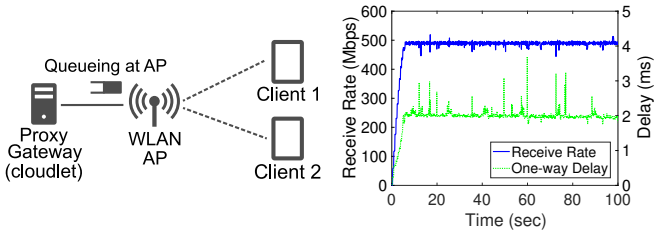
Low-Delay Proportional Fair Rate Allocation For 802.11ac WLAN Downlink

Francesco Gringoli, Douglas J. Leith

***Abstract*—In this paper we consider a next generation edge architecture where traffic is routed via a proxy located close to the network edge (e.g. within a cloudlet). This creates freedom to implement new transport layer behaviour over the wireless path between proxy and clients. We use this freedom to develop a novel approach to achieving high rate, low latency communication on the downlink. This works by adjusting the send rates to clients so as to regulate the aggregation level of transmitted frames which in turn robustly regulates the queueing delay at the AP. We derive the low-delay proportional fair rate allocation in the presence of aggregation and using this consider primal-dual and PI controller approaches for achieving the low-delay proportional fair rate allocation. We find that the primal-dual approach is fragile in the sense that it is sensitive to modelling errors, and in contrast the PI controller approach is much more robust. We present numerical simulation results evaluating the performance.**



(a) (b)

Fig. 1. (a) Cloudlet-based edge transport architecture with bottleneck in the WLAN hop (therefore queueing of downlink packets occurs at the AP as indicated on schematic). One great advantage of this architecture is its ease of rollout since the new transport can be implemented as an app on the clients plus a proxy deployed in the cloud; no changes are required to existing AP's or servers. (b) Illustrating low-latency high-rate operation in an 802.11ac WLAN (measurements are from a hardware testbed located in an office environment).

I. Introduction

In this paper we consider next generation edge transport architectures of the type illustrated in Figure 1(a) with the objective of reliably achieving high rate, low latency communication on the downlink. Traffic to and from client stations is routed via a proxy located close to the network edge (e.g. within a cloudlet). This creates the freedom to implement new transport layer behaviour over the path between proxy and clients, which in particular includes the last wireless hop.

For high rate, low latency communication we would like to select a downlink send rate which is as high as possible yet ensures that a persistent queue backlog does not develop at the wireless access point (AP). While measurements of one-way delay might be used to infer the onset of queueing and adjust the send rate, measuring one-way delay is known to be challenging[1] as is inference of queueing from one-way delay[2]. Use of round-trip time to estimate the onset of queueing is also known to be inaccurate when there is queueing in the reverse path. In this paper we avoid these difficulties by using measurements of the aggregation level of the frames transmitted by the AP. Use of aggregation is ubiquitous in modern WLANs since it brings goodput near to line-rate by reducing the relative time spent in accessing the channel when transmitting several packets to the same destination. Intuitively, the number of packets aggregated in transmitted frames is strongy coupled to the queue backlog at the AP and so adjusting the send rate so as to regulate the aggregation level acts to regulate the queueing delay at the AP.

The main contributions of the paper are: (i) derivation of the low-delay proportional fair rate allocation in the presence of packet aggregation, (ii) derivation of primal-dual and PI controller approaches for achieving the proportional fair rate allocation and (iii) presentation of numerical simulation results evaluating the performance. Figure 1(b) shows typical results obtained from a hardware testbed located in an office environment. It can be seen that the one-way delay is low, at around 2ms, while the send rate is high, at around 500Mbps (this data is for an 802.11ac downlink using three spatial streams and MCS 9). Increasing the send rate further leads to sustained queueing at the AP and an increase in delay, but the results in Figure 1(b) illustrate the practical feasibility of operation in the regime where the rate is maximised subject to the constraint that sustained queueing is avoided.

II. Related Work

In recent years there has been an upsurge in interest in userspace transports due to their flexibility and support for innovation combined with ease of rollout. This has been greatly facilitated by the high efficiency possible in userspace with the support of modern kernels. Notable examples of new transports developed in this way include Google QUIC [1], UDT [2] and Coded TCP [3], [4], [5]. ETSI has also recently set up a working group to study next generation protocols for 5G [6]. The use of performance enhancing proxies, including in the context of WLANs, is also not new e.g. RFC3135 [7] provides an entry point into this literature. However, none of these exploit the use of aggregation in WLANs to achieve high rate, low delay communication.

[1]E.g. due to clock offset and skew between sender and receiver.

[2]For example, the transmission delay across a wireless hop can change significantly over time depending on the number of active stations, e.g. if a single station is active and then a second station starts transmitting the time between transmission opportunities for the original station may double, and it is difficult to distinguish changes in delay due to queueing and changes due to factors such as this.

Interest in using aggregation in WLANs pre-dates the development of the 802.11n standard in 2009 but has primarily focused on analysis and design for wireless efficiency, managing loss etc. For a recent survey see for example [8].

Control theoretic analysis of WLANs has received relatively little attention in the literature, and has almost entirely focussed on MAC layer resource allocation, see for example [9], [10], [11], [12] and references therein. In contrast, there exists a substantial literature on control theoretic analysis of congestion control at the transport layer, see for example the seminal work in [13], [14]. However, this has mainly focussed on end-to-end behaviour in wired networks with queue overflow losses and has largely ignored detailed analysis of operation over WLANs. This is perhaps unsurprising since low delay operation at the network edge has only recently come to the fore as a key priority for next generation networks.

TCP BBR [15] is currently being developed by Google and this also targets high rate and low latency, although not specifically in edge WLANs. The BBR algorithm tries to estimate the bottleneck bandwidth and adapt the send rate accordingly to try to avoid queue buildup. The delivery rate in BBR is defined as the ratio of the in-flight data when a packet departed the server to the elapsed time when its ACK is received. This may be inappropriate, however, when the bottleneck is a WLAN hop since aggregation can mean that increases in rate need not correspond to increases in delay plus a small queue at the AP can be benificial for aggregation and so throughput.

III. Preliminaries

A. Aggregation In 802.11ac

A feature shared by all WLAN standards since 2009 (when 802.11n was introduced) has been the use of aggregation to amortise PHY and MAC framing overheads across multiple packets. This is essential for achieving high throughputs. Since the PHY overheads are largely of fixed duration, increasing the data rate reduces the time spent transmitting the frame payload but leaves the PHY overhead unchanged. Hence, the efficiency, as measured by the ratio of the time spent transmitting user data to the time spent transmitting an 802.11 frame, decreases as the data rate increases unless the frame payload is also increased i.e. several data packets are aggregated and transmitted in a single frame.

In 802.11ac two types of aggregation are used. Namely, one or more ethernet frames may aggregated into a single 802.11 AMSDU frame, and multiple AMSDUs then aggregated into an 802.11 AMPDU frame. Typically, two ethernet frames are aggregated into one AMSDU and up to 64 AMSDUs aggregated into one AMPDU, allowing up to 128 ethernet frames to be sent in a single AMPDU.

The level of aggregation can be readily measured at a receiver either directly from packet MAC timestamps when these are available (which depends on hardware support and the availability of root access) or by applying machine-learning techniques to packet kernel timestamps (which are always available via the standard socket API), see [16] for details.

B. Modelling Aggregation Level in Paced 802.11ac WLANs

For control analysis and design we need a model of the relation between send rate and aggregation level on the downlink. The model does not need to be exact since the feedback action of the controller can compensate for model uncertainty, but should capture the broad relationship. In general, finite-load analytic modelling of aggregation in 802.11ac is challenging due to: (i) the randomness in the time between frame transmissions caused by the stochastic nature of the CSMA/CA MAC and (ii) correlated bursty packet arrivals. As a result most finite-load analysis of packet aggregation to date has resorted to use of simulations. Fortunately, since we control the sender we can use packet pacing and when packets are paced simple yet accurate analytic models are known [17]. Following [17], let n denote the number of client stations in the WLAN, x_i the mean send rate to client i in packets/sec and $\boldsymbol{x} = (x_1, \ldots, x_n)$ the vector of mean send rates. Let $\mu_{T_{oh}}$ denote the time per frame used for CSMA/CA channel access, PHY and MAC headers plus transmission of the MAC ACK so that $c = n\mu_{T_{oh}}$ is the aggregate overhead for a round of transmissions to the n client stations, w_i the mean time to transmit one packet to client i at the PHY data rate and $\boldsymbol{w} = (w_1, \ldots, w_n)^T$ the vector of transmit times, N_{max} the maximum number of packets that can be sent in a frame (typically 32 or 64). Parameters c and $\boldsymbol{w}$ capture the WLAN PHY/MAC configuration. Define μ_{N_i} to be the mean number of packets in each frame sent to client i and $\mu_{\boldsymbol{N}} = (\mu_{N_1}, \ldots, \mu_{N_n})$ the vector of mean aggregation levels. Then,

$$\mu_{\boldsymbol{N}} = \Pi \frac{c\boldsymbol{x}}{1 - \boldsymbol{w}^T\boldsymbol{x}} = \Pi \boldsymbol{F}(\boldsymbol{x}) \tag{1}$$

where Π denotes projection onto interval $[1, N_{max}]$ and $\boldsymbol{F}(\boldsymbol{x}) := \frac{c\boldsymbol{x}}{1-\boldsymbol{w}^T\boldsymbol{x}}$. The mean delay μ_{T_i} of packets sent to client i is upper bounded by

$$\mu_{T_i} = \max\{\min\{\frac{c}{1 - \boldsymbol{w}^T\boldsymbol{x}}, \frac{N_{max}}{x_i}\}, \frac{1}{x_i}\} \tag{2}$$

Observe that $\boldsymbol{F}(\boldsymbol{x})$ is monotonically increasing for feasible rate vectors $\boldsymbol{x}$ since $\frac{\partial F_i'(\boldsymbol{x})}{\partial x_i} = \frac{c}{(1-\boldsymbol{w}^T\boldsymbol{x})^2}(1-\boldsymbol{w}^T\boldsymbol{x}+w_ix_i) > 0$ and $\frac{\partial F_i'(\boldsymbol{x})}{\partial x_j} = \frac{cw_ix_i}{(1-\boldsymbol{w}^T\boldsymbol{x})^2} > 0$ when $x_i \geq 0$ and $\boldsymbol{w}^T\boldsymbol{x} < 1$. Hence, $\boldsymbol{F}(\boldsymbol{x})$ is one-to-one and so invertible. In particular,

$$\boldsymbol{F}^{-1}(\mu_{\boldsymbol{N}}) = \frac{\mu_{\boldsymbol{N}}}{c + \boldsymbol{w}^T\mu_{\boldsymbol{N}}} \tag{3}$$

and it can be verified that $\boldsymbol{F}(\boldsymbol{F}^{-1}(\mu_{\boldsymbol{N}})) = \mu_{\boldsymbol{N}}$. Given rate vector $\boldsymbol{x}$ we can therefore obtain the corresponding aggregation level from $F(\boldsymbol{x})$ and, conversely, given aggregation level vector $\mu_{\boldsymbol{N}}$ we can obtain the corresponding rate vector from $\boldsymbol{F}^{-1}(\mu_{\boldsymbol{N}})$. This will prove convenient in the analysis below since it means we can freely change variables between $\boldsymbol{x}$ and $\mu_{\boldsymbol{N}}$. For example, substituting $\boldsymbol{x} = \boldsymbol{F}^{-1}(\mu_{\boldsymbol{N}})$ the term $\frac{c}{1-\boldsymbol{w}^T\boldsymbol{x}}$ in the mean delay (2) can be expressed equivalently in terms of $\mu_{\boldsymbol{N}}$ as,

$$\frac{c}{1 - \boldsymbol{w}^T\boldsymbol{x}} = c + \boldsymbol{w}^T\mu_{\boldsymbol{N}} \tag{4}$$

IV. Proportional Fair Low-Delay Rate Allocation

A. Utility Fair Optimisation Is Convex

Our interest is in achieving high rates while maintaining low delay at the AP. The proportional fair low delay rate allocation is the solution to the following utlity-fair optimisation P:

$$\max_{\boldsymbol{x}\in\mathbb{R}_+^n} \sum_{i=1}^{n} \log x_i \tag{5}$$

$$s.t.\ \mu_{T_i}(\boldsymbol{x}) \leq \bar{T}, i = 1,\ldots,n \tag{6}$$

$$\mu_{N_i}(\boldsymbol{x}) \leq \bar{N}, i = 1,\ldots,n \tag{7}$$

Constraint (6) ensures that the mean delay at the AP is no more than upper limit $\bar{T}$, where $\bar{T}$ is a QoS parameter. Constraint (7) ensures that we operate at an aggregation level no more than $\bar{N} < N_{max}$.

Substituting from (2) the constraints (6) can be written[3] as $\frac{c}{1-\boldsymbol{w}^T\boldsymbol{x}} \leq \bar{T}$. Rearranging gives $c \leq \bar{T}(1-\boldsymbol{w}^T\boldsymbol{x})$ i.e. $\boldsymbol{w}^T\boldsymbol{x} \leq 1-c/\bar{T}$. In this form it can be seen that the constraint is linear, and so convex. Similarly, substituting from (1) the constraints (7) can be written equivalently as $\frac{cx_i}{1-\boldsymbol{w}^T\boldsymbol{x}} \leq \bar{N}$, $i = 1,\ldots,n$. Rearranging gives $cx_i \leq \bar{N}(1-\boldsymbol{w}^T\boldsymbol{x})$ i.e. $cx_i + \bar{N}\boldsymbol{w}^T\boldsymbol{x} \leq \bar{N}$, which again is linear. Hence, optimisation P can be equivalently rewritten as optimisation P':

$$\max_{\boldsymbol{x}\in\mathbb{R}_+^n} \sum_{i=1}^{n} \log x_i \tag{8}$$

$$s.t.\ \boldsymbol{w}^T\boldsymbol{x} \leq 1 - c/\bar{T} \tag{9}$$

$$cx_i + \bar{N}\boldsymbol{w}^T\boldsymbol{x} \leq \bar{N}, i = 1,\ldots,n \tag{10}$$

which is convex.

B. Characterising The Proportional Fair Solution

The Lagrangian of optimisation P' is $L(\boldsymbol{x},\theta,\boldsymbol{\lambda}) := -\sum_{i=1}^{n}\log x_i + \theta(\boldsymbol{w}^T\boldsymbol{x} - (1-c/\bar{T})) + \sum_{i=1}^{n}\lambda_i(cx_i + \bar{N}\boldsymbol{w}^T\boldsymbol{x} - \bar{N})$ where θ and λ_i, $i = 1,\ldots,n$ are multipliers associated with, respectively, (9) and (10). Since the optimisation is convex the KKT conditions are necessary and sufficient for optimality. Namely, an optimal rate vector $\boldsymbol{x}^*$ satisfies

$$-\frac{1}{x_i^*} + \lambda_i c + \sum_{j=1}^{n}\lambda_j\bar{N}w_i + \theta w_i = 0 \tag{11}$$

i.e.

$$x_i^* = \frac{1}{\lambda_i c + Dw_i} \tag{12}$$

where $D := (\bar{N}\sum_{j=1}^{n}\lambda_j + \theta)$.

Let $U = \{i : \mu_{N_i}(\boldsymbol{x}^*) < \bar{N}\}$ denote the set of stations for which the aggregation level is strictly less than $\bar{N}$ at the optimal rate allocation. By complementary slackness $\lambda_i = 0$ for $i \in U$ and so $x_i^* = 1/(Dw_i)$. That is, $\mu_{N_i} = \frac{cx_i^*}{(1-\boldsymbol{w}^T\boldsymbol{x}^*)} = \frac{c}{D(1-\boldsymbol{w}^T\boldsymbol{x}^*)}\frac{1}{w_i}$. Observe that the first term is invariant with i and so the aggregation level of station $i \in U$ is proportional to $1/w_i = \mu_{R_i}/L$ i.e. to the mean MCS rate of the station. For stations $j \notin U$ the aggregation level $\mu_{N_j}(\boldsymbol{x}^*) = \bar{N}$.

Putting these observations together, it follows that

$$\mu_{N_i}(\boldsymbol{x}^*) = \min\{\frac{c}{D(1-\boldsymbol{w}^T\boldsymbol{x}^*)}\frac{1}{w_i}, \bar{N}\},\ i = 1,\ldots,n \tag{13}$$

Assume without loss that the station indices are sorted such that $w_1 \geq w_2 \geq \cdots \geq w_n$. Then

$$\mu_{N_i}(\boldsymbol{x}^*) = \min\{\mu_{N_1}(\boldsymbol{x}^*)\frac{w_1}{w_i}, \bar{N}\},\ i = 2,\ldots,n \tag{14}$$

Hence, once the optimal $\mu_{N_1}(\boldsymbol{x}^*)$ is determined we can find the optimal aggregation levels for the rest of the stations. With these we can then use inverse mapping (3) to recover the proportional fair rate allocation, namely $x_i^* = \mu_{N_i}/(c + \boldsymbol{w}^T\mu_{\boldsymbol{N}})$.

It remains to determine μ_{N_1}. We proceed as follows.

Lemma 1. *At an optimum $\boldsymbol{x}^*$ of P' then either (i) $\mu_{N_i}(\boldsymbol{x}^*) = \bar{N}$ for all $i = 1,\ldots,n$ or (ii) $\mu_{T_i}(\boldsymbol{x}^*) = \bar{T}$ for all $i = 1,\ldots,n$.*

Proof. We proceed by contradiction. Suppose at an optimum $\mu_{N_i}(\boldsymbol{x}^*) = \frac{cx_i^*}{1-\boldsymbol{w}^T\boldsymbol{x}^*} < \bar{N}$ for some i and $\mu_{T_i}(\boldsymbol{x}^*) = \frac{c}{1-\boldsymbol{w}^T\boldsymbol{x}^*} < \bar{T}$. Then we can increase x_i^* without violating the constraints (with this change $\frac{c}{1-\boldsymbol{w}^T\boldsymbol{x}^*}$ and $\frac{cx_i^*}{1-\boldsymbol{w}^T\boldsymbol{x}^*}$ will both increase, but since the corresponding constraints are slack if the increase in x_i^* is sufficiently small then they will not be violated). Hence, we can improve the objective which yields the desired contradiction since we assumed optimality of $\boldsymbol{x}^*$. Hence when $\mu_{N_i}(\boldsymbol{x}^*) < \bar{N}$ for at least one station then $\mu_{T_i}(\boldsymbol{x}^*) = \bar{T}$. Alternatively, $\mu_{N_i}(\boldsymbol{x}^*) = \bar{N}$ for all stations. □

It follows from Lemma 1 that $\mu_{N_1} = \min\{\bar{T}x_1^*, \bar{N}\}$. Substituting into (14) and combining with inverse mapping (3) it follows that

$$\boldsymbol{x}^* = \boldsymbol{F}^{-1}(\mu_{\boldsymbol{N}}(\boldsymbol{x}^*)) \tag{15}$$

with

$$\mu_{\boldsymbol{N}}(\boldsymbol{x}^*) = \min\{\bar{T}x_1^*, \bar{N}\}\boldsymbol{W} \tag{16}$$

where vector $\boldsymbol{W} = [1, \frac{w_1}{w_2}, \ldots, \frac{w_1}{w_n}]^T$. Recall that the station indices are sorted such that $w_1 \geq w_2 \geq \cdots \geq w_n$ and so all of the elements of $\boldsymbol{W}$ are less than or equal to one, with equality only when the PHY data rate is maximal amongst the client stations. The proportional fair rate $\boldsymbol{x}^*$ and associated aggregation level $\mu_{\boldsymbol{N}}$ can now be found by solving equations (15)-(16). When $\bar{N} = +\infty$ or $\bar{T} = +\infty$ it can be verified that the proportional fair solution is an equal airtime one.

V. Online Algorithms

A. Primal-Dual Approach

We can use the following primal-dual algorithm to solve for the proportional fair rate allocation

$$\boldsymbol{x}_{k+1} \in \arg\min_{\boldsymbol{x}\in\mathbb{R}_+^n} L(\boldsymbol{x},\theta_k,\boldsymbol{\lambda}_k) \tag{17}$$

$$\theta_{k+1} = [\theta_k + \alpha(\boldsymbol{w}^T\boldsymbol{x}_k - (1-c/\bar{T}))]^+ \tag{18}$$

$$\lambda_{i,k+1} = [\lambda_{i,k} + \alpha(cx_{i,k} + \bar{N}\boldsymbol{w}^T\boldsymbol{x}_k - \bar{N})]^+ \tag{19}$$

[3]Note that constraint (7) ensures $\mu_{N_i}(\boldsymbol{x}) \leq \bar{N} < N_{max}$ and so $\mu_{T_i}(\boldsymbol{x}) < N_{max}/x_i$. Since our interest is primarily in applications requiring high rates we assume for simplicity that $\frac{c}{1-\boldsymbol{w}^T\boldsymbol{x}} \geq \frac{1}{x_i}$ although this could be added as the additional linear constraint $cx_i + \boldsymbol{w}^T\boldsymbol{x} \geq 1$ if desired.

where $[x]^+ = x$ when $x \geq 0$ and 0 otherwise, α is a step-size parameter. Since the optimisation is convex and the Slater condition is satisfied, by standard results this algorithm converges to the solution of P' i.e. to the low-delay proportional fair rate allocation.

The optimisation at step (17) in the primal-dual algorithm can be avoided as follows. Dropping constant terms (that do not depend on $\boldsymbol{x}$, we have that

$$\begin{aligned}&\arg \min_{\boldsymbol{x}\in\mathbb{R}_+^n} L(\boldsymbol{x}, \theta_k, \boldsymbol{\lambda}_k)\\ &= \arg \min_{\boldsymbol{x}\in\mathbb{R}_+^n} -\sum_{i=1}^n \log x_i + \theta_k \boldsymbol{w}^T\boldsymbol{x} + \sum_{i=1}^n \lambda_{i,k}(cx_i + \bar{N}\boldsymbol{w}^T\boldsymbol{x})\end{aligned} \tag{20}$$

Similarly to (12), applying the KKT conditions to the LHS and rearranging yields that the minimiser $\boldsymbol{x}$ has i'th element equal to $\frac{1}{\lambda_i c + D w_i}$ where $D := \bar{N}\sum_{j=1}^n \lambda_j + \theta$. Hence we can replace (17) by

$$\boldsymbol{x}_{k+1} = \left(\frac{1}{\lambda_{k,1}c + D_k w_1}, \ldots, \frac{1}{\lambda_{k,n}c + D_k w_n}\right) \tag{21}$$

with $D_k := \bar{N}\sum_{j=1}^n \lambda_{j,k} + \theta_k$.

The resulting low delay rate allocation algorithm requires knowledge of $\boldsymbol{w}$, $\bar{N}$, $\bar{T}$ and c. The PHY MCS rates of received frames can be readily observed at client stations and reported back to the sender, so $\boldsymbol{w}$ is known. $\bar{N}$ and $\bar{T}$ are design parameters and so also known. However, parameter $c = n\mu_{T_{oh}}$ cannot be directly measured because the mean channel access time $\mu_{T_{oh}}$ cannot be measured directly (since we consider the transport layer we assume we do not have access to the MAC on the AP) and it depends on the channel state and so may be strongly affected by neighbouring WLANS, interference etc. Nevertheless we can estimate c as follows. Recall that $\mu_{N_i} = \frac{cx_i}{1-\boldsymbol{w}^T\boldsymbol{x}}$, i.e. $c = \frac{\mu_{N_i}}{x_i}(1 - \boldsymbol{w}^T\boldsymbol{x})$. Motivated by this observation we use the following as an estimator of c,

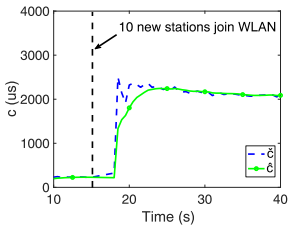
$$\hat{c}(k+1) = (1-\beta)\hat{c}(k) + \beta\check{c}(k) \tag{22}$$

with $\check{c}(k) := \frac{\tilde{\mu}_{N_1}(k)}{x_1(k)}(1 - \boldsymbol{w}^T\boldsymbol{x}(k))$, where β is a design parameter which controls the window over which the moving average is calculated (a typical value is $\beta = 0.05$).

Figure 2(b) illustrates the ability of this estimator to track a fairly significant change in the network conditions, namely 10 new stations joining the WLAN at time 15s and starting downlink transmissions. These new stations cause a change in c from a value of around $200\mu s$ to around $2200\mu s$ i.e. a change of more than an order of magnitude. It can be seen that estimator (22) tracks this large change without difficulty. We observe similar tracking behaviour for changes in MCS and also when the channel is shared with other legacy WLANs.

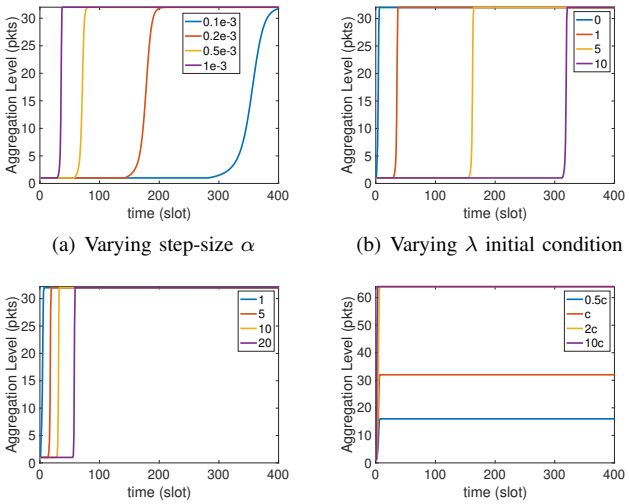
B. Convergence Rate of Primal-Dual Algorithm

The convergence rate of the primal-dual algorithm to the optimal rate allocation is determined by dynamics of the multiplier updates (18)-(19). These can be tuned by adjusting the step-size α and the initial condition for the multipliers θ and $\boldsymbol{\lambda}$. Figure 3(a) illustrates the impact of the choice of step-size on the rate of convergence. As might be expected,



(a)

Fig. 2. Illustrating c estimator (22) tracking a sharp change in the number of stations from $n = 1$ to $n = 11$ at time 15s. NS3 simulation: one client station, $\bar{N} = 32$, NSS=1, MCS=9



(a) Varying step-size α

(b) Varying λ initial condition

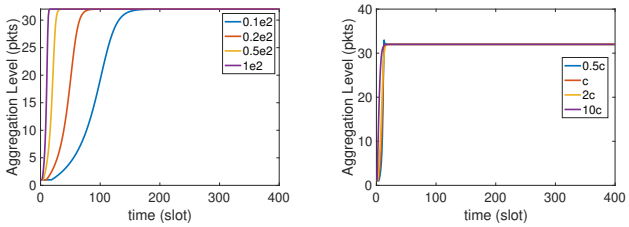
(c) Varying number n of stations

(d) Varying c

Fig. 3. Impact on primal-dual convergence rate of choice of step-size α, initial value for multiplier λ, number of client stations n and mismatch in model and WLAN parameter c. One client station, $\bar{N} = 32$, $\bar{T} = +\infty$, NSS=1, MCS=9. In (a) the initial value of multiplier $\lambda(0)$ is 1, in (b) $\alpha = 10^{-3}$, in(c) and (d) $\lambda(0) = 0$, $\alpha = 10^{-3}$.

a larger step-size leads to faster convergence, although too large a choice of step-size can lead to the dynamics becoming divergent (which happens for step-sizes larger than about 2×10^{-3}). Observe the initial delay in Figure 3(a) during which the aggregation level stays close to zero. The delay increases as the step-size decreases, but even for a relatively large step size of 1×10^{-3} the delay is around 50 time slots (about 500s if a slot is 10s duration). This is due to the sensitivity of the multiplier dynamics to the choice of initial condition for $\boldsymbol{\lambda}$, see Figure 3(b). When a single station is used it can be seen from Figure 3(b) that selecting $\lambda(0) = 0$ gives fast convergence (in Figure 3(a) $\lambda(0) = 1$). Unfortunately, the choice of $\lambda(0)$ for fast convergence is dependent on the number of client stations in the WLAN, see Figure 3(c) and while $\lambda(0) = 0$ gives fast convergence with a single station it leads to a delay of around 50 slots with 10 stations.

Still more seriously, the rate allocation depends strongly on the accuracy of the MAC ovehead parameter c used in the primal-dual rate algorithm. For example, Figure 3(d) plots the performance when there is a mismatch between the value of c used in the prima-dual update and the true value in the



(a) Varying control gain K_1 (b) Varying c

Fig. 4. Impact on PI controller convergence rate of choice of control gain K_1 and mismatch in model and WLAN parameter c. One client station, $\bar{N} = 32$, $\bar{T} = +\infty$, NSS=1, MCS=9. In (b) $K_1 = 100$.

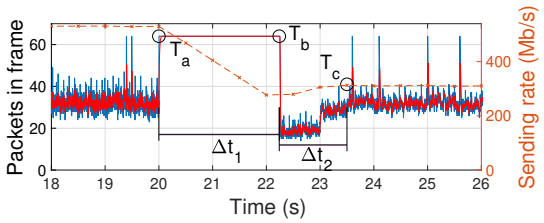


Fig. 5. Illustrating adaption of send rate by feedback algorithm in response to a change in channel conditions (from use of 2 spatial streams down to 1 spatial stream). NS3 simulation, single client station, MCS 9, $K_0 = 1$, $\Delta = 500$ms, $N_\epsilon = 32$, $N_{max} = 64$.

WLAN – the plot legend shows the scaking factor relating the WLAN c value and the value used in the update. It can be seen that the performance of the primal-dual is quite fragile in that even relatively small mistmatches in c lead to poor tracking of the target aggregation level $\bar{N} = 32$ packets. While it might be tempting to modify multiplier update (19) to use a direct measurement of the aggregation level in place of expression $cx_{i,k} + \bar{N}\boldsymbol{w}^T\boldsymbol{x}_k$, it turns out that this is of little help since c also affects the mapping (21) from $\boldsymbol{\lambda}$ to rate $\boldsymbol{x}$ and so mismatches in the value of c in the WLAN and in the primal-dual update continue to result in poor performance.

C. PI-Controller

The analysis in the previous section highlights that the standard primal-dual update for finding the low-delay proportional fair rate allocation is rather fragile in the sense that its performance rapidly degrades when there is a mismatch between the model used in the update and the behaviour of the actual WLAN - such mismatches can be expected to be common of course since all models are only approximate. In addition, even when the model happens to be accurate the convergence rate is sensitive to the choice of multiplier initial conditions and since the appropriate choice depends on the number of stations no fixed choice is enough to ensure fast convergence.

These observations motivate us to consider an alternative approach that exploits the structure of the proportional fair rate allocation and uses a feedback controller to directly regulate the send rates. Namely, we know that the proportional fair allocation has rate vector $\boldsymbol{x} = (x_1, \frac{w_1}{w_2}x_1, \ldots, \frac{w_1}{w_n}x_1)$ when the stations are ordered so that the station with highest MCS rate is given index 1 (and so rate x_1). We also know that rate x_1 is such that either the aggregation level for station 1 is $\bar{N}$ or the delay is $\bar{T}$, whichever occurs first. Hence, we can use a feedback controller to adjust rate x_1 to ensure this. Assuming, for the moment, that the $\bar{N}$ bites first we can use the following simple PI controller

$$x_1(k+1) = x_1(k) + K_1(\bar{N} - \mu_{N_1}(k)),\ \boldsymbol{x}(k) = x_1(k)\boldsymbol{W} \tag{23}$$

where $\boldsymbol{W} := (1, \frac{w_1}{w_2}, \ldots, \frac{w_1}{w_n})$. This update adjusts the send rate so that the aggregation level $\mu_{N_1}(k)$ of station 1 equals $\bar{N}$ and then sets the rate for the other stations i proportional to $\frac{w_1}{w_i}$.

Figure 4(a) illustrates the impact of the choice of control gain K_1 on the convergence rate. As might be expected, it can be seen that the convergence rate increases as the gain is increased, although of course if the gain is made too large the feedback loop eventually becomes unstable. In contrast to the primal-dual update, use of feedback allows the update to compensate for mismatches between the model and the actual WLAN behaviour, see Figure 4(b). That is, the PI control approach is much more robust to modelling errors than the primal-dual approach.

VI. Experimental Measurements

A. NS3 Simulator Implementation

We implemented the PI controller in the NS3 packet-level simulator. Based on the received feedbacks it periodically configures the sending rate of `udp-client` applications colocated at a single node connected to an Access Point. Each wireless station receives a single UDP traffic flow at a `udp-server` application that we modified to collect frame aggregation statistics and periodically transmit these to the controller at intervals of Δ ms. We also developed a round-robin scheduler at the AP with separate queue for each destination, and we added new functions to let stations determine the MCS of each received frame together with the number of MPDU packets it contains. The maximum aggregation level permitted is N_{max}=64. We configured 802.11ac to use a physical layer operating over an $80MHz$ channel, VHT rates for data frames and legacy rates for control frames. The PHY MCS and the number of spatial streams NSS used can be adjusted.

B. Responding to Channel Changes

The feedback algorithm used by the sender regulates the send rate to maintain a target aggregation level. It therefore adapts automatically to changing channel conditions. To investigate this we change to using NS3 simulations since this allows us to change the channel in a controlled manner (we also have experimental measurements showing adaptation to changing channel conditions, not shown here, but in this case we do not know ground truth).

Fig. 5 illustrates typical behaviour of the feedback algorithm. Initially the AP uses 2 spatial streams and then at $t = T_a = 20s$ it switches to 1 spatial stream. For $\Delta t_1 = 2.24s$ all AMPDUs hit the maximum aggregation level of 64 packets and we start observing losses. During this time it can be

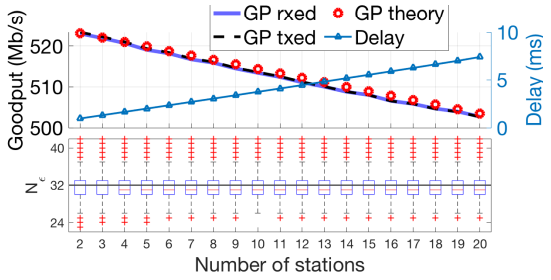


Fig. 6. Sum-goodput and delay vs number of receivers (top) and corresponding distribution of aggregation level about the target value of $N_\epsilon = 32$ (bottom). In the top plot the *GP theory* line (GP, goodput) is a theoretical upper limit computed by assuming an AMPDU with $N_\epsilon = 32$ packets, 10 feedback packets per second per receiver, 10 beacons per second, and no collisions. NS3 simulations, $K_0 = 1$, $\Delta = 500$ms, $N_{max} = 64$.

seen that the algorithm, which updates the send rate twice per second ($\Delta = 500ms$), is slowing down the sending rate. It takes four rounds to reach a rate compatible with the channel, but it takes a little bit more to stabilise the aggregation level at the AP in $t = T_b$. After another three rounds (for approximately $\Delta t_2 = 1.26s$) it can be seen that the sending rate settles at its new value in $t = T_c$.

C. Fairness With High Rate & Low Delay

We now explore performance with multiple client stations. Fig. 6 (top) shows measurements of the aggregated application layer goodput and average delay vs the number of receivers where client stations are all located at the same distance from the AP. It can be seen that the aggregated goodput measured at the receivers is close to the theoretical limit supported by the channel (MCS) configuration, being only a few Kb/s below this for 20 receivers. This goodput is evenly shared by the receivers (the measured Jain's Fairness Index is always 1). The average delay increases almost linearly at the rate of $350\mu s$ per additional station. The lower plot in Fig. 6 shows the measured distribution of frame aggregation level, with the edges of the boxes indicating the 25th and 75th percentiles. It can be seen that the feedback algorithm tightly concentrates the aggregation level around the target value of $N_\epsilon = 32$. As expected, since the delay is regulated to a low level we did not observe any losses.

VII. Conclusions

In this paper we consider a next generation edge architecture where traffic is routed via a proxy located close to the network edge (e.g. within a cloudlet). This creates freedom to implement new transport layer behaviour over the wireless path between proxy and clients. We use this freedom to develop a novel approach to achieving high rate, low latency communication on the downlink. This works by adjusting the send rates to clients so as to regulate the aggregation level of transmitted frames which in turn robustly regulates the queueing delay at the AP. We derive the low-delay proportional fair rate allocation in the presence of aggregation and using this consider primal-dual and PI controller approaches for achieving the low-delay proportional fair rate allocation. We find that the primal-dual apprroach is fragile in the sense that it is sensitive to modelling errors, and in contrast the PI controller approach is much more robust. We present numerical simulation results evaluating the performance.

References

[1] J. Iyengar and I.Swett, "QUIC: A UDP-Based Secure and Reliable Transport for HTTP/2," *IETF Internet Draft*, 2015. [Online]. Available: https://tools.ietf.org/html/draft-tsvwg-quic-protocol-00

[2] Y. Gu and R. L. Grossman, "Udt: Udp-based data transfer for high-speed wide area networks," *Comput. Netw.*, vol. 51, no. 7, pp. 1777–1799, May 2007. [Online]. Available: http://dx.doi.org/10.1016/j.comnet.2006.11.009

[3] M. Kim, J. Cloud, A. ParandehGheibi, L. Urbina, K. Fouli, D. J. Leith, and M. Medard, "Congestion control for coded transport layers," in *Proc IEEE International Conference on Communications (ICC)*, 2014, pp. 1228–1234.

[4] M. Karzand, D. J. Leith, J. Cloud, and M. Medard, "Design of FEC for Low Delay in 5G," *IEEE Journal Selected Areas in Communications (JSAC)*, vol. 35, no. 8, pp. 1783–1793, 2016.

[5] A. Garcia-Saavedra, M. Karzand, and D. J. Leith, "Low Delay Random Linear Coding and Scheduling Over Multiple Interfaces," *IEEE Trans on Mobile Computing*, vol. 16, no. 11, pp. 3100–3114, 2017.

[6] *Next Generation Protocols – Market Drivers and Key Scenarios*. European Telecommunications Standards Institute (ETSI), 2016. [Online]. Available: http://www.etsi.org/images/files/ETSIWhitePapers/etsi_wp17_Next_Generation_Protocols_v01.pdf

[7] J. Border, M. Kojo, J. Griner, G. Montenegro, and Z. Shelby, "Performance enhancing proxies intended to mitigate link-related degradations," Internet Requests for Comments, RFC Editor, RFC 3135, June 2001.

[8] R. Karmakar, S. Chattopadhyay, and S. Chakraborty, "Impact of ieee 802.11n/ac phy/mac high throughput enhancements on transport and application protocols: A survey," *IEEE Communications Surveys Tutorials*, vol. 19, no. 4, pp. 2050–2091, Fourthquarter 2017.

[9] A. Banchs, P. Serrano, and A. Azcorra, "End-to-end delay analysis and admission control in 802.11 DCF wlans," *Computer Communications*, vol. 29, no. 7, pp. 842–854, 2006. [Online]. Available: https://doi.org/10.1016/j.comcom.2005.08.006

[10] G. Boggia, P. Camarda, L. A. Grieco, and S. Mascolo, "Feedback-based control for providing real-time services with the 802.11e MAC," *IEEE/ACM Trans. Netw.*, vol. 15, no. 2, pp. 323–333, 2007. [Online]. Available: http://doi.acm.org/10.1145/1279660.1279666

[11] A. Garcia-Saavedra, A. Banchs, P. Serrano, and J. Widmer, "Distributed opportunistic scheduling: A control theoretic approach," in *Proceedings of the IEEE INFOCOM 2012, Orlando, FL, USA, March 25-30, 2012*, 2012, pp. 540–548. [Online]. Available: https://doi.org/10.1109/INFCOM.2012.6195795

[12] P. Serrano, P. Patras, A. Mannocci, V. Mancuso, and A. Banchs, "Control theoretic optimization of 802.11 wlans: Implementation and experimental evaluation," *Computer Networks*, vol. 57, no. 1, pp. 258–272, 2013. [Online]. Available: https://doi.org/10.1016/j.comnet.2012.09.010

[13] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, "Rate control for communication networks: shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, vol. 49, no. 3, pp. 237–252, Mar 1998. [Online]. Available: https://doi.org/10.1057/palgrave.jors.2600523

[14] C. Jin, D. X. Wei, and S. H. Low, "FAST TCP: motivation, architecture, algorithms, and performance," in *Proceedings IEEE INFOCOM 2004, The 23rd Annual Joint Conference of the IEEE Computer and Communications Societies, Hong Kong, China, March 7-11, 2004*, 2004, pp. 2490–2501. [Online]. Available: https://doi.org/10.1109/INFCOM.2004.1354670

[15] N. Cardwell, Y. Cheng, C. S. Gunn, S. H. Yeganeh, and V. Jacobson, "Bbr: Congestion-based congestion control," *Commun. ACM*, vol. 60, no. 2, pp. 58–66, 2017.

[16] F. G. Hamid Hassani and D. J. Leith, "Quick and Plenty: Achieving Low Delay and High Rate in 802.11ac Edge Networks," *Computer Networks*, vol. 187, no. 14, 2021.

[17] F. Gringoli and D. J. Leith, "Modelling Downlink Packet Aggregation in Paced 802.11ac WLANs," 2021. [Online]. Available: http://arxiv.org/abs/2101.07562