}, f(x) = \sum_{i=1}^{n} b_i e^{a_i^T x + c_i}$, where a_i is a column vector, $b_i > 0$ and $c_i \geq 0$. Then $\log f(x)$ is convex.*

Proof. As $f(x)$ is a sum of exponentials of affine functions, by Boyd and Vandenberghe, p.74 & p.79 [28] it follows that $\log f(x)$ is convex. □

Lemma 2. $\log X(e^{\tilde{x}})$ *is convex in $\tilde{x}$.*

Proof. Letting $\tilde{x}_i = \log x_i$, we have

$$X(e^{\tilde{x}}) = T_e + \sum_{j=1}^{N} \left(T_{s,j} e^{\tilde{x}_j} \prod_{k=1}^{j-1} (1 + e^{\tilde{x}_k}) \right)$$

Expanding the RHS further, it is can be seen that X is of the form $\sum_i b_i e^{a_i^T \tilde{x} + c_i}$. By Lemma 1 it follows that $\log X(e^{\tilde{x}})$ is convex in $\tilde{x}$. □

Thus the *network utility U can be expressed as an affine combination of convex terms, and therefore is convex.*[6]

4   Proportional-fair Allocation

In this section we formulate proportional-fair allocation as a convex optimisation problem whose objective is network utility maximisation. We solve this problem explicitly and show that the stations' airtimes are equalised at the optimum. We then explain how the solution is used to derive the stations' contention window configurations that achieve this objective.

Let us first denote $z_i = 1 - p_{n,i}$. The proportional-fair throughput allocation is the solution to the following optimisation problem

$$\max_x \sum_{i=1}^N \log S_i$$
$$s.t. \ S_i \leq z_i \frac{x_i}{X} L_i, \ i = 1, 2, \ldots, N; \quad \text{(throughput feasibility)}$$
$$0 \leq x_i, \ i = 1, 2, \ldots, N. \quad (0 \leq \tau_i \leq 1)$$

Rewriting the optimisation in terms of the log-transformed variables $\tilde{x}_i = \log x_i$, $\tilde{z}_i = \log z_i$ and $\tilde{S}_i = \log S_i$ the problem becomes

$$\max_{\tilde{x}} \sum_{i=1}^N \tilde{S}_i \tag{11}$$
$$s.t. \ \tilde{S}_i - \tilde{z}_i - \tilde{x}_i + \log X - \log L_i \leq 0, \ i = 1, 2, \ldots, N. \tag{12}$$

Following the results of §3.4, this optimisation problem is convex, hence a solution exists. The Slater condition is satisfied and so strong duality holds. The Lagrangian is

$$L = -\sum_{i=1}^N \tilde{S}_i + \sum_{i=1}^N \lambda_i \left(\tilde{S}_i - \tilde{z}_i - \tilde{x}_i + \log X - \log L_i \right)$$

The Karush-Kuhn-Tucker (KKT) condition [29] for $\tilde{S}_i$ is

$$\frac{\partial L}{\partial S_i} = 0,$$

which yields

$$\lambda_i = 1.$$

It follows immediately from complementary slackness that constraint (12) is tight at the optimum. Since any optimal solution must lie on the boundary of the log-transformed rate region and, following [26], this is strictly convex, each boundary point has a unique supporting hyperplane, hence, the optimum is unique.

The solution is obtained by solving the KKT condition for $\tilde{x}_i$, i.e.

$$\frac{1}{x_i} - \frac{1}{X} \frac{\partial X}{\partial x_i} N = 0, \forall i = 1, 2, \ldots N.$$

Solving the partial derivative, rearranging the above, and using (9), yields

$$T_i = \frac{1}{N}, \forall i. \tag{13}$$

---

[6]Denoting $z_i = 1 - p_{n,i}$, then U can be expanded to $U = \sum_i (\log z_i + \log x_i + \log L_i - \log X)$. The terms $\log z_i$ and $\log L_i$ are constant, $-\log x_i$ is convex, and by Lemma 2 $\log X$ is also convex. It follows that the utility is a linear (affine) combination of convex terms, which is convex.
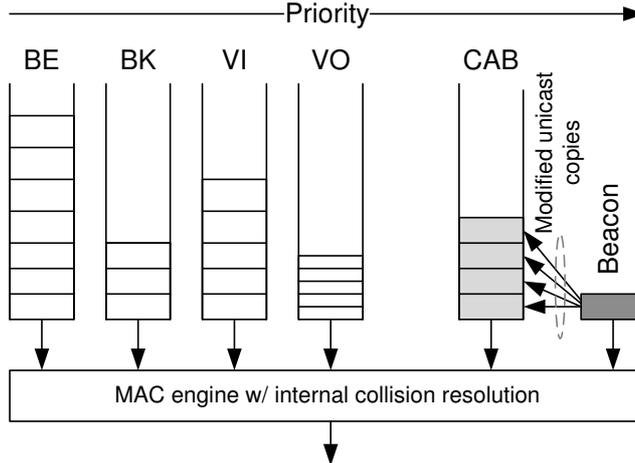
Figure 2: Software queues exposed by `mac80211` and the implementation principle behind the proposed unicast beaconing mechanism. A copy of the broadcast beacon is made for each client, updated with the corresponding destination address and CW settings, and placed on the CAB queue.

This means *the solution to the proportional-fair allocation problem assigns equal airtime to all stations, inversely proportional to their total number.* It is also straightforward to observe that the assigned airtimes sum to unity.

With (8) and (13) we form a system of N equations and N unknowns (the transmission attempt probability τ_i of each station), which we can solve for τ_i numerically. Then, using (1) we can compute the CW configuration of each station to maximise network utility. As we explain next, our AP-based implementation with open-source drivers distributes the computed CWs to clients by means of unicast beacon frames.

Note that deriving a closed-form solution analytically is a significant result, as the KKT conditions for non-linear programs such as ours cannot be solved directly in most cases. This enables us to develop a practical implementation that solves the optimisation in real-time, taking into consideration the changing network conditions, while running on commodity APs with open-source drivers and requiring minimal computational effort.

5 Prototype Implementation

In this section we present a prototype we develop based on our analytical results to enforce proportional-fair allocation in real multi-rate Wi-Fi deployments. *The key advantage of our design is that it only involves light-weight software updates incrementally deployable at the AP and no modifications to the clients' software or hardware.* Further, our approach is modular and thus suitable for future research purposes.[7]

We base our implementation on commercial off-the-shelf APs equipped with 802.11 wireless cards and running the Linux `mac80211` subsystem that provides a modular framework for wireless device drivers with fine-grained hardware control [30]. Our solution requires only small changes to `mac80211`[8] and has the major benefit of being independent of any underlying (compliant) hardware, e.g. Atheros, Broadcom, etc. The changes we introduce enable the AP to estimate the average duration of successful transmissions, as well as to distribute the CW configurations to the associated clients, while the optimisation problem is solved in user-space with minimum effort. Note that we run the optimisation periodically (every beacon interval), to ensure our scheme adapts to changes in the WLAN. Our implementation comprises thus two key building blocks: a modified kernel module and an user-space optimisation tool.

---

[7]We intend to make our code available to the community upon publication.
[8]The patch comprising our `mac80211` modifications is *only* 14KB.
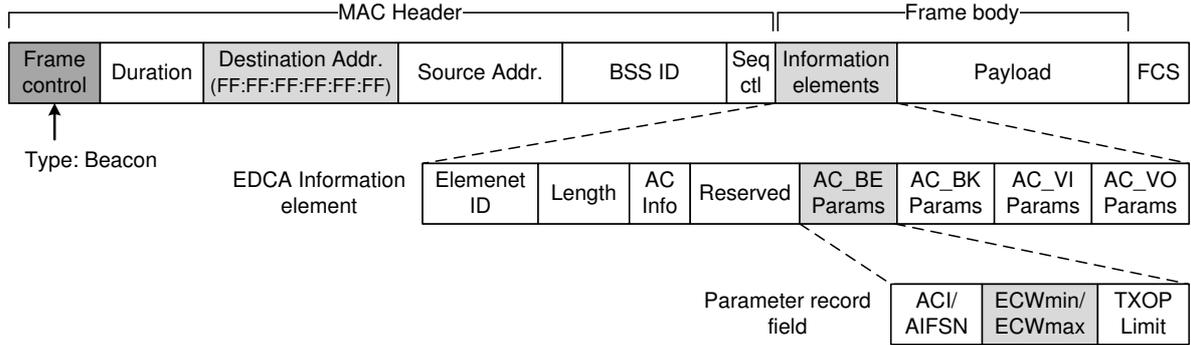
Figure 3: Structure of a beacon management frame. Fields modified to generate unicast beacons that configure individual clients are highlighted. The destination address field is changed from broadcast to that of the target station. ECWmin and ECWmax settings are updated with the values that solve utility maximisation.

5.1 Kernel-space Modifications

First we develop the low level time sensitive functionality in kernel-space, by augmenting the `mac80211` capabilities to meet our requirements. The AP already collects per-station statistics and we report those in user-space via a dedicated `debugfs`[9] space↔user-space interface, while all packets pass through the AP in infrastructure operational mode. Therefore, between each iteration the AP aggregates the total time each station spends transmitting, by summing the duration of all the frames received correctly and counting their respective number, then passes this information to user-space using this interface. Similarly, we provide the driver with per-station CW settings that enforce the proportional-fair allocation. To distribute these to the clients, we exploit a standard feature of the 802.11 operation. More precisely, APs periodically broadcast beacon frames to advertise network presence and synchronise associated clients. Beacons can also contain information elements that enforce the contention parameters to be used in the WLAN. We extend this feature and generate *unicast beacons* following the broadcast instance, thus giving each station the CW that maximises network utility. By distributing CWs in this fashion, *we do not require any modifications to the user equipment.*

Note that drivers typically manage six software queues, as shown in Fig. 2. Four of these correspond to different access categories, i.e. best-effort (BE), background (BK), video (VI) and voice (VO), but as applications and routers tag the DiffServ Code Point (DSCP) field in the packets' IP headers with 0 by default, the BE queue is used for most data traffic.[10] The fifth queue handles multicast frames (referred to as the CAB queue), while the sixth is a dedicated beacon queue. The latter stores a single management frame, whose contents are updated every beacon interval and which is then passed to the hardware for transmission. The frame descriptor is never removed from the queue. Broadcast beacon transmission triggers immediately the CAB queue, which is emptied before handling data traffic. Therefore, we create a copy of the broadcast beacon for each station, change the destination address field to the MAC address corresponding to that client, write the computed CW value into the ECWmin and ECWmax fields of the EDCA (Enhanced Distributed Channel Access) information element (see Fig. 3), and finally load the unicast beacons into the CAB queue.

5.2 User-space Optimisation Tool

The second component of our implementation is a user-space module that retrieves the measurements from the driver, computes the average duration of a transmission of each station ($T_{s,i}$) and finds the optimal set of CWs that maximise network utility by solving the system of equations given by (13). The AP handles this task in user-space, since it is more computationally expensive and we do not wish

---

[9]RAM-based file system used for debugging purposes, where unlike with `procfs` and `sysfs`, no strict rules apply to information exchange.

[10]Traffic differentiation at the MAC layer requires appropriate DSCP tagging of the IP packets, which is largely overlooked in practice (see e.g. [31]). Thus data traffic is served by the best effort (BE) queue with the contention parameters we compute.

Figure 4: Test bed used for the evaluation of the proposed proportional-fair allocation scheme. We use nine Soekris net6501-70 devices, one acting as AP and the others as stations.

to interrupt driver operations that require responses with microsecond granularity to preserve accurate protocol behaviour. Subsequently, we move the computed CWs to the kernel, to be distributed to each client.

The optimisation tool relies on a simple Python script for exchanging statistics and CW configurations with the driver and uses a GNU Octave[11] programme to solve the optimisation task. Specifically, the programme orders stations according to their transmission durations, formalises the system of equations that models the utility maximisation problem and employs a standard iterative solver to compute the CW corresponding to each station. The solutions are rounded to nearest powers of 2 and passed to the driver as $\log_2 CW$ values.[12]

In what follows we demonstrate the effectiveness of our implementation by conducting extensive experiments in a real 802.11 deployment over a broad range of network conditions.

6   Experimental Evaluation

In this section we first describe the 802.11 network test bed we deploy to evaluate the proposed proportional-fair allocation scheme, and then report on the performance of the developed prototype under different traffic regimes, rate dynamics and channel conditions. We investigate a broad range of scenarios, ranging from heavily loaded networks, to set-ups whereby users run applications with strict real-time constraints, and circumstances involving lower demands, where the number of active clients and their traffic patterns vary in time. We demonstrate that our approach achieves significant gains in terms of throughput, network utility, and latency, over standard DCF operation ($CW_{min} = 16$, $CW_{max} = 1024$) in multi-rate Wi-Fi networks.

6.1   Test Bed Description

Our test bed consists of nine Soekris net6501-70 embedded PCs[13] equipped with Compex WLE300NX-6B wireless cards (Atheros AR9390 chipset) that implement the 802.11a/b/g/n specifications, and omni-

---

[11]https://www.gnu.org/software/octave/

[12]While drivers only operate with powers of 2, it would be possible to drive the *average* CW close to the exact value returned by our optimisation, if combined with e.g. sub-gradient methods [32]. We leave such refinements for future work.

[13]http://soekris.com/products/net6501.html
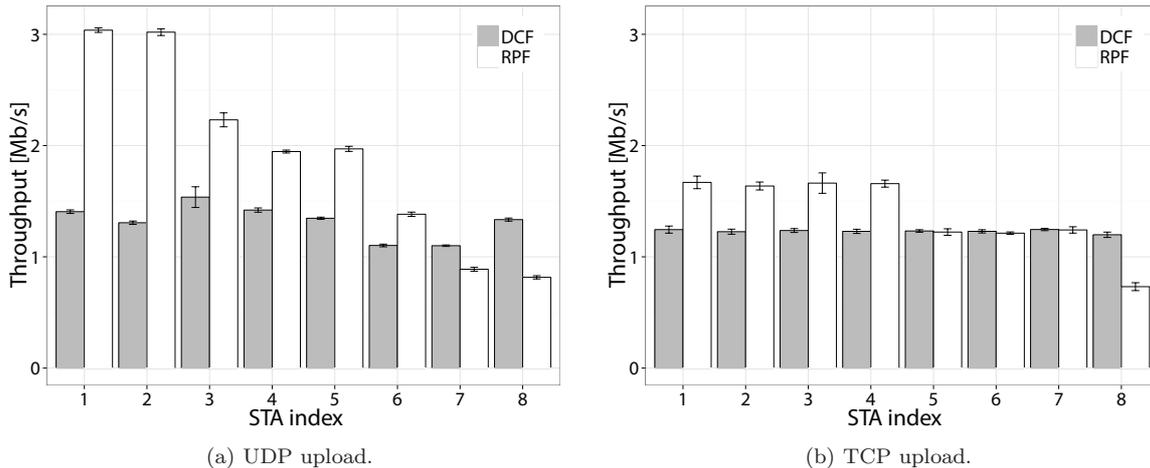
(a) UDP upload.

(b) TCP upload.

Figure 5: WLAN with 8 backlogged stations sending 1400-byte packets to the AP. Each station uses the OFDM PHY layer (802.11a) and transmits using a different MCS. Individual throughput performance with standard DCF configuration and the proposed proportional-fair (RPF) allocation scheme over UDP (left) and TCP (right). Experimental data.

directional antennas with 8dBi gain. The devices run Ubuntu 14.04 (kernel version 3.13) with `mac80211` and the `ath9k` driver. We use one of the devices as AP and the others as client stations, as shown in Fig. 4.

We deploy our network in the 5GHz frequency band on channel 149 (centre frequency 5.745GHz). No other deployments are detected in this band at the test bed's location and thus we conclude it is an interference free environment. In our experiments all stations employ the OFDM PHY layer (802.11a) over 20MHz channels, and since all clients are within carrier-sensing range of each other, we disable the RTS/CTS mechanism.

Unless otherwise stated, we run each experiment for a total duration of 60 seconds and repeat the tests 30 times to compute the average and standard deviation of the metrics of interest with good statistical significance.

6.2 Uplink Data Traffic

First we consider a scenario with 8 stations, each transmitting at a different bit rate from the set of all possible MCSs, i.e. {54, 48, 36, 24, 18, 12, 9 and 6} Mb/s. We index clients according to the magnitude of the bit rate employed (i.e. STA 1 transmits at 54Mb/s, STA 2 at 48Mb/s, etc.). All stations are backlogged with 1400-byte packets and transmit to the AP first using UDP and later TCP as transport layer protocol. In both cases we examine the individual throughput performance when the network operates with the standard DCF settings and respectively with our proportional-fair (RPF) scheme. The results of these experiments are illustrated in Fig. 5.

We observe that when all stations use the same CW configuration, the UDP throughput obtained by each is approximately the same, irrespective of the PHY bit rate used for transmission.[14] Conversely, when the AP assigns the CW configurations obtained with our approach (RPF), faster stations improve their throughput by up to 120%, while the performance of clients transmitting at inferior rates is only marginally affected (Fig. 5a). In this experiment, our proposal improves the network utility by 100%.

On the other hand it is interesting to observe that when the packets are sent over TCP, the transport layer acknowledgements in the downlink direction have collision-mitigation and pseudo-scheduling effects. As a consequence, stations experience identical performance with DCF and similar to that of UDP. The overall network utility, however, remains suboptimal, which our approach is able to correct even in this case by deriving the appropriate CW settings (Fig. 5b). Specifically, our scheme provides faster stations with 40% more throughput, while improving network utility by 40%.

---

[14]We observe small differences in the individual performance, which we attribute to the *capture effect*. Note that we investigate this phenomenon more closely in §6.6.
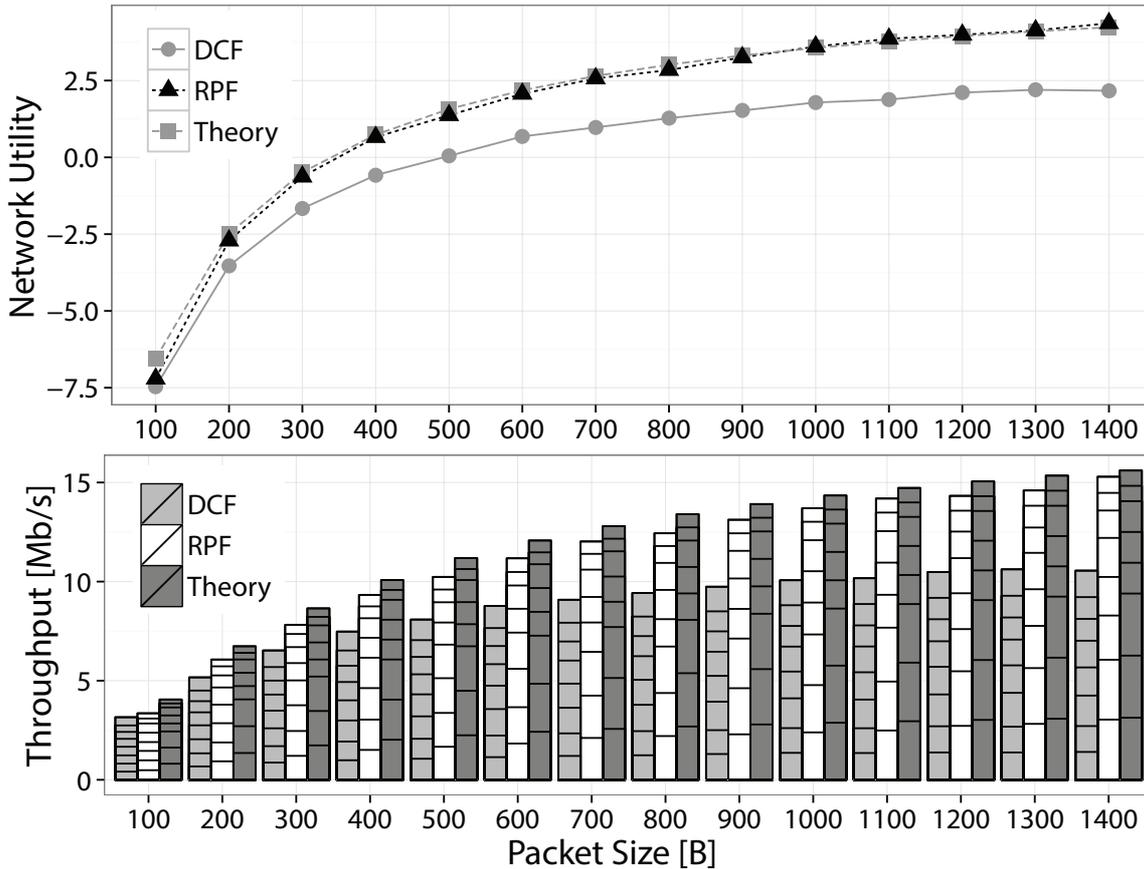
Figure 6: WLAN with 8 backlogged stations transmitting to the AP using UDP. Stations use the OFDM PHY layer (802.11a) and fully heterogeneous bit rates. Network utility (top) and individual/total throughput (below) with standard DCF configuration and the proposed proportional-fair (RPF) allocation scheme as the packet size is varied. Experimental data vs. theoretical optimum prediction.

Next we investigate the impact of frame sizes on both the individual throughput and network utility, again considering a scenario with eight stations transmitting at heterogeneous bit rates and using the UDP transport protocol. To this end, we vary the size of packets sent by each station from 100 to 1400 bytes with 100-byte increments, while examining the network performance with both the default DCF configuration and the proposed proportional-fairness enforcing solution. The results of these experiments are shown in Fig. 6, where for comparison we also include the theoretical optimum values predicted by our model.

Note again that DCF gives all stations equal transmission opportunities and thus individual throughputs are capped at that of the slowest contender. Conversely, our approach favours faster clients without starving slower ones and as a consequence improves significantly the network utility, especially as the packet size increases, in which case the utility is doubled. Furthermore, the performance of RPF in practice is very close to that predicted with our analytical model, even though this is not an idealised network.

6.3 Video Streaming

In what follows we study the performance of our prototype in the presence of real-time traffic. To this end we consider a scenario where one station experiences modest link quality when performing a TCP upload and thus transmits at a 6Mb/s bit rate (background traffic). Simultaneously, a second station is streaming video from the AP over HTTP (using the VLC media player) while operating at a 54Mb/s PHY bit rate. For this experiments we use a 10-minute fragment of the 'Big Buck Bunny' cartoon film, encoded in MP4 at 1280x720.[15]

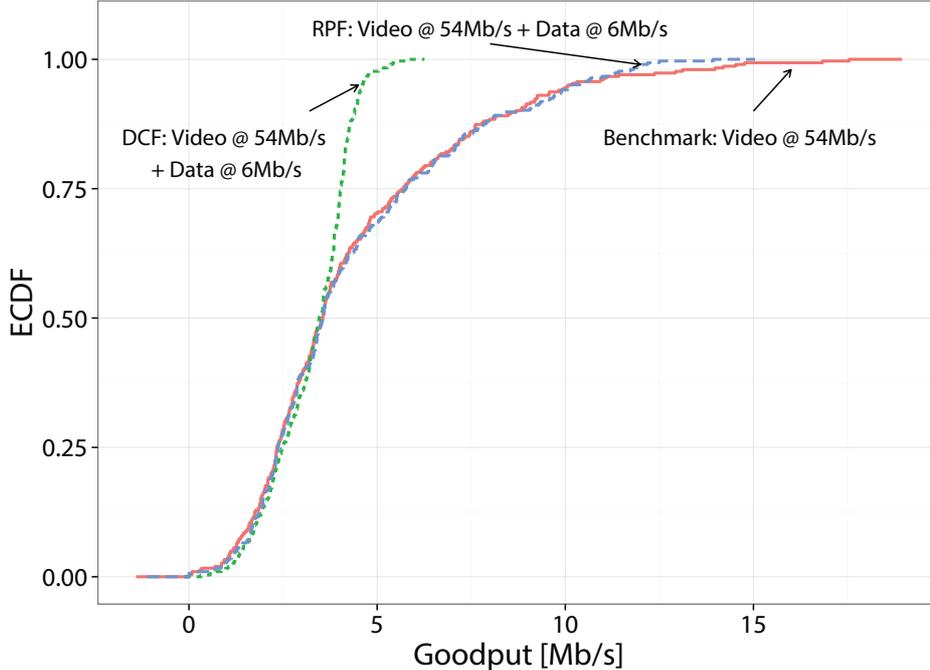[15]Available under (CC) license at `http://www.bigbuckbunny.org`

Figure 7: WLAN with 2 stations: one streams MP4 video from the AP over a wireless link operating at 54Mb/s; the other sends backlogged uplink TCP traffic at 6Mb/s PHY bit rate. Empirical CDF of the goodput attained by the video stream alone and with competing traffic, when the AP operates with legacy DCF and respectively with the proposed RPF scheme. Experimental data.

Initially we stream the video without background traffic and use the measured goodput as a benchmark for the subsequent tests. Then we evaluate the performance of the same video stream with the sluggish data station sending traffic, first with the AP advertising the default DCF configuration, and second with the AP running our RPF prototype. For all experiments, we collect samples of the stream's rate every one second and plot the empirical CDF of the attained goodput in Fig 7.

From this figure we observe that, in the presence of sluggish background traffic, the legacy DCF protocol struggles to satisfy the demand of the video stream. In contrast, when enforcing proportionally-fair allocation via our scheme, the capacity of the video link is sufficiently enhanced to ensure perfect quality of the video streaming experience.

6.4 Small File Downloads

Doubtlessly, proportional-fair allocation enhances the utility of wireless networks that exhibit multi-rate links, as we have shown so far for applications that involve transmissions of relatively large volumes of data. Many other popular applications such as *web browsing* or *e-mail*, however, exchange frequent and small information objects. For this sort of applications, the user's primary interest is in downloading these objects with minimum latency.

To evaluate the performance of our prototype with such Internet applications, in this subsection we consider a scenario whereby one station performs an upload of a large file using TCP and transmitting at 6Mb/s bit rate, while a second client downloads multiple small files over a link that operates at 54Mb/s. More specifically, we configure the AP as an HTTP server where we store a set of files with sizes that range between 64KB and 4096KB, and use the `wget` tool on the client station to retrieve these files.

We repeat this experiment 50 times for each file size and measure the average and standard deviation of the download duration when the network operates with both the default DCF configuration and respectively with our RPF implementation. The corresponding results depicted in Fig. 8 demonstrate a major reduction of the download times when using the proposed RPF scheme, with latency decreasing by 315% for the smallest object and by 790% for the largest one. Note that for such small transfers the underlying TCP congestion control algorithms used by the HTTP transfers remain in the slow-start
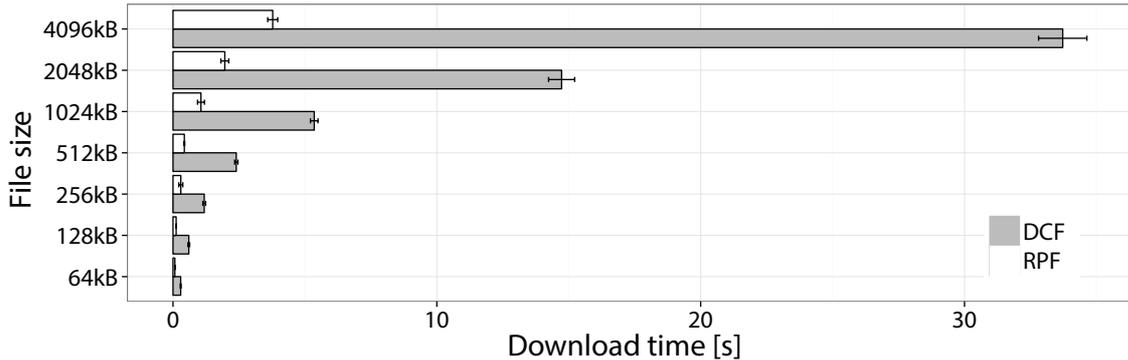
Figure 8: WLAN with 2 stations: one downloads small-sized files from the AP using HTTP over a link operating at 54Mb/s; the other is backlogged and uploads to the AP using TCP over a 6Mb/s link. Download time of different-size objects with DCF and the proposed RPF scheme. Experimental data.

phase for most of the time and, given that our RPF prototype grants faster stations a larger number of Layer 2 transmission opportunities, the round-trip time (RTT) reduction achieved translates to the high performance gains observed at the application layer.

6.5 Rate Adaptation

Current operating systems implement rate control algorithms to dynamically adapt the MCS of wireless stations to variable channel quality (e.g. Linux's `mac80211` subsystem supports both `minstrel` and a PID, i.e. proportional-integral-derivative, rate control algorithm). Thus in what follows we assess our RPF prototype's ability to track drastic changes of the PHY bit rate used by clients for their transmissions.

To this end we devise a network set up with two stations transmitting backlogged UDP traffic to the AP, initially both at 54Mb/s. While we keep the bit rate of STA 1 fixed for the entire duration of the experiment, STA 2 switches the modulation scheme between 6 and 54Mb/s every 25 seconds, and we examine the time evolution of individual throughputs for a total duration of 250s when the WLAN operates with the default DCF settings and respectively with the proposed RPF scheme. We repeat this experiment several times and present one representative snapshot in Fig. 9, while we confirm that the observed behaviour is consistent across all experiments.

Observe in Fig. 9 that our scheme rapidly detects the changes in the network and adapts the CWs accordingly. The case when STA 2 reduces its bit rate from 54 to 6Mb/s is particularly important as the network suddenly operates sub-optimally with DCF. In contrast, our proposal drives the network close to optimal behaviour whereby utility is maximised in less than 10 seconds. On the other hand when STA 2 switches from 6 to 54Mb/s, we observe a brief overshoot in STA 1's throughput due to a temporary over-allocation, but our scheme corrects this immediately and ensures both contenders experience identical performance thereafter. Note that we observe similarly quick responses to changes in the network conditions for all our experiments (which include up to 8 stations). Also shown in Fig. 9 is the theoretical optimum throughput values that stations performing (perfect) rate control are expected to achieve. This confirms that with RPF running on the AP, stations required to reduce their bit rates (due to degrading links) will receive only marginally less throughput than with DCF and close to the theoretical optimum, while the performance of the clients not experiencing changes in the link quality remains largely unaffected and nearly optimal.

6.6 Capture Effect

In real WLAN deployments the so-called *capture effect* is frequently encountered, allowing a wireless receiver to decode a frame with relatively higher signal strength even in the presence of a collision, provided the difference in the power levels of the concurrent transmissions is sufficiently large [33]. Indeed,
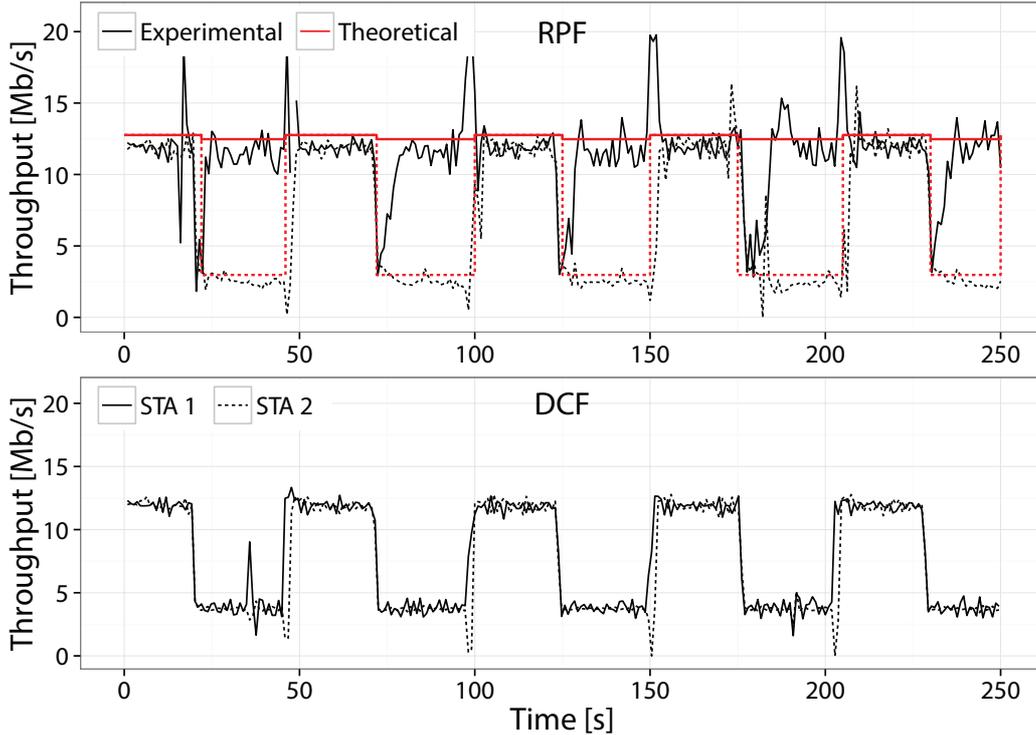
Figure 9: WLAN with 2 backlogged stations sending UDP traffic to the AP. STA 1 always operates at 54Mb/s. STA 2 alternates its bit rate between 54 and 6 Mb/s every 25 seconds. Time evolution of the individual throughputs with both DCF and the proposed RPF scheme. Experimental data vs theoretical optimum prediction.

this happens in practice even for (apparently) homogeneous set-ups in terms of AP–client distances and TX powers. This is the case in the experiment reported earlier in Fig. 5a.

While the capture phenomenon may increase the total throughput of the network [33], DCF yields an unfair distribution of the individual throughputs, as it provides all competing stations with the same CW settings, without accounting for the fact that some will not double their CW upon collisions and thus attempt more frequently. Specifically, stations delivering frames at larger signal to noise ratios (SNR) will have higher transmission attempt rates and consequently attain superior performance. In contrast to DCF, the proposed RPF allocation scheme targets equalising *airtimes* and, to achieve this, the AP considers all transmissions, including those that captured during collisions, thereby alleviating this effect.

To confirm this, we conduct another experiment whereby two competing clients transmit UDP flows in the uplink direction, both at 6Mb/s. In our experiment, the transmission power of STA 1 remains fixed at 16dBm and we vary STA 2's transmission power while we measure their individual throughputs with both DCF and the proposed RPF scheme. Precisely, STA 2 starts with the maximum supported level (30dBm), thus benefiting initially from capture with DCF, and we gradually reduce this until the SNR of both stations is similar, i.e. no capture effect is observed. Continuing this process the SNR of STA 2 falls below that of STA 1, who now benefits, as illustrated in Fig. 10 (top sub-plot).

In contrast to DCF, our RPF scheme provides both stations with the same throughput even in the presence of capture effect (Fig. 10, middle sub-plot), while this does not impact negatively on the total throughput performance of the network, as depicted in the bottom sub-plot of Fig. 10.

6.7 Comparison against Other Approaches

As discussed in §2, some of the previously proposed solutions to the performance anomaly problem have been implemented in practice, though they involve non-trivial modifications of the clients' firmware/drivers. Such modifications render a performance comparison through experiments infeasible. Therefore to provide a quantitative assessment of the advantages of the proposed RPF scheme over
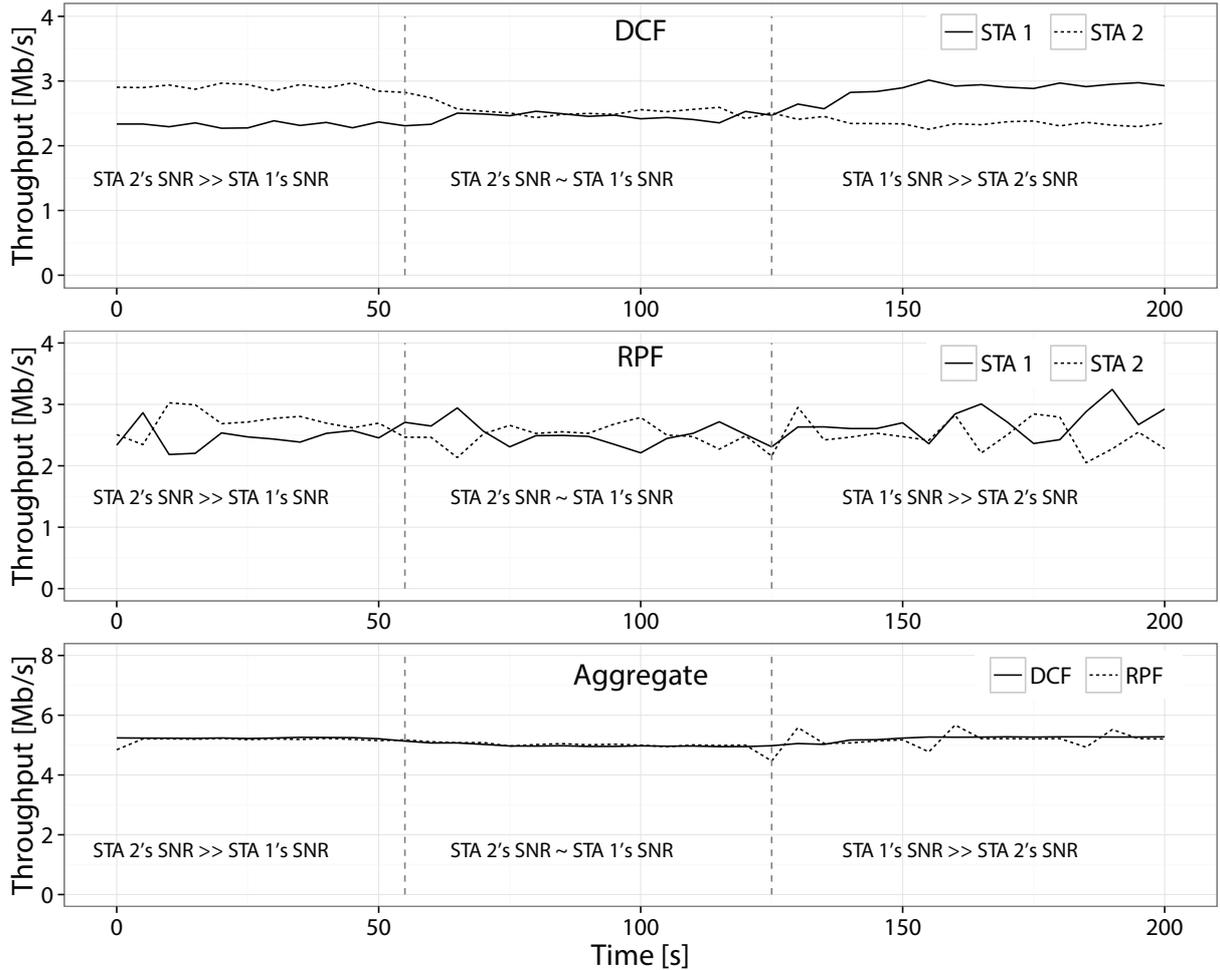
Figure 10: WLAN with 2 stations sending at 6Mb/s. STA 1 uses fixed TX power (16dBm). STA 2 starts with maximum supported power (30dBm) and reduces this gradually down to the minimum (0dBm). Time evolution of individual throughputs with DCF (top), the proposed RPF scheme (middle), and the aggregate network throughput in both scenarios (bottom). Experimental data.

earlier prototypes, in this section we compare the performance of two such approaches, Idle Sense [20] and O-DCF [22], against the standard DCF and RPF, by means of system level simulations. We note that in contrast to other schemes that target proportional fairness, we have already shown that RPF's performance is very close to the theoretical optimum, therefore an exhaustive comparison would not be justified.

We consider again the case of a wireless LAN with eight backlogged client stations, each transmitting 1000-byte UDP packets at one of the bit rates available with the OFDM PHY layer (802.11a), and measure the total throughput, network utility, and Jain's fairness index [34] achieved by each approach. In each case, we focus on three different scenarios, namely all stations supporting the proposed mechanism ("All smart"), half of them boasting the necessary modifications ("Half smart"), and respectively all being legacy 802.11 clients ("None smart"). Remember that for both standard DCF and RPF no client side modifications are required.

We run each simulation for a total duration of 5 minutes, and collect statistics after a 10-second warm up interval. We repeat each simulation 10 times and report average values and 95% confidence intervals for the metrics of interest. The results are depicted in Fig. 11, where we observe that our approach maximises network performance in all scenarios and outperforms the other approaches, especially as the number of legacy clients increases. Furthermore, RPF achieves a good compromise between maximum network throughput and fair throughput distribution, as Jain's fairness index is reasonably high.

From the above experiments we conclude that the proposed proportional-fair allocation scheme signif-
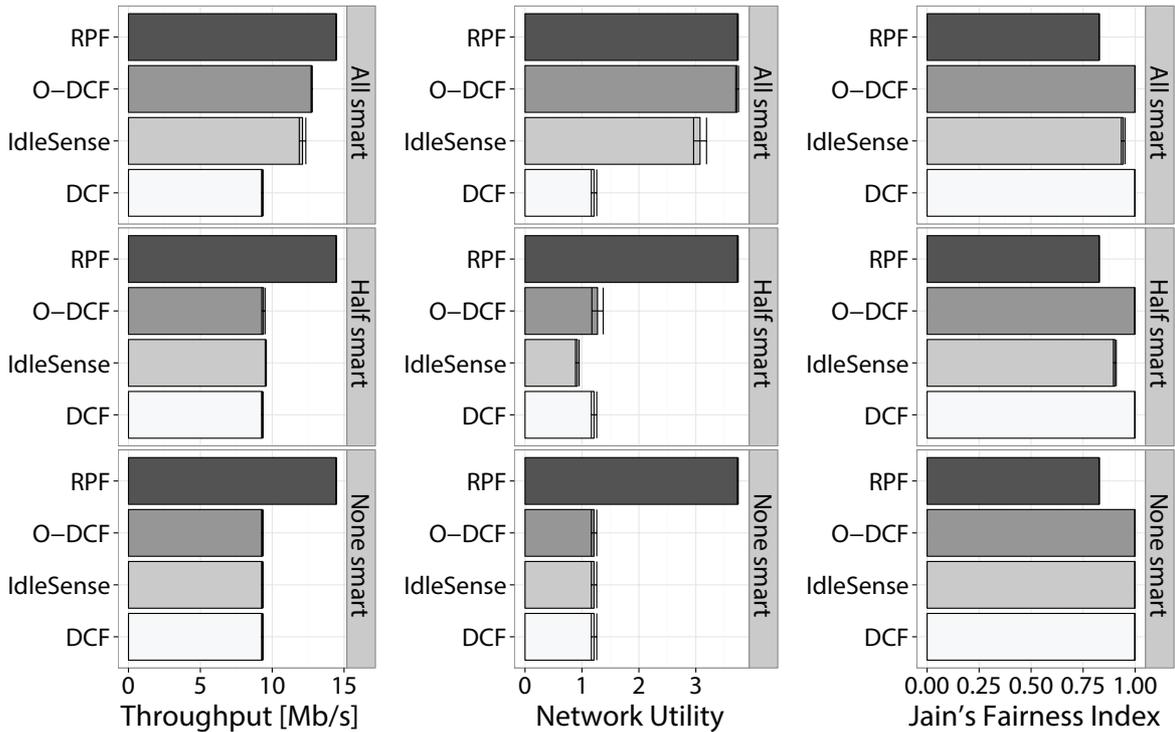
Figure 11: WLAN with 8 backlogged stations transmitting using UDP. Stations use the OFDM PHY layer (802.11a), fully heterogeneous bit rates and employ our proposal (RPF), Idle Sense, O-DCF, and standard DCF. Total throughput (left), network utility (centre), and Jain's fairness index (right) in three scenarios: all clients smart (implementing changes as required, top), half smart (middle), and respectively all legacy (none smart, bottom). Simulations results.

icantly increases network utility, improves the performance of video streams, reduces small file download times, reacts fast to rate control decisions, and alleviates the capture effect in real 802.11 networks.

7   Conclusions

In this paper we made the case for accurate modelling of multi-rate 802.11 operation, establishing a rigorous analysis that accounts for different packet sizes, channel bandwidths, PHY bit rates and frame error rates. We formulated network utility maximisation as a convex optimisation problem that we solved explicitly. We designed and implemented a practical scheme that achieves the desired proportional-fair resource allocation only with light-weight modifications to commodity APs' software and no changes to user equipment. Experimental results in a real 802.11 deployment demonstrated substantial performance gains in terms of throughput, utility, video performance, and file download times, over a broad range of channel conditions, client activity levels, and traffic regimes.

Acknowledgements

References

[1] ABI Research, Global Wi-Fi Hotspots Will Grow to 7.1 Million in 2015 as a Method to Offload Traffic (May 2014).

[2] IEEE 802.11, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Revision of IEEE Std 802.11-1999, 2007.

[3] M. Heusse, F. Rousseau, G. Berger-Sabbatel, A. Duda, Performance anomaly of 802.11b, in: Proc. IEEE INFOCOM, 2003, pp. 836–843.

[4] IEEE 802.11ac, Wireless LAN Medium Access Control (MAC) And Physical Layer (PHY) Specifications. Amendment 4: Enhancements for Very High Throughput for Operation in Bands below 6 GHz, 2013.

[5] F. Kelly, Charging and rate control for elastic traffic, European Transactions on Telecommunications 8 (1) (1997) 33–37.

[6] S. Shenker, Fundamental Design Issues for the Future Internet, IEEE Journal on Selected Aareas in Communications 13 (7) (1996) 1176–1188.

[7] Y. Le, L. Ma, W. Cheng, X. Cheng, B. Chen, Maximizing throughput when achieving time fairness in multi-rate wireless LANs, in: Proc. IEEE INFOCOM, 2012, pp. 2911–2915.

[8] Y. Le, L. Ma, W. Cheng, X. Cheng, B. Chen, A Time Fairness Based MAC Algorithm for Throughput Maximization in 802.11 Networks, IEEE Transactions on Computers 64 (1) (2015) 19–31.

[9] A. Checco, D. J. Leith, Proportional Fairness in 802.11 Wireless LANs, IEEE Comms. Lett. 15 (8) (2011) 807–809.

[10] G. Tychogiorgos, A. Gkelias, K. K. Leung, Utility-proportional fairness in wireless networks, in: Proc. IEEE PIMRC, 2012, pp. 839–844.

[11] M. Laddomada, F. Mesiti, M. Mondin, F. Daneshgaran, On the throughput performance of multirate IEEE 802.11 networks with variable-loaded stations: analysis, modeling, and a novel proportional fairness criterion, IEEE Trans. Wireless Comm. 9 (5) (2010) 1594–1607.

[12] A. Banchs, P. Serrano, H. Oliver, Proportional Fair Throughput Allocation in Multirate 802.11e EDCA Wireless LANs, Wireless Networks 13 (5) (2007) 649–662.

[13] L. B. Jiang, S. C. Liew, Proportional fairness in Wireless LANs and ad hoc networks, in: Proc. IEEE WCNC, 2005, pp. 1551–1556.

[14] L. Li, M. Pal, Y. Yang, Proportional Fairness in Multi-Rate Wireless LANs, in: Proc. IEEE INFO-COM, Phoenix, USA, 2008, pp. 1004–1012.

[15] V. A. Siris, G. Stamatakis, Optimal CWmin selection for achieving proportional fairness in multi-rate 802.11e WLANs: test-bed implementation and evaluation, in: Proc. ACM WiNTECH, 2006, pp. 41–48.

[16] A. Patro, S. Govindan, S. Banerjee, Observing Home Wireless Experience Through WiFi APs, in: Proc. ACM MobiCom, 2013, pp. 339–350.

[17] A. Gupta, J. Min, I. Rhee, WiFox: Scaling WiFi Performance for Large Audience Environments, in: Proc. ACM CoNEXT, 2012, pp. 217–228.

[18] S. Chakraborty, P. Swain, S. Nandi, Proportional Fairness in MAC Layer Channel Access of IEEE 802.11s EDCA Based Wireless Mesh Networks, Ad Hoc Networks 11 (1) (2013) 570–584.

[19] S. Singh, U. Madhow, E. Belding, Beyond Proportional Fairness: A Resource Biasing Framework for Shaping Throughput Profiles in Multihop Wireless Networks, in: Proc. IEEE INFOCOM, 2008, pp. 2396–2404.

[20] M. Heusse, F. Rousseau, R. Guillier, A. Duda, Idle sense: an optimal access method for high throughput and fairness in rate diverse wireless LANs, in: Proc. ACM SIGCOMM, Philadelphia, Pennsylvania, USA, 2005, pp. 121–132.

[21] Y. Grunenberger, M. Heusse, F. Rousseau, A. Duda, Experience with an implementation of the Idle Sense wireless access method, in: Proc. ACM CoNEXT, New York, New York, 2007, pp. 1–12.

[22] J. Lee, H. Lee, Y. Yi, S. Chong, B. Nardelli, M. Chiang, Making 802.11 DCF near-optimal: Design, implementation, and evaluation, in: Proc. IEEE SECON, 2013, pp. 86–94.

[23] M. Hegde, P. Kumar, K. R. Vasudev, N. N. Sowmya, S. V. R. Anand, A. Kumar, J. Kuri, Experiences with a Centralized Scheduling Approach for Performance Management of IEEE 802.11 Wireless LANs, IEEE/ACM Trans. Netw. 21 (2) (2013) 648–662.

[24] Y. Yiakoumis, M. Bansal, A. Covington, J. van Reijendam, S. Katti, N. McKeown, BeHop: A Testbed for Dense WiFi Networks, in: Proc. ACM WiNTECH, Maui, Hawaii, USA, 2014, pp. 1–8.

[25] G. Bianchi, Performance Analysis of the IEEE 802.11 Distributed Coordination Function, IEEE Journal on Selected Areas in Communications 18 (3) (2000) 535–547.

[26] D. Leith, V. Subramanian, K. Duffy, Log-convexity of rate region in 802.11e WLANs, IEEE Comms. Lett. 14 (1) (2010) 57–59.

[27] D. Giustiniano, D. Malone, D. Leith, K. Papagiannaki, Experimental assessment of 802.11 MAC layer channel estimators, IEEE Comms. Lett. 11 (12) (2007) 961–963.

[28] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, 2009.

[29] F. S. Hillier, G. J. Lieberman, Introduction to Operations Research, 9th Edition, McGraw-Hill, 2009.

[30] mac80211, wireless driver API for SoftMAC devices, `http://wireless.kernel.org/en/developers/Documentatio`

[31] Juniper Networks, Junos OS for M Series Multiservice Edge Routers, Release 13.3, 2014.

[32] V. Valls, D. J. Leith, Max-weight revisited: Sequences of non-convex optimisations solving convex optimisations, arXiv 0707.3168.

[33] P. Patras, H. Qi, D. Malone, Exploiting the Capture Effect to Improve WLAN Throughput, in: Proc. IEEE WoWMoM, 2012, pp. 1–9.

[34] R. Jain, Chiu, D.M., W. Hawe, A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Systems, DEC Research Report TR-301.

Appendix

Table 2: Summary of notation used throughout the undertaken performance analysis.

Symbol	Interpretation
N	Total number of stations
L_i	Length of packet transmitted by station i
C_i	Transmission rate of station i
W_i	Contention window employed by client i
τ_i	Probability a station i transmits in a randomly chosen slot
$p_{f,i}$	Conditional failure probability seen by station i
$p_{n,i}$	Link error probability experienced by station i
p_i	Collision probability experience by station i
$p_{s,i}$	Probability a slot contains a successful TX of station i
$p_{u,i}$	Probability a slot contains a successful TX of station i
P_e, P_s, P_u	Probabilities that a slot is empty,
	contains a successful, and unsuccessful transmission
$T_{s,i}$	Duration of a successful transmission of client i
T_s, T_u	Average duration of a successful and failed transmission
T_e	Duration of an empty slot (PHY layer constant)
T_{slot}	Average slot duration
T_{PLCP}	Duration of the Physical Layer
	Convergence Protocol preamble and header
H	MAC overhead (header and frame check sequence)
T_{ack}	Duration of an acknowledgement frame
SIFS, DIFS,	Short, DCF, and Extended Inter-frame Space
EIFS	(PHY layer constants)
S_i	Throughput attained by station i
T_i	Fraction of the total airtime occupied by station i