

# Averting Speed Inefficiency in Rate-Diverse WiFi Networks through Queueing and Aggregation

Martín Zubeldía, Andrés Ferragut and Fernando Paganini  
Universidad ORT Uruguay, Montevideo, Uruguay

**Abstract**—IEEE 802.11n, the latest version of the widely used standard for wireless LANs, promises significant increases in speed by incorporating multiple enhancements at the physical layer. In this paper we demonstrate that, on the contrary, the straightforward deployment of 802.11n in conjunction with TCP over a simple, single access-point network, can dramatically underachieve the promised speeds. Part of the deficiency is due to overheads and can be improved by the technique of packet aggregation present in the standard. However more subtle problems are identified, in particular the downward equalization of throughputs that occurs under physical rate diversity, or the unreasonable portion of resources taken by uplink flows when competing with the more numerous downlink connections. These difficulties are demonstrated and their causes explained through a sequence of experiments with the ns3 packet simulator. Our analysis leads us to propose a desirable resource allocation for these situations of competition, and an architecture for control in the access-point to achieve it. Our solution involves a combination of packet aggregation, multiple queues and TCP-ACK isolation, compatible with the standard and where all the control resides at the AP. We demonstrate analytically and through extensive simulation that our method is able to provide significant enhancements in performance under a variety of traffic conditions.

## I. INTRODUCTION

Wireless local area networks (WLANs) based on IEEE 802.11 [1] are present in nearly every networking deployment around the world. WLAN hotspots are shared by multiple users at a time through Medium Access Control (MAC) protocols, and newer versions of the standard have progressively upgraded the available physical channel speeds.

In the latest IEEE 802.11n version [2], many new enhancements in modulation and transmission techniques (OFDM, MIMO) have been incorporated to allow stations to transmit at rates reaching 600 Mbps. It is clear, however, that these higher data rates are only achievable in the best channel conditions, and thus stations are allowed to transmit at lower data rates if necessary to reduce frame transmission errors. The net effect of this adaptation is that multiple users with diverse data rates coexist in the same cell. This fact is not considered by the Distributed Coordination Function (DCF) for channel access, that provides equal access opportunities to all stations, regardless of their physical rate; we see below that this is a source of inefficiency. Channel access differentiation is allowed in the Enhanced Distributed Channel Access (EDCA) function of the standard, but its intended use is to differentiate traffic classes, not individual station data rates.

Research supported by ANII-Uruguay scholarship INI\_X\_2010\_2\_2986, and by AFOSR-US under grant FA9550-09-1-0504.

Another issue regarding the efficient use of the medium are protocol overheads. In particular, the coexistence of different data rates imposes the use of a Physical Layer Convergence Protocol (PLCP) to provide synchronization and indicate the data rate of the forthcoming frame. This header, which must be sent at the basic (lowest) data rate, can occupy a significant amount of time in comparison to the data frame at a high physical rate. To mitigate this, the standard has included the use of *packet aggregation*, in which a single channel access by a station is used to transmit multiple higher layer packets, whether in a single frame (A-MSDU) or in multiple contiguous frames (A-MPDU). The use of frame aggregation is well known to enable almost 100% channel utilizations in point to point communications. However, the use of frame aggregation in rate diverse environments, as well as the implications it has on higher layer protocols has received far less attention.

In this paper, we study the performance obtained by TCP connections when packet aggregation is used at the MAC level and stations show rate diversity. We identify in Section II various reasons why the packet aggregation mechanisms alone may fail to deliver the promised speeds: lack of proper attention to the bidirectional nature of TCP; inefficient allocation of transmission opportunities between rate-diverse stations sharing a common queue; destructive competition between uplink and downlink flows. Our packet simulations exhibit some striking inefficiencies in the use of the wireless medium.

Section III describes our proposal to overcome these limitations. We first argue for what we believe is the proper assignment for rate diverse cells, a proportionally fair allocation studied in our previous work [9]. Then we proceed to describe an architecture that combines queueing and packet aggregation algorithms to achieve this allocation in 802.11n. The proposed architecture is implemented at the access point, relying only on locally available information, and does not require substantial modifications in the stations. The method is initially developed for the downlink case, but later extended to mixed downlink-uplink traffic scenarios, still based on control at the AP. We validate its performance through packet-level simulations with TCP connections, initially taken to be permanent.

In Section IV we consider the more realistic traffic scenario of a varying number of connections under a stochastic model for traffic demand. We show that the proposed algorithm enables a flow level throughput allocation that is both efficient and robust to different job size statistics. Finally, conclusions and lines of future work are given in Section V.

Parameter	Value
Slot time	9 $\mu$ s
SIFS	16 $\mu$ s
DIFS	43 $\mu$ s
PLCP Header	32 $\mu$ s
PHY rates	{6.5, 13, 19.5, 26, 39, 52, 58.5, 65} Mbps
$CW_{min}$	15
$CW_{max}$	1023

Table I  
IEEE 802.11N PARAMETERS AND PHYSICAL LAYER RATES.

### Related work

Several works have analyzed the impact of frame aggregation on 802.11 throughput. In [15] the impact of aggregation is first discussed, with uncontrolled traffic sources. Also [14], [18] discuss the case of 802.11n. In [11], the impact of frame aggregation on TCP throughput is discussed in the context of wireless mesh networks. The bandwidth allocation achieved by TCP under rate-diverse wireless LAN environments is analyzed in [8]. Also [10] discuss uplink vs. downlink unfairness issues and its relationship with TCP. In [4], [12], [21] some queueing and access control algorithms are proposed to deal with the unfairness issues. The flow level performance of wireless cells is thoroughly analyzed in [5].

## II. INEFFICIENCIES IN 802.11N CELLS WITH TCP AND PACKET AGGREGATION

In this Section we explore by simulation the effects of packet aggregation on the throughput of TCP flows in several scenarios. All simulations were performed in the network simulator ns3 [17], which we modified to include the 802.11n physical layer rates shown in Table I, as well as all the other time parameters included in the standard. We focus here mainly on non-MIMO channels, due to simulator limitations; nevertheless as we shall see, the main conclusions of these experiments do not depend on the physical rates involved. Simulations involve a single cell consisting of an Access Point (AP) and one or several client stations (STAs).

### A. Aggregating TCP frames in the AP

Consider first a single transmission in the downlink sense. A traffic source is directly connected to the AP and transmits over the wireless link to the STA. The source generates packets of standard length  $L = 1500$  bytes, that are aggregated in the AP using A-MPDU, which enables frames of size up to 64KB. Taking into account the protocol overheads, the effective transmission time when aggregating  $n$  packets in a single frame can be calculated as:

$$T_n = DIFS + Backoff + H + \frac{nL}{PHY} + SIFS + H + \frac{L_{mac}}{PHY},$$

where  $H$  is the time to transmit the PLCP header,  $L$  is the packet size,  $PHY$  is the modulation rate and  $L_{mac}$  is the MAC layer ACK length. By averaging the backoff time and assuming no collisions, the throughput is given by:

$$Thr = \frac{nL}{T_n + \frac{CW_{min}}{2} T_{slot}} = \frac{nL}{\frac{nL}{PHY} + const}.$$

A-MPDU limit	UDP Throughput (Mbps)	TCP Throughput (Mbps)
1500 Bytes	29.77	19.51
4000 Bytes	40.57	22.02
8000 Bytes	52.03	22.81
16000 Bytes	57.37	22.74
32000 Bytes	60.47	22.47
64000 Bytes	62.25	22.48

Table II  
EFFECT OF THE AGGREGATION ON UDP AND TCP THROUGHPUT AT  $PHY = 65$  MBPS.

As  $n$  increases in the above formula we see that throughput improves, reducing the impact of the fixed time overheads; it should eventually approach the value of the  $PHY$  rate. Let us test this fact by simulation with two different traffic sources: first we use an uncontrolled UDP traffic for reference. Then, the same setting is simulated with a single TCP connection. The  $PHY$  rate of the station is fixed at 65 Mbps. Results are shown in Table II.

We conclude from the results that indeed aggregation has an impact on the UDP throughput, as predicted. However, the TCP flow in the same situation benefits far less from packet aggregation, under-utilizing the channel by a factor of 3.

One reason to expect a slower performance from TCP is that such connections involve not only the packet transmission but also the TCP ACK transmission in the uplink sense. While TCP ACKs are designed to be small (40 bytes), the time overheads of the wireless layer cannot be disregarded, and since at least one TCP ACK packet must be sent for every downlink frame<sup>1</sup>, the net throughput is lowered. This was already analyzed in [9] for the 802.11g standard where no aggregation is performed.

In this scenario where aggregation is performed *only* at the AP, there is a second factor with more impact. For every aggregate frame sent, the receiver generates  $n$  TCP ACKs; since the STA performs no aggregation, sustaining a steady flow of aggregate frames would require  $n$  channel accesses of the STA for every AP access. But the DCF mechanism gives the AP and the STA equal channel access opportunities. The net result is that the AP queue empties, while the TCP flow waits for the uplink ACKs to generate replacement packets. In Figure 1 we plot the AP queue for an A-MPDU limit of 64K. Note that most of the time only one or two packets can be aggregated, so the maximum aggregation is not achieved, with the throughput saturating around 22 Mbps.

This problem can be solved by enabling aggregation on the STA, such that the TCP ACKs also get bundled in a single MAC frame, and thus require only one channel access to be transmitted. Once aggregation is enabled in the STA, the throughputs increase, as shown in Table III. From this experiment, we conclude that *aggregation must be implemented in both directions to give real benefits*.

<sup>1</sup>In our simulations, one TCP ACK is generated for each packet, to simplify the analysis of TCP effects. Typical TCP implementations also send one ACK every two packets, and our results can be adapted to that situation.

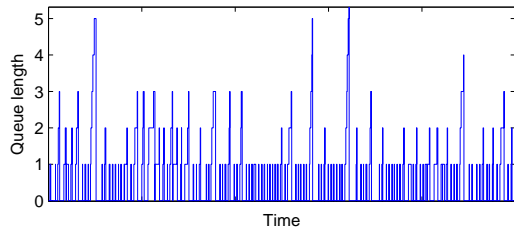


Figure 1. AP queue for a single TCP flow with aggregation

A-MPDU limit	TCP with ACK aggregation (Mbps)
1500 Bytes	19,51
4000 Bytes	30,99
8000 Bytes	42,58
16000 Bytes	49,71
32000 Bytes	54,50
64000 Bytes	57,01

Table III  
TCP THROUGHPUT WITH TCP ACK AGGREGATION IN THE STA

### B. Rate-diverse cells and differentiated aggregation

A typical effect already observed (c.f. [9], [16]) in WiFi cells is that slow stations slow down the whole network. This is a consequence of having different transmission times for the same packet lengths, with faster stations having to wait for slow stations to finish transmission before sending another packet. It would seem that packet aggregation provides a tool to address this issue: by enabling differentiated packet aggregation in proportion to the physical rates, one could equalize the packet transmission times leading to a more efficient use of the medium. Packet simulations tell, however, a more complicated story involving multiple protocol layers.

Consider a scenario of differentiated aggregation with two stations, operating at *PHY* rates 65 and 6.5 Mbps respectively. We modify the queueing algorithm in the AP queue to aggregate packets *only for the fastest station*; this means in particular that the FIFO discipline of the queue must be modified, when a packet of the fast station reaches the head-of-line, it enables the transmission of other  $n - 1$  packets that get to “jump the queue” for aggregation.

In the simulation, each STA establishes a single downlink TCP connection, and TCP ACKs are aggregated as needed in the STAs so they do not become a bottleneck, as discussed before. Results are shown in Table IV. We observe that the fast STA suffers greatly from the presence of the slow one, going from a throughput of 57 Mbps when alone to one of less than 7 Mbps here; but more importantly, aggregation had a very modest influence in correcting this outcome.

An explanation can be found in the fact that TCP connections are controlled through packet losses that occur in arrivals to a *single*, common AP queue. Packets of slow and fast flows see the same loss probability when arriving at this queue, and therefore TCP congestion control will roughly equalize the mean congestion windows of both flows [19]. Since rate equals window/round-trip-time, the only chance at throughput differentiation would come from RTT differentiation; some of that is observed, and is consistent with the advantage of

A-MPDU limit (fast STA)	Thr (65Mbps)	Thr (6.5Mbps)
1500 Bytes	4.26	3.68
4000 Bytes	5.06	4.01
8000 Bytes	5.75	4.16
16000 Bytes	6.15	4.20
32000 Bytes	6.62	4.18
64000 Bytes	6.97	4.19

Table IV  
BANDWIDTH SHARING WITH DIFFERENTIATED AGGREGATION.

A-MPDU limit (fast STA)	Agg. factor	Thr. (65Mbps)	Thr. (6,5Mbps)	Rel. Thr.
1500 Bytes	1	4.43	3.39	1.31
4000 Bytes	2.6	8.17	3.55	2.30
8000 Bytes	5.3	16.49	2.95	5.59
16000 Bytes	10.6	25.77	2.27	11.35
32000 Bytes	21.2	34.86	1.60	21.78
64000 Bytes	44.4	41.33	1.11	37.23

Table V  
BANDWIDTH SHARING WITH DIFFERENTIATED AGGREGATION AND SEPARATED QUEUES.

“jumping the queue”, but by no means this can achieve the desired level of throughput differentiation.

The main conclusion of this experiment is that *aggregation alone cannot differentiate throughputs in a rate diverse environment*, due to the closed loop behavior of the TCP protocol.

The situation is totally different if we give each flow a different queue in the AP. The EDCA mechanism in the standard enables us to implement this, although here we establish no class priorities, both queues are given equal channel access opportunities. Aggregation is again only performed in the fast station. Results are shown in Table V. The aggregation factor (ratio between the A-MPDU limit and the base packet length of 1500 bytes) now has a significant impact on the rate allocation: indeed, there is a roughly linear relationship between the relative throughput (between both flows) and the aggregation factor. Clearly, in this last scenario we have found a suitable “knob” to affect the resource allocation; the fair way to use it is discussed in the Section III.

### C. Competing uplink traffic

A third major issue in 802.11 cells is the resource allocation between downlink and uplink traffic. In typical Internet access settings, most of the traffic is downlink and the AP is serving client stations. However, once uplink traffic is present, the downlink traffic can be severely affected.

We illustrate this effect in the following simulation example, where *PHY* rates are now homogeneous at 65 Mbps. Initially we have 3 downlink stations, and later on a fourth station opens an uplink connection. No aggregation is used. Results are shown in Figure 2. When the downlink stations are sharing the medium throughputs are equalized, each connection getting approximately 7 Mbps, a third of the throughput they would get alone using TCP without aggregation (see Table II). Once the uplink connection starts, the allocation changes, with the downlink connections getting approximately 3.5 Mbps each, while the uplink obtains around 11 Mbps. This is clearly an inconvenient result, and matters can be worse if we add a rate-diverse environment.

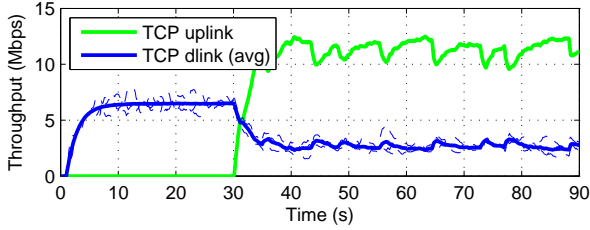


Figure 2. Throughputs for 3 downlink connections and a competing uplink, all stations at 65 Mbps.

One way to interpret the outcome is to note that there are two queues sending data packets, the AP and the uplink STA, and the DCF does not discriminate between them in the channel access opportunities. Hence the roughly 50-50 split between the uplink and total downlink throughput<sup>2</sup>, which is not sensitive to the different number of *flows* served by each of the queues. Similar issues were already noted in [8], [10].

We conclude that for the case of multiple queues accessing a medium, *access opportunities should be related to the number of flows the queue is handling*.

### III. RATE BASED QUEUEING AND AGGREGATION

From the discussion in Section II, it should be clear that aggregation and differentiated queueing can have an impact in the resource allocation achieved by TCP flows in an 802.11n environment, when performed correctly. Moreover, access opportunities should take into account the number of flows a given node is offering to the network, whether in the downlink or uplink sense. This is particularly important to protect the AP from having less transmission opportunities when handling multiple downlink flows. The purpose of this section is to devise and test a queueing and aggregation algorithm that the AP can perform in order to find a proper resource allocation.

#### A. The target allocation

Consider several stations that want to communicate over the wireless link, say in the downlink sense although this is not a restriction. Assume moreover that, when transmitting alone, station  $i$  can achieve a throughput  $C_i$ , with the overheads taken into account. When two or more STAs compete for the medium, it is reasonable to allocate the rates  $x_i$  such that:

$$\frac{x_i}{x_j} = \frac{C_i}{C_j}. \quad (1)$$

This way, STAs which are more effective in using the medium are rewarded with higher rates. Alternatively, the *time-proportions*  $x_i/C_i$  allocated for each STA are equalized by (1): transmission time is equally shared between all stations.

This notion of fairness can also be related to the theory of Network Utility Maximization, where it coincides with the familiar notion of *proportional fairness* introduced by [13] for wired networks and in [20] in the wireless case. In this formulation, rates are chosen to solve:

<sup>2</sup>This explanation oversimplifies matters since ACK traffic is not considered, but captures nevertheless the essence of the problem.

*Problem 1:* Maximize  $\sum_{i=1}^N \log(x_i)$ , subject to

$$\sum_{i=1}^N \frac{x_i}{C_i} \leq 1. \quad (2)$$

This differs from the standard case of [13] by the capacity constraint: in a rate-diverse situation, (2) states that the sum of time proportions in the medium can be no larger than unity. For completeness, we briefly derive<sup>3</sup> the solution of Problem 1, by introducing the Lagrangian

$$\mathcal{L}(x, p) = \sum_{i=1}^N \log(x_i) + p \left( \sum_{i=1}^N \frac{x_i}{C_i} - 1 \right).$$

Here  $p \geq 0$  is the Lagrange multiplier associated with constraint (2), and the Karush-Kuhn-Tucker (KKT) conditions for optimality imply that

$$\frac{\partial \mathcal{L}}{\partial x_i} = \frac{1}{x_i} - \frac{p}{C_i} = 0. \quad (3)$$

From (3), we deduce that  $x_i/C_i = 1/p$  for all  $i$ , verifying the equality of time-proportions mentioned before. Using the constraint (2) yields

$$x_i^* = \frac{C_i}{N}, \quad (4)$$

$N$  being the total number of flows. In particular, rates are allocated proportionally to the effective capacities  $C_i$ .

*Remark 1:* The allocation defined by (4) verifies the following attractive property: whenever a given flow in a cell changes its radio conditions, the allocated rate changes only for that flow. This is especially important in rate-diverse environments such as 802.11n cells. If several flows are transmitting at the maximum possible rate, and one of them changes to a lower rate, in a typical 802.11n cell this will downgrade the rates of all flows. If (4) is used, faster flows are protected and as a result, the throughput of the cell will be significantly higher.

#### B. Implementation: Downlink traffic

The implementation question is how to drive the system to allocation (4) using the 802.11n capabilities. In order to achieve (4), we should:

- Give each flow equal channel access opportunities.
- Allow each flow to transmit during the same amount of time during a channel access.

This could in principle be implemented by putting each flow in a separate queue, with equal access opportunities (i.e. each flow has a single EDCA queue with the same AIFS and backoff parameters), and use rate-based aggregation such that physical layer frames of different stations with rates last the same time.

The first part of the solution is not practical due to the potentially large, and variable number of flows. We propose instead the following Rate Based Queueing and Aggregation architecture (RBQA), which consists of three ingredients; we describe it first in the case of downlink flows.

<sup>3</sup>More details are found in [9].

PHY (Mbps)	A-MPDU limit (bytes)	$C_i$ (Mbps)
6.5	1500	4.87
13	3000	9.82
19.5	4500	14.8
26	6000	19.7
39	9000	29.6
52	12000	39.5
58.5	13500	44.4
65	15000	49.4

Table VI

PHY RATES, AGGREGATION AND MAXIMUM EFFECTIVE TCP RATES.

1) *Queues*: The AP maintains one queue for each *PHY* data rate. This is implemented using the EDCA algorithm, but in principle each queue has the same AIFS parameter, and therefore, transmission opportunities. We use the MAC destination address to determine the current *PHY* rate and put the packet in the corresponding queue.

2) *Aggregation*: To achieve the desired time-fairness, each queue implements A-MPDU aggregation. The slowest *PHY* has an aggregation limit of 1500 bytes, which amounts to 1 packet when data transfers are in place. As the *PHY* rate increases, the A-MPDU limit increases in proportion, reaching 15000 bytes for 65 Mbps. Since the fixed overheads are the same for all rates, this amounts to equalizing channel usage times. In Table VI we summarize the aggregation parameters of each data rate, and the corresponding effective rates a single flow would get when alone in the cell (considering all MAC layer and TCP ACK overheads), which correspond to the  $C_i$  of equation (4).

Of course, higher  $C_i$ 's for all classes could be achieved by scaling all aggregation factors by a common number; we have refrained, however, from using aggregations beyond 10 packets to keep our buffering requirements in check.

3) *Channel access for multiple flows*: To give flows equal access opportunities without having to resort to per-flow queues, the proposal is to control the aggressiveness of channel access of each AP queue  $j$  in proportion to the number of connections  $n_j$  present in it. We assume for simplicity that each STA has a single connection, and thus we can identify  $n_j$  with the number of MAC addresses present in the queue<sup>4</sup>, something that can be tracked by the AP.

We wish to regulate the frequency  $\tau_j$  of channel accesses of queue  $j$ , in proportion to  $n_j$ . For this purpose, we choose to adapt the minimum contention window parameter  $CW_{min_j}$  associated with each queue, which is related to  $\tau_j$  through

$$\frac{\tau_j}{\tau_k} = \left( \frac{1 - 2\gamma_j}{1 - 2\gamma_k} \right) \left( \frac{CW_{min_k}}{CW_{min_j}} \right).$$

Here  $\{\gamma_i\}$  are the collision probabilities seen by each queue; the above can be established using the analysis of [16] for the backoff process, details are omitted. For small collision probabilities, we see that  $\tau_j$  is inversely proportional to  $CW_{min_j}$ . The backoff adaptation algorithm is thus defined by

$$CW_{min_j} = CW_0 \frac{n_{max}}{n_j}, \quad (5)$$

<sup>4</sup>With this approach our fairness model is established between STAs rather than TCP flows, a valid alternative.

PHY (Mbps)	Per-flow thr. (Mbps) No A-MPDU	Per-flow thr. (Mbps) A-MPDU limit 64K
65	0.90	1.48
39	0.89	1.29
19.5	0.91	1.35
6.5	0.88	1.22

Table VII

PER-FLOW THROUGHPUT IN A RATE DIVERSE ENVIRONMENT WITH STANDARD 802.11N AND MAXIMAL A-MPDU AGGREGATION.

PHY (Mbps)	$C_i$ (Mbps)	Prop. fair alloc. $C_i/N$ (Mbps)	Measured per-flow throughput (Mbps)
65	48.9	4.45	4.02
39	29.8	2.71	2.23
19.5	14.7	1.34	1.35
6.5	4.94	0.45	0.40

Table VIII

PER-FLOW THROUGHPUT IN RATE-DIVERSE ENVIRONMENT WITH RBQA.

where  $n_{max} = \max_j n_j$  and  $CW_0$  is a base contention window setting, used by the queue with most connections, which we set to  $CW_0 = 16$  slots. The remaining queues have less aggressive backoff processes. By using (5), we ensure that channel access frequencies  $\tau_j$  are set proportionally to the number of stations with flows traversing queue  $j$ . Thus we approximate per-flow queueing with an architecture that keeps the number of queues to a minimum and fixed in time, thereby simplifying implementation.

### C. Simulation results: Downlink

The complete set of algorithms was implemented at the MAC layer of the AP in ns3. To test its performance, we simulated a rate-diverse scenario consisting of 3 STAs connected at a *PHY* rate of 65 Mbps, 1 at 39 Mbps, 5 at 19.5 Mbps and 2 at 6.5 Mbps. In Table VII we present the per-flow throughputs achieved when standard 802.11n is deployed, and when full size A-MPDU aggregation is in use. In the first case throughputs are equalized across all classes. In the second case, aggregation provides a slight differentiation.

In Table VIII we provide the results in the same scenario, when the RBQA algorithm is in place. For comparison purposes, we also include the corresponding effective rates  $C_i$  and the proportional fair desired allocation  $x_i^* = C_i/N$ . The measured throughputs are now clearly different between classes, approximating the desired proportional fair allocation. Moreover, the total cell throughput is 9.9 Mbps in the standard case, and 14.9 Mbps with maximal aggregation. Instead, the RBQA algorithm achieves a total throughput 21.8 Mbps, a 120% increase in efficiency with respect to standard 802.11n. We note that examples could be given where the increase in efficiency is even more dramatic; the above scenario was chosen to exhibit what we found to be a representative case.

### D. Uplink traffic throttling and global solution

Up to now we have considered downlink traffic, and through queueing and aggregation, we improved on the resource allocation in an 802.11n cell. However, as we already discussed in Section II-C, if STAs open uplink connections, the channel access algorithm will give them an important share of the

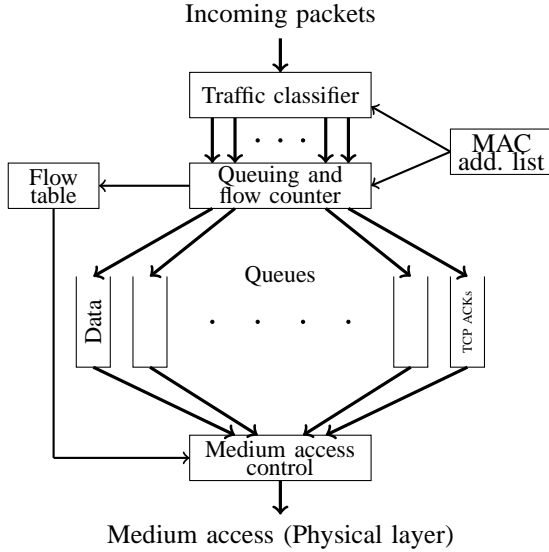


Figure 3. RBQA architecture for the AP.

resources. We would like to enhance our algorithm in order to throttle the uplink sources from the AP side, without modifying the STAs, which typically cannot be controlled directly.

Our approach here, already considered in [10] for the single rate case, is to use the TCP feedback behavior in order to force the STAs to regulate themselves by controlling the number of TCP ACK packets going in the downlink sense. This implies using separate queues for TCP ACK packets at the AP, in our case as many as the available  $PHY$  rates; access probabilities for the ACK queues should also be made proportional to the number of flows, i.e. we use (5) to set their contention window.

In a multi-rate environment, the remaining question is what aggregation to use in the ACK queues to reach the proportional fair allocation (4) between all flows (uplink or downlink) in the cell. The answer is that TCP ACKs should use the aggregation factor corresponding to their  $PHY$  rate, as if they were data packets. For instance, to regulate 65 Mbps sources, we aggregate up to 10 TCP ACKs on each transmission. Note that this is different from aggregating to 15000 bytes.

The effect of the proposed aggregation is the following: the transmission rate in ACKs/sec from the AP back to the source STA will be equal to the packets/sec allocated to a downlink flow of the same  $PHY$  rate. Since the TCP source throttles its transmission to this ACK stream, its uplink rate in data packets/sec, and hence in Mbps, will equalize to that of downlink flow of the same  $PHY$  rate, as desired.

The complete RBQA architecture for the 802.11n AP is shown in Figure 3. It can be implemented in the AP resorting only to local information already at its disposal. The only necessary modification to the STAs is to enable aggregation with a high A-MPDU limit.

#### E. Simulation results: Uplink and Downlink

To test the performance of our proposed algorithm, we revisit the uplink example of Section II-C. Three downlink

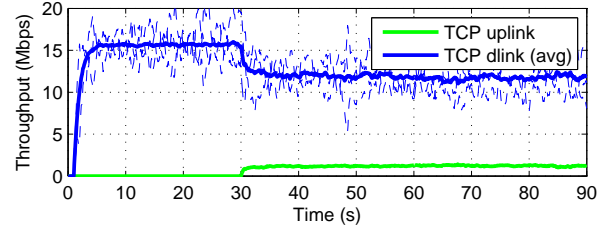


Figure 4. Results for a mixed downlink-uplink scenario with RBQA.

TCP connections at  $PHY = 65$  Mbps are established and some time later, an uplink connection enters, this time we assign it a  $PHY$  rate of 6.5 Mbps to test rate diversity. Results are shown in Figure 4. Note that, due to the use of aggregation, downlink performance is improved. Most importantly, once the uplink connection is started, it is throttled so the downlink connections are not unduly penalized. The resulting rates are approximately 12.5 Mbps for the faster flows, and 1.2 Mbps for the slow uplink flow, which coincides with the desired proportional fair allocation (4).

#### IV. FLOW LEVEL PERFORMANCE

In Section III, we proposed the RBQA algorithm to enforce a proportional fair allocation of rates to permanent TCP connections. We now analyze the behavior of our proposal in a more realistic traffic environment, with a time varying number of ongoing flows. A frequently used model [6] for this setting is to consider that new TCP connections of class  $i$ , associated with  $PHY$  rate  $PHY_i$ , and effective rate  $C_i$ , arrive as a Poisson process of intensity  $\lambda_i$ . Each connection brings a random amount of workload, which are independently and identically distributed with mean  $1/\mu$ . Connections are allocated an instantaneous service rate given by (4).

When job sizes are exponentially distributed, this type of model was studied in [9], where it is shown that the vector valued process  $n(t) = (n_i(t))$  recording the number of ongoing connections in each class constitutes a continuous time Markov chain with transition rates:

$$q_{n,n+e_i} = \lambda_i \quad q_{n,n-e_i} = \mu C_i \frac{n_i}{\sum_j n_j}, \quad (6)$$

where  $e_i$  is a vector with 1 in coordinate  $i$  and zeros elsewhere.

The Markov chain defined by (6) is a particular case of a *Discriminatory Processor Sharing* queue [3], with equal weights for all classes. In particular, if we define the load of the system by  $\rho = \sum_i \frac{\lambda_i}{\mu C_i}$ , then the flow-level queue is stable only if  $\rho < 1$ , i.e. the time proportions needed to serve all flows on average are less than unity. This particular case can also be solved explicitly, with the average number of flows in equilibrium on class  $i$  satisfying:

$$E[n_i] = \frac{\rho_i}{1 - \rho},$$

where  $\rho_i = \frac{\lambda_i}{\mu C_i}$  is the load of class  $i$ . The throughput perceived by a typical connection of class  $i$  can also be estimated as:

$$Thr_i = C_i(1 - \rho). \quad (7)$$

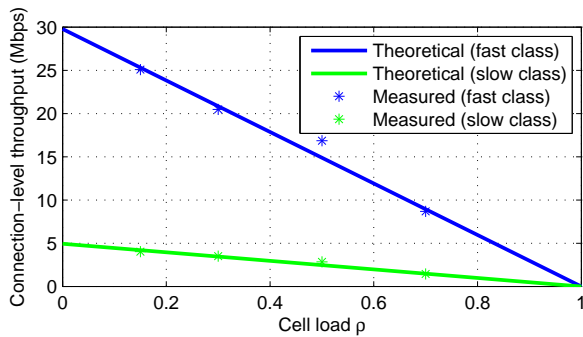


Figure 5. Flow level throughputs for a cell with Poisson traffic.

Job size distr.	$Thr_{39Mb}$ (Mbps)	$Thr_{6.5Mb}$ (Mbps)
Exponential	8,68	1,44
Pareto	9,30	1,57
Deterministic	9,23	1,60
Theoretical:	8,93	1,48

Table IX

FLOW LEVEL THROUGHPUT FOR TWO PHY CLASSES WITH POISSON ARRIVALS AND DIFFERENT JOB SIZE DISTRIBUTION.

In this context,  $(1 - \rho)$  is called the *slowdown* of the processor sharing queue. This throughput is decreasing with the cell load.

We conclude that the system provides a flow-level throughput proportional to the effective rates  $C_i$ , and only coupled with the remaining classes through the total cell load, which is a desirable result. A second remark is that, due to the insensitivity properties of reversible PS networks [7], equation (7) still holds for general job size distribution with mean  $1/\mu$ . Therefore, the flow-level throughputs achieved by the system do not depend on how job sizes are drawn.

To validate the above model, we simulated a single cell with downlink traffic, using the RBQA algorithm in the AP. Connections are equally split between two classes of *PHY* rates 39 and 6.5 Mbps respectively. Job sizes are exponentially distributed with average  $3MB$ , and the arrival rates are varied such that the load goes from 0 to 1. In Figure 5 we plot the measured connection-level throughputs as well as those predicted by (7). We can see that the system indeed keeps a connection-level throughput differentiation across all loads, showing good fit against the theoretical predictions. To show the robustness of our proposal, we also simulated the system with different job sizes, in particular Pareto (heavy tailed) and deterministic distributions, for a fixed value of the load. Results are shown in Table IX, which shows the predicted insensitivity of the allocation.

## V. CONCLUSIONS AND FUTURE WORK

In this paper, we analyzed several performance issues related to Wireless Local Area Networks based on the IEEE 802.11n standard. We showed that packet aggregation and medium access must take into account cross layer issues regarding the upper layer protocols to be effective. We proposed a Rate Based Queueing and Aggregation architecture that can be implemented in the Access Point, and that ensures that all data flows receive a proportionally fair share of bandwidth

allocation and improves the total throughput of the cell. This algorithm relies only on locally available information and is also able to throttle the uplink flows. Moreover, we provided a packet-level implementation and simulations that validate its behavior in several settings.

In future work, we plan to analyze how to improve our algorithm to take into account other classes of traffic, such as real time or streaming, which should themselves be protected from data transfers, as well as implementing the architecture in a real network deployment.

## REFERENCES

- [1] "IEEE 802.11-2007, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," 2007.
- [2] "IEEE 802.11n-2009, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," 2009.
- [3] E. Altman, K. Avrachenkov, and U. Ayesta, "A survey on discriminatory processor sharing," *Queueing Systems: Theory and Applications*, vol. 53, no. 1-2, pp. 53–63, Jun. 2006.
- [4] G. Banaghe, D. Raychaudhuri, and I. Seskar, "Backlogged queue based MAC frame aggregation," *Pervasive and Mobile Computing*, vol. 7, no. 4, pp. 449–466, 2011.
- [5] T. Bonald, S. Borst, N. Hegde, M. Jonckheere, and A. Proutière, "Flow-level performance and capacity of wireless networks with user mobility," *Queueing Systems*, vol. 63, no. 1–4, pp. 131–164, 2009.
- [6] T. Bonald and A. Proutière, "Flow-level stability of utility-based allocation in non-convex rate regions," in *Proc. of the 40th Annual Conference on Information Systems and Sciences (CISS 2006)*, 2006, pp. 327–332.
- [7] T. Bonald and A. Proutière, "Insensitivity in processor-sharing networks," *Performance Evaluation*, vol. 49, no. 1–4, pp. 193–209, 2002.
- [8] J. Choi, K. Park, and C. Kwon Kim, "Cross-layer analysis of rate adaptation, DCF and TCP in multi-rate WLANs," in *IEEE Infocom*, May 2007.
- [9] A. Ferragut and F. Paganini, "Resource allocation over multirate wireless networks: A network utility maximization perspective," *Computer Networks*, vol. 55, no. 11, pp. 2658–2674, Aug. 2011.
- [10] J. Ha and C.-H. Choi, "TCP fairness for uplink and downlink flows in WLANs," in *IEEE Globecom*, Dec. 2006.
- [11] J. Karlsson, A. Kassler, and A. Brunstrom, "Impact of packet aggregation on TCP performance in wireless mesh networks," in *IEEE WoWMoM*, Jun. 2009.
- [12] K. Kashibuchi, A. Jamalipour, and N. Kato, "Channel occupancy time based TCP rate control for improving fairness in IEEE 802.11 DCF," *IEEE Trans. on Vehicular Tech.*, vol. 59, no. 6, pp. 2974–2985, 2010.
- [13] F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, vol. 39, pp. 237–252, 1998.
- [14] B. S. Kim, H. Y. Hwang, and D. K. Sung, "Effect of frame aggregation on the throughput performance of IEEE 802.11n," in *IEEE WCNC*, Mar. 2008.
- [15] Y. Kim, S. Choi, K. Jang, and H. Hwang, "Throughput enhancement of IEEE 802.11 wlan via frame aggregation," in *IEEE 60th. Vehicular Technology Conference*, Sep. 2004.
- [16] A. Kumar, E. Altman, D. Miorandi, and M. Goyal, "New insights from a fixed-point analysis of single cell IEEE 802.11 WLANs," *IEEE/ACM Transactions on Networking*, vol. 15, no. 3, pp. 588–601, Jun. 2007.
- [17] NS-3Project, *NS-3 Manual*, 2011.
- [18] D. Skordoulis, N. Qiang, H.-H. Chen, A. Stephens, L. Changwen, and A. Jamalipour, "IEEE 802.11n MAC frame aggregation mechanisms for next-generation high-throughput WLANs," *IEEE Wireless Communications*, vol. 15, no. 1, pp. 40–47, 2008.
- [19] R. Srikant, *The Mathematics of Internet Congestion Control*. Boston, MA: Birkhäuser, 2004.
- [20] P. Viswanath, D. Tse, and R. Laroia, "Opportunistic beamforming using dumb antennas," *IEEE Tran. on Inf. Theory*, vol. 48, no. 6, pp. 1277–1294, 2002.
- [21] Q. Xia, X. Jin, and M. Hamdi, "Active queue management with dual virtual proportional integral queues for TCP uplink/downlink fairness in infrastructure WLANs," *IEEE Trans. on Wireless Communications*, vol. 7, no. 6, pp. 2261–2271, 2008.