Institut Eurécom
Department of Mobile Communications
2229, route des Crêtes
B.P. 193
06904 Sophia-Antipolis
FRANCE

Research Report RR-07-206

Improving Flow level Fairness and Interactivity in WLANs using Size-based Scheduling Policies

November 7th, 2007

Guillaume Urvoy-Keller*, and Andre-Luc Beylot+
* Institut Eurecom
+ ENSEEIHT, Laboratoire IRIT
F-31071 Toulouse Cedex 7

Tel : (+33) 4 93 00 81 00
Fax : (+33) 4 93 00 82 00
Email : Guillaume.Urvoy@eurecom.fr, beylot@enseeiht.fr

[1]Institut Eurécom's research is partially supported by its
industrial members: BMW Group Research & Technology - BMW Group Company, Bouygues Télécom, Cisco Systems, France Télécom, Hitachi Europe, SFR, Sharp, STMicroelectronics, Swisscom, Thales.

Improving Flow level Fairness and Interactivity in WLANs using Size-based Scheduling Policies

Guillaume Urvoy-Keller, and Andre-Luc Beylot

Abstract

While 802.11-based Wireless LANs see increasing public deployment, their performances remain way below the ones obtained with a traditional Ethernet access. With 802.11 WLANs access, the bandwidth available at a given station can heavily fluctuate over time. In addition, the access point, with its limited resources and a probability to access the medium similar to any other wireless stations, is a source of major unfairness for TCP connections. This is especially the case when data flows in both directions, i.e., from and to the wireless stations. The problem stems from the firece competition that takes place at the access point between the non responsive TCP ack streams and the responsive TCP data streams.

In this paper, we propose the use of size based scheduling policies at the IP layer of the access point to both enforce fairness among TCP connections and to improve the interactivity perceived by the end user. The latter is defined as the ability of the network to maintain small response times to the short flows that are generated by the interactive applications of the users, e.g., mail, or web browsing, when the overall load increases. We first demonstrate that the popular Least Attained Service (LAS) policy, which has been proposed in the context of the wired Internet, is able to work properly only when data flows in a single direction. To deal with the general case when uploads compete with downloads, we propose a new flavor of LAS, called LASACK. LASACK mitigates the impact of the non responsivness of TCP ack streams by assigning a priority to a TCP ack packet that is a function of the number of bytes sent by the corresponding data stream. We demonstrate using simulations under a variety of realistic workloads, that LASACK is able to enforce fairness and maintain a good interactivity even when the load increases. Another contribution of this work is to investigate the use of simple queuing models to capture the dynamics of the queue at the access point. In particular, we develop a queueing model for the LASACK policy that turns out to be accurate under a variety of loads and flow size distributions.

Contents

List of Figures

1 Introduction

Computer users tend to prefer wireless access over legacy wired Ethernet access to access a network. This trend is noticeable in home networking, public hotspots and Intranet environments. While a wireless access is an appealing solution, its performance is in general way below the one of a legacy Ethernet access. A main drawback of the 802.11 protocol in WLAN scenarios is that the medium is shared among all connections and all stations, including the access point itself, have the same probability to access the medium. The shared nature of the medium coupled with the low resources available at the access point in terms of buffer sizes represent a major threat to TCP connections that bear most of the bytes in WLANs like it does in classical LANs [1, 2].

In this paper, we focus on scenarios where fixed users access the Internet through a single access point. Our focus is on the performance of TCP connections and the unfairness that results from the competition between uploads and downloads. A specific problem with TCP is that the ack streams are unresponsive. In a WLAN, where TCP transfers are performed in both directions, i.e., from and to the wireless stations, a fierce competition takes place between TCP data streams and TCP ack streams [3].

We investigate the use of specific scheduling policies at the access point to enforce fairness among TCP transfers.

As a starting point, we consider as an alternative to the legacy FIFO scheduler at the IP layer, a size-based scheduling policies called Least Attained Service (LAS). With LAS, the priority of a flow decreases when the amount of service received by this flow, i.e., the amount of bytes sent, increases. This simple trick allows to enforce fairness among competing flows with similar service requirements as if one flow is slowed down, e.g., after a loss, it has a chance to catch up later due to the higher priority it will be granted by the scheduler.

In addition to enforcing fairness, LAS also has the appealing property that it favors short flows at the expense of long flows, which in an Internet context, means that interactive applications like mail, web browsing are favored at the expense of long bulk transfers, typically p2p file downloads[1] [5].

To address the specific problem of TCP ack streams that compete with TCP data streams over the shared wireless medium, we propose a new flavour of LAS called LASACK that enforces fairness between TCP responsive data streams and non responsive TCP ack streams.

Contributions of this paper are:

- We investigate the relative performance of FIFO, LAS and LASACK under a variety of scenarios. We consider both the case of long bulk transfers and the case of a mix of short and long bulk transfers that constitute a realistic

[1]We do not focus here on the specific case of multimedia transfers, e.g. Skype or MSN transfers, that might require specific care and different flavors of LAS [4].

user traffic wokload. This is in contrast with previous works that in general focus on long term transfers only [3, 6, 7].

- We show how LAS and LASACK enable TCP to rip the full benefit of the resources available at the IP layer, e.g., by reducing the loss rate of connections.
- We demonstrate the ability of LAS to enforce fairness between long lived flows and to favor short flows with a reasonnable penalty for the longest flows. More importantly, we show that the performance of the short flows degrades gracefully with an increasing load. Even in overload, which is not a rare event in WLANs, LAS and LASACK manage to preserve the interactivity (i.e. a small response time for the short flows) as compared to FIFO whose performance significantly degrades.[2]
- We investigate the use of single server queuing models to model the response time at the access point. Application of queuing models in WLAN context bears a number of difficulties. First, in a WLAN, the server represents in fact a shared medium. Second, choosing the capacity of the server is a non trivial task. For the latter issue, we propose different methods to assess the IP level capacity of 802.11 networks subject to TCP traffic.

2 Related Work

In [3], the authors focus on the case of an 802.11 wireless LAN with simultaneous uploads and downloads to and from the Internet and relate the observed unfairness (the uploading flow taking advantage over all downloads) to the buffer size at the base station. They propose to enforce fairness by adjusting the advertised window of TCP connections. The authors however acknowledge that their solution is not scalable as it requires to passively estimate the RTT of each connections to adjust the advertised window, which is a non trivial task. In [6], the authors propose a new scheduling policy that allows the transmission of bursts of frames where the burst size is a function of the channel failures experienced by a client. Their objective is to enforce short term fairness among competing TCP flows. Note that their technique requires to modify the 802.11 MAC protocol. In constrast, we rely on scheduling techniques deployed at the IP layer with no modification of the underlying protocol stack.

In the present work, we propose to use LAS, a size-based scheduling policy, in a wireless context. Size-based scheduling has already been proposed in a wireless context in [8]. The considered size-based policy is not LAS but the shortest job first (SJF) policy. The objective of the authors in [8] is to apply this policy at the

[2]Note that since buffer sizes are finite and we do consider finite sized flows, the response time does not ramp up infinitely as in an M/G/1 queue with a non preemptive discipline like FIFO or PS in overload conditions.

packet level to ensure fairness between short and long packets and more generally to favor multimedia applications. In contrast, we use LAS to solve fairness issues at the TCP level and we thus apply it at the flow level.

3 The Least Attained Service Policy

LAS is a size-based scheduling policy. It has been initially proposed and studied in the context of time-sharing computers in the late 60s [9]. Under LAS priority is given to the job that has received the least amount of service. In case of ties, jobs share the server in a round-robin manner. A salient feature of LAS is that it has no internal parameter to tune.

In [5], LAS has been studied in the context of a packet network like the Internet, where it is extended to incorporate a buffer management policy. The resulting policy that we refer to as LAS in the remaining of the paper works as follows. Upon reception, a packet is assigned a priority which is inversely proportional to the number of bytes sent so far by the corresponding connection (the first packet of a new connection thus has maximum priority). If ever the queue is full upon the arrival of a new packet, this packet is assigned its priority, inserted in the queue and the packet with the lowest priority is discarded.

The results on LAS relevant in the context of this paper are the following. In [5], LAS has been proved to improve the performance of most flows except a small fraction of the largest ones as compared to FIFO if ever the flow size distribution is highly skewed, i.e., a clear minority of the largest flows (say less than 5%) convey most of the bytes. Internet traffic has often been observed to exhibit highly skewed distributions [10]. The coefficient of variation of a distribution, which is the ratio of the standard deviation to the mean of the distribution, is often used to assess the skweness of a distribution. Typical values observed in the Internet range in the interval $[5, 10]$, e.g. [11].

LAS has also been proved to interact nicely with TCP by protecting flows in their slow-start phase in [12] and to solve classical unfairness situations: UDP vs. TCP or TCP flows with different RTTs [4].

The implementation of LAS can be performed at the flow or at the connection level. In order not to penalize applications that generate long connections but send at low rate, e.g, a routing process between two machines that sends keep-alive messages at a low rate most of the time, it seems wiser to implement LAS at a flow level. This is the option we take in the rest of this paper. The overhead of LAS as compared to FIFO is that per flow statistics must be kept. It should however be an affordable task for an access point that should not service a large number of simultaneous connections.

3.1 Accounting for non responsive TCP ACK streams

The focus of this paper is on TCP flows only. This assumption is justified by the predominance of TCP as a transport layer for both client/server and peer-to-peer applications. Unsuprisingly, even in the case of large (packet) wireless networks, TCP bears most of the traffic [1, 2].

WLANs constitute shared mediums with relatively low capacity[3]. While TCP aims at avoiding to exhaust network resources (by backing off after losses for instance), TCP ack streams can constitute a threat for the data traffic as it not subject to any congestion control algorithm. The unresponsiveness of TCP ack flows, combined with the cumulative acknowledgement strategies that mitigates the loss of acknowledgement, is at the root of the problem identified in [3].

The legacy LAS policy does not provide any specific solution to TCP ack streams. It even worsens the problem identified in [3] by giving a high priority to TCP level ACK packets since their size is in general small (piggy-backed packets are an exception), i.e. the attained service of TCP ack streams is in general smaller than the attained service of any TCP packet streams.

We propose and evaluate in this paper a new flavour of LAS, called LASACK that handles TCP ack streams in a specific way. Specifically, LASACK assigns to a TCP ack a priority equal to the number of bytes acknowledged by this packet. We demonstrate in this paper that this simple trick allows to enforce fairness between TCP ack and TCP data streams in WLANs.

statistics, it can not pretend to solve the hidden/exposed node problem.

4 The case of persistent flows

In this section, we evaluate the fairness of FIFO, LAS and LASACK for the case of long lived TCP transfers.

The network configuration that we consider throughout the paper consists of 10 wired stations and 10 wireless hosts serviced by a single point. The protocol used in the wireless part is 802.11b, with RTS/CTS disabled. The 10 wireless stations are at the same physical distance of the access point and in line of sight of each other. The 10 wired stations are connected to a router with an output rate 10 times larger than its input rate, which means that the output queue never builds up. The bottleneck, if any, is thus the access point.

In [3], the authors underscore a fundamentally unfair behavior of the FIFO scheduling policy that results in the uploading flows obtaining a larger share of the network capacity than the downloading flows. In addition, the authors provide a simple analytical model to estimate the ratio of the download to the upload throughputs through the ratio of congestion windows. In general, for small buffer sizes at the access point, the congestion window of the downloading flows is con-

[3]We are talking here about actual and not claimed capacity - see Section 5.2 for a discussion of this issue in the case of 802.11b networks.

strained both by the number of downloading flows and the buffer size at the access point. In contrast, the congestion window of the uploads is in general equal to the advertised window of the receiver. The model provides accurate results for buffer size values (at the access point) larger than 10 MSS. For smaller values, the behavior of the ratio is more chaotic and the ratio of congestion window overestimates the throughput ratio. A key observation that explains the simplicity of the model is that the RTTs of the downloads are equal to the RTTs of the uploads for all buffer size values. We made a similar observation, as can be seen in Figure 1.

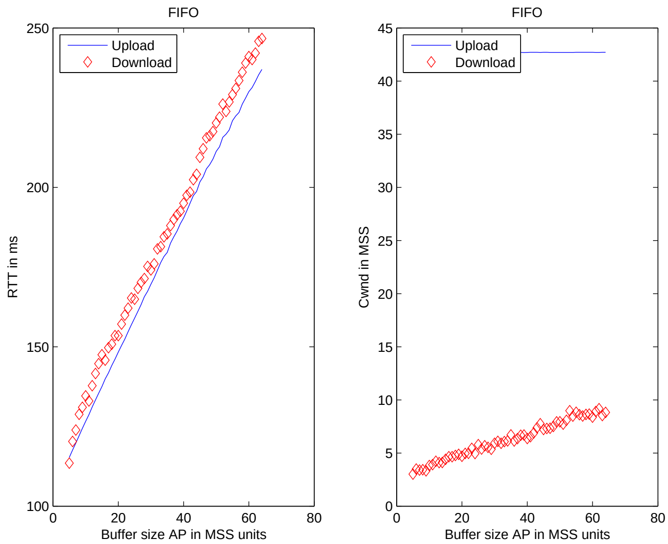

Figure 1: Average RTTs (left) and Cwnd(right) w.r.t. buffer size at the Access point,under the FIFO policy

We evaluated the model in [3] using simulations in Qualnet 3.9.5. It turns out to be accurate, even for buffer sizes smaller than 10 MSS, as can be seen from Figure 2. We suspect that the reason why the model is inaccurate for small buffer size in [3]is that buffer sizes in ns-2 (as used in [3]) are expressed in packets, irrespectively of the packet size, whereas in Qualnet, the buffer size is expressed in bytes. As a consequence, the competition between the acknowledgment packets from the uploads and the data packets is milder in Qualnet. However, the fundamental question is not which simulator performs the best, but what is the typical size of buffers in access points. Indeed, in [13], the authors investigate the same scenario (single AP with a mix of uploads and downloads) using Linux based access points, with large buffer sizes (above 100 MSS), and do not observe any unfairness. Note that this observation is in line with the results from Figure 2 where we observe that for increasing buffer sizes, the throughputs ratio decreases. In practice, it turns out that typical access point buffer sizes is often equivalent to 20 to 50 MSS packets. Hence, the range of values used in [3] makes more sense. In addition, we note that while the results in [13] confirm that the observed unfairness vanishes with increasing buffer sizes, inflating the buffer size at the access point can be detrimental to short flows. This was not observed in [13] as only persistent flows where considered. Furthermore, the extent to which the buffer size needs to be stretched to ensure fairness is a function of the number of simultaneously competing flows. It is thus difficult to dimension a priori the buffer.

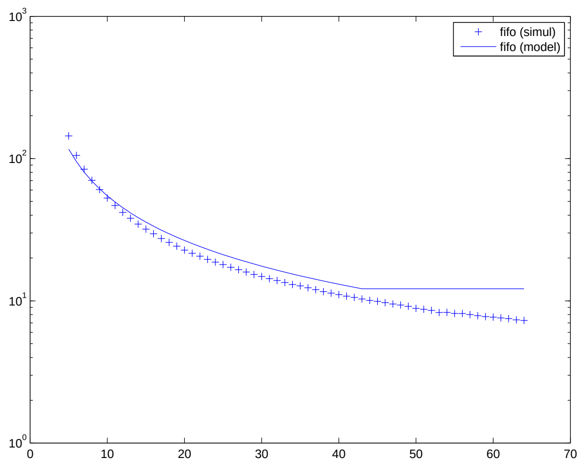

Figure 2: Ratio of upload to download throughputs w.r.t. buffer size at the Access point, under the FIFO policy

Let us now focus on the behavior of LAS and LASACK, for the same network setting as above. Figure 3 depicts the ratio of the long term throughput of the downloading and uploading flows.

We first observe from Figure 3 that the legacy LAS policy is less fair than FIFO, as the uploads obtain throughputs consistently two orders of magnitude larger than downloads. The explanation behind this observation is simple: TCP ack flows of the downloads consistently have the highest priority at the access point and lock out the data packets from the downloads. On the other hand, LASACK enforces a perfect fairness (ratio of 1) for all the buffer sizes, as it assigns a priority to the ack streams of the uploads that is a function of the amount of data uploaded so far. We further checked, see Figure 4, that fairness was not obtained at the expense of performance as the aggregate throughputs under LASACK is larger than the ones of FIFO and LAS. In an attempt to better understand the modus operandi of LAS and LASACK, we have computed the average RTTs and average congestion windows, both for the uploads and the downloads, as a function of the buffer size at the access point. Those two metrics are represented in Figures 5 and 6. We observe from those figures that the size-based scheduling policies impact both the RTT and the congestion windows of the flow. As a consequence, the derivation of a forecasting model similar the one presented in [3] for FIFO above is not straightforward for

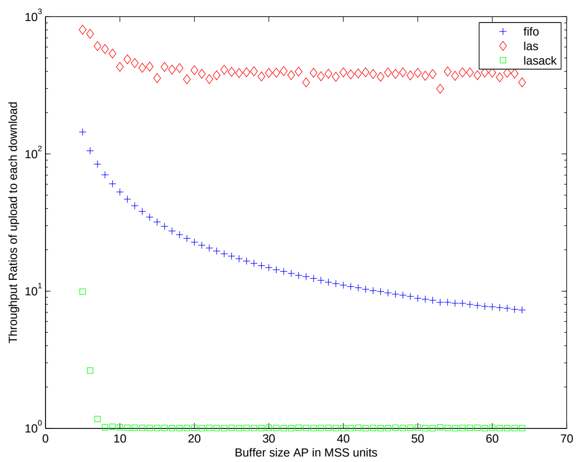

Figure 3: Ratio of upload to download throughputs w.r.t. buffer size at the Access point

the case of LAS and LASACK. We leave this issue for future work and focus now on more realistic workloads.

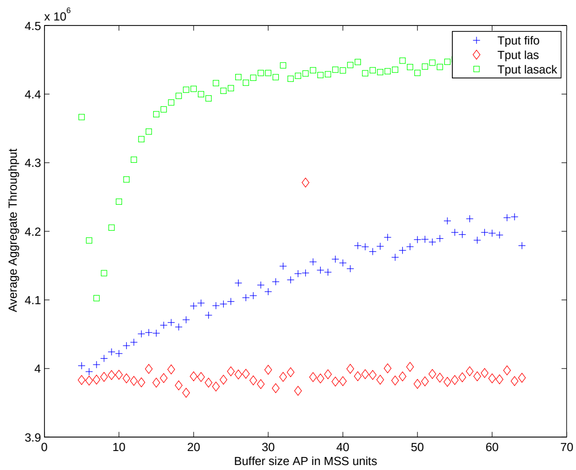

Figure 4: Aggregate throughputs w.r.t. buffer size at the Access point

5 Evaluating operational performance using realistic workloads

In this section, we introduce the methodology we use to evaluate the performance of FIFO, LAS and LASACK. We first introduce our worload model. We

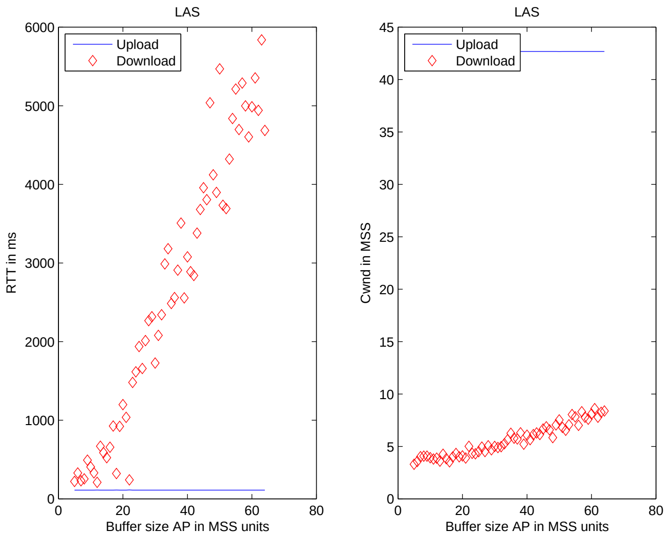

Figure 5: RTTs (left) and Cwnd(right) w.r.t. buffer size at the Access point, under LAS

next present a technique to assess the IP level bandwidth of a 802.11b network for the specific scenarios we focus on. Eventually, we discuss the use of queuing models to estimate the response time at the access point.

5.1 Traffic workload characteristics

In a single access point scenario with static wireless stations, the key characteristics for the flow level traffic are the flow arrival process and the flow duration distribution. [1,2] present seminal contributions in the field of WLAN traffic modeling. In those two papers, several datasets corresponding to large scale WLAN networks comprising several hundreds APs and several thousands users are considered. In both studies, the authors observe that the distribution of flow durations and flow sizes are heavy tailed. In [1], a biPareto distribution is proposed to model the flow sizes while in [2], several candidate distributions are considered: Weibull, Lognormal, Extreme-value, Pareto.

As for the flow inter-arrival times, the authors in [1,2] observe that this process is stationnary at small time scales (say half an hour to one hour periods), and more bursty than the often used Poisson process. Specifically, they propose to use a Weibull [2] or a BiPareto [1] distribution to model the flow inter-arrival process at small time scales, while they modulate the parameters at a larger (day) time scale.

In this paper, we rely on a Poisson process to model the flow level arrival process. This allows us to compare the simulation results with the theoretical results provided by analysis of the M/G/1/PS and M/G/1/LAS queues. In addition, while the Poisson process is less bursty than a renewall process with Weibull inter-arrival

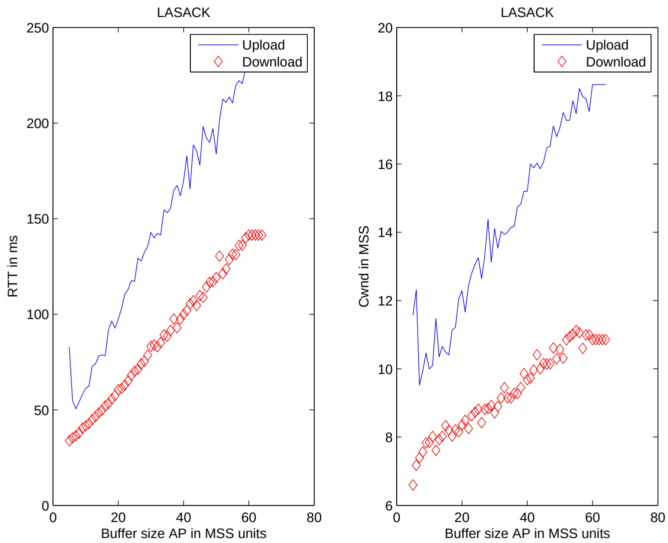

Figure 6: RTTs (left) and Cwnd(right) w.r.t. buffer size at the Access point, under LASACK

time distribution, we can still rely on increasing the load to stress the system and observe the performance of the different scheduling strategies.

As for the flow size distribution, we used three different distributions: Bounded Pareto, Lognormal and Exponential. Those distributions are flexible enough to investigate different degree of variability. Note that as we study the system at small time scale (say 20 to 30 min), we freeze the parameters of the distribution for the duration of each simulation.

5.1.1 Flow size distribution

In this section, we discuss the choice of the parameters of the flow size distributions used in the simulations and for the queueing models. Distribution of flow sizes (or equivalently service requirements) observed in the Internet are almost consistently characterized by a wide variability. A simple metric to characterize the variability of the distribution is the coefficient of variation (CoV), which the ratio of the standard deviation to the mean of the distribution. The CoV expresses the variability of the distribitution in mean units. constant A first question to address is which CoV we do need to consider. Our starting point is the work in [2] where the authors report the estimated μ and σ parameters for the Lognormal distribution they use to model flow sizes. They observe that μ ranges from 7.37 to 8.5, while σ ranges between 0.98 and 1.92. Figure 7 depicts the corresponding CoV and mean values. We do observe from Figure 7 that CoV values lie between 1.5 and 6. This is in line with what has been observed in the Internet, e.g., [14]. We will consider CoV values between 0.62 and 5.5 in this paper.

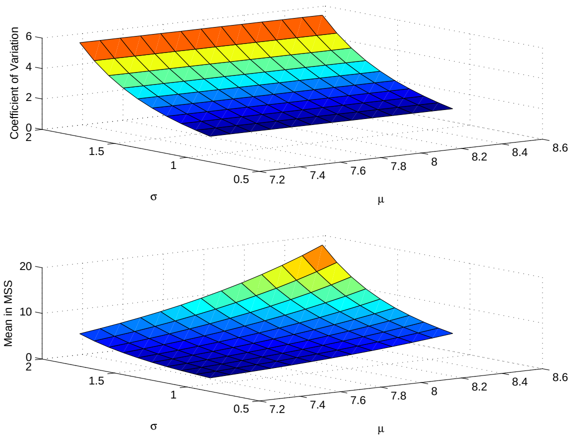

Figure 7: CoV and mean value for a Lognormal distribution for μ and σ parameters.

Another candiate distribution is the bounded Pareto distribution, commonly denoted by $BP(k, P, \alpha)$, where k and P are the minimum and the maximum job sizes and α is the exponent of the power law. The pdf of the Pareto is given as:

$$f(x) = \frac{\alpha k^{\alpha}}{1 - (k/P)^{\alpha}} x^{-\alpha-1}, \qquad k \leq x \leq P, 0 \leq \alpha \leq 2 \tag{1}$$

We consider in this paper a minimum file size of 4.5 Kbytes, i.e. 3 MSS with an MSS value of 1.5 Kbytes. The maximum job size we consider is 20 Mbytes, which surely constitutes an upper bound of what a user might download using a WLAN access. With this choice of k and p, we are left with a single parameter, α to tune both the CoV and the mean of the distribution. Figure 8 depicts the CoV and mean values for α that varies between 1 and 2. We do observe that CoV values can not be arbitrarily chosen. In general, the observed mean is around 10 packets, which corresponds in our case to CoV values between 2 and 6.

5.2 Wireless network capacity

A specific problem related to performance studies in wireless networks is to assess the IP level capacity of the wireless channel. In the scenarios we are looking at, we have one access point and a set of fixed and homogeneous wireless clients. The wireless MAC protocol is 802.11b, with RTS/CTS disabled (stations are in line of sight of each other) and a MAC level bandwidth of 11 Mbits/s. The DCF access method is used. In Section 6, we investigate scenarios where bulk transfers are operated between servers on the wired part of the network and wireless clients. In Section 7, we will have traffic flowing in both directions.

Let us first focus on the case of downloads only (Section 6). In this scenario, bulk transfers are performed using TCP with delayed ack turned on. We are in a

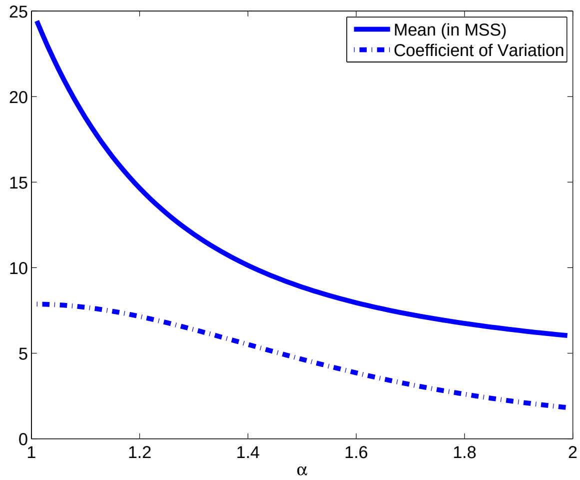

Figure 8: CoV and mean value for a Bounded Pareto distribution for $k = 4.5$ kbytes, $P = 20$ Mbytes

situation where the access point delivers TCP data packets to the wireless stations while the wireless stations reply with TCP ack packets for every two correctly and in order TCP data packets they receive. While the 802.11 protocol ensures a fair access to the medium to all wireless nodes (clients or access point), the typical (idealized) traffic pattern, neglecting losses, on the wireless channel should consist of two TCP data packets sent from the access point and one TCP ack packet sent from one wireless client. The idea behind this pattern is that if we observe less than two TCP data packets per cycle, then there would not be anymore TCP acks to send at some point as they are triggered by the TCP data packets. The operation at the MAC layer corresponding to this ideal situation is depicted in Figure 9.

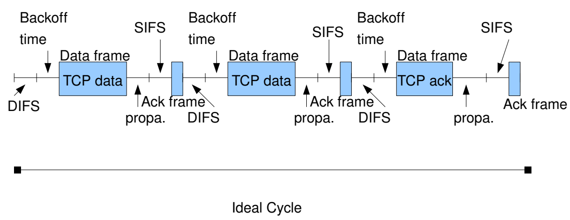

Figure 9: Ideal cycle of data exchanges on the wireless channel for download traffic only scenario

Based on Figure 9, we can estimate the duration T_{down} of a cycle. Let us adopt the following notations:

- $A = (DIFS + SIFS)\tau + Pre + PHY$ sums the bits in the guard bands, preamble and physical header, τ is the slot time, i.e. $20\mu s$ for 802.11b;

- l_a the size of a MAC level acknowledgment;

- h the MAC level header of a data frame;

- r the MAC rate, i.e. 11 Mbits/s in our case;

- c is the number of slots in the back-off time.

- d_{prop} is the propagation delay;

- $Sack$ the size of a TCP ack packet including the IP header, i.e. 40 bytes.

We obtain the following formula for T_{down}:

$$T_{down} = 3(A + \frac{l_a}{r}) + 6d_{prop} + 3c\tau + \frac{2MSS + Sack}{r} \qquad (2)$$

The maximum IP level bandwidth for the access point is $\frac{2MSS}{T_{down}}$. The IP level bandwidth is a function of the level of contention on the wireless medium. For a first collision, c is chosen uniformly between 1 and $Cw = Cwmin = 31$ slots. Cw doubles upon consecutive collisions until reaching 1023. We plot in Figure 10 ('down' curve) the estimated IP level bandwidth seen by the access point as a function of c where c is chosen in the middle of a contention window for all possible cotention window size values (from 31 to 1023). We can observe from Figure 10 that the IP level bandwidth decays rapidly with the level of contention.

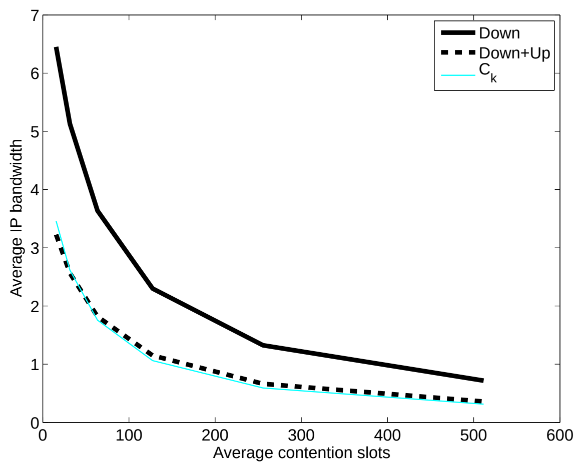

Figure 10: Estimated IP level bandwidth at the access point for the cycle of Figure 9

In the second type of scenarios we investigate, data servers might be either on the wired or on the wireless part of the network. Estimating the IP level capacity of the access point becomes much more complex in this situation because the pattern of TCP packets exchanged on the wireless medium heavily depends on the way resources of the access point (buffer and server) are shared among downloading and uploading traffic, as exemplified by Figure 3. We can however perform a reasoning similar to the one made for the download only case: an idealized cycle for those scenarios corresponds to the sending of two TCP data packets and one TCP ack packet for each direction of the traffic. Note that we assumed that the traffic intensity is similar in the upload and download directions. The duration of the corresponding cycle is $2T_{down}$, where T_{down} is computed using Equation (2). Assuming this ideal scenario, the download and upload traffic would each perceive an IP level bandwidth of $\frac{2MSS}{2T_{down}}$. This corresponds to the 'down+up' curve in Figure 10.

Note that the IP bandwidth values obtained in this section constitute upper bounds to operational values, that will depend on additional factors related to the TCP and MAC layers. TCP will alter the ideal cycle depicted above when, for instance, the TCP sender emits SYN packets or when the receiver sends one ACK for every data packet whenever misordering is detected. As for the MAC layer, it can alter the ideal cycles in many ways. For the download case scenarios for instance, the MAC layer can alter the ideal cycle by giving to the wireless stations the opportunity to send more than one TCP ack per cycle. This is possible since the wireless stations have in theory the same probability to access the medium as the access point. We will further discuss in Section 6.1.1 a method to assess the IP level bandwidth for a given experiment.

5.3 Queuing models

One of the objective of this work is to test whether simple queuing models can predict the response time at the access point for TCP flows in a WLAN environment. We consider 2 queueing models in this paper: the M/G/1/PS queue and the M/G/1/LAS, where G is a distribution with finite mean and variance and M stands for memoryless distribution of the interarrival times (i.e., arrivals form a Poisson process). For wired networks, the M/G/1/PS queuing model has been shown to capture the dynamic of TCP flows with homogeneous RTTs sharing a FIFO router, while the M/G/1/LAS queue turns out to accuratly model the behavior of a LAS router [4]. The use of those models bears a number of assumptions/challenges:

- What those queuing models allow to assess is the response time at the queue of the access point, i.e. the total time spent by the packets of a connection in the queue at the access point. This is not the same as the response time of the flow which is the time elapsed between the sending of the first packet and of the last packet of a connection at the sender side.
- A single queue model (as opposed to a network of queues) can be legitimate when there is a single and shared bottleneck in the path of all connections. This should be the case for WLANs where the wireless part constitutes the

bottleneck of the connections. However, even if the wireless section of the path constitutes the only existing bottleneck, multiple queues can build up, either at the access point or the wireless stations, which might affect the accuracy of the queueing models.

- We consider infinite queues for the queuing models. We do not consider losses because of the difficulty to model the TCP behavior when it experiences losses.

Let the average job arrival rate be λ. Let $f(x)$ be the probability density function of the service requirement G, and $F(x) \triangleq \int_0^x f(t)dt$, the cumulative distribution function. We further denote the survivor function of G as $F^c(x) \triangleq 1 - F(x)$. We define $m_n(x)$ as $m_n(x) \triangleq \int_0^x t^n f(t)dt$. Thus $m_1 \triangleq m_1(\infty)$ is the mean and $m_2 \triangleq m_2(\infty)$ is the second moment of the job size distribution. The load of jobs with size less than or equal to x is given as $\rho(x) \triangleq \lambda \int_0^x tf(t)dt$, and $\rho \triangleq \rho(\infty)$ is the total load in the system.

We are interested in the conditional response time $E[T|G = x]$ ($T(x)$ in short), i.e., the individual response time for each flow size [4] x. The expression for the conditional response time for PS is:

$$T(x)_{PS} = \frac{x}{1-\rho} \tag{3}$$

Conditional response time for $M/G/1/LAS$ is given by the following formula ([15] p. 172):

$$\begin{aligned} T(x)_{LAS} &= \frac{\lambda(m_2(x) + x^2 F^c(x))}{2(1-\rho(x) - \lambda x F^c(x))^2} \\ &\quad + \frac{x}{1-\rho(x) - \lambda x F^c(x)} \end{aligned} \tag{4}$$

5.3.1 M/G/1/LASACK

The LASACK policy differs from the legacy LAS policy whenever upload and download data streams compete with each other to access the wireless medium. We assume in our simulations and for the queuing models we consider that the distributions of both upload and download streams service requirements are similar. The load of the upstream flows is denoted as λ_U and the load of the download streams as λ_D. Under normal circumstances, we can reasonably expect that $\lambda_D \geq \lambda_U$.

Under LASACK, priority of ack streams of the upload flows is altered in such a way that an ack packet is treated by the scheduler like a data packet belonging

[4]The distribution G accounts for the service requirement (in second) of the client TCP connections which is related to the flow size (in bytes) X by $G = X/C$, where C is the capacity of the server. Note that we assign a specific distribution - Bounded Pareto, Lognormal, Exponential - to X and not to G.

to a stream that sent a number of bytes equal to the acknowledgement number carried by the ack. Otherwise, LASACK is similar to LAS. The response time in a LAS queue is the truncated moments $\hat{m}_1(x)$ and $\hat{m}_2(x)$ of G (See [15]). They are expressed as:

$$\hat{m}_i(x) = \int_0^x yf(y)dy + x^i F^c(x) \tag{5}$$

At the buffer of the access point, we have data flows whose service requirement follows a distribution G and ack flows whose service requirement follows the scaled distribution $\frac{Sack}{2MSS}G$ since on average for every two MSS packet there is one TCP ACK of size $Sack$. In addition, data flows arrive at rate λ_D and ack flows arrive at rate λ_U. Hence, we obtain for the first two truncated moments:

$$\begin{aligned}
\hat{m}_1(x) &= \Big(\frac{\lambda_D}{\lambda_D + \lambda_U} + \frac{S_{ack}}{2MSS}\frac{\lambda_U}{\lambda_D + \lambda_U}\Big) \times \\
&\quad \Big(\int_0^x yf(y)dy + xF^c(x)\Big) \qquad (6) \\
\hat{m}_2(x) &= \Big(\frac{\lambda_D}{\lambda_D + \lambda_U} + \Big(\frac{S_{ack}}{2MSS}\Big)^2 \frac{\lambda_U}{\lambda_D + \lambda_U}\Big) \times \qquad (7) \\
&\quad \Big(\int_0^x y^2 f(y)dy + x^2 F^c(x)\Big)
\end{aligned}$$

We obtain the following formula for the conditional response time of the downloading flows:

$$\begin{aligned}
T(x)_{LASACK} &= \frac{(\lambda_D + \lambda_U)\hat{m}_2(x)}{2(1 - (\lambda_D + \lambda_U)\hat{m}_1(x))^2} \\
&\quad + \frac{x}{1 - (\lambda_D + \lambda_U)\hat{m}_1(x)} \qquad (8)
\end{aligned}$$

We can observe from Equation (8) that the conditional response time under LASACK is similar to the response time of a LAS system with only download traffic of intensity λ_D as in general $\lambda_D + \frac{S_{ack}}{2MSS}\lambda_U \simeq \lambda_D$ and $\lambda_D + \Big(\frac{S_{ack}}{2MSS}\Big)^2 \lambda_U \simeq \lambda_D$ (since S_{ack} is equal to 40 bytes, MSS is equal to 1500 bytes in general and $\lambda_U < 1$). Note however that the IP level capacity perceived by the downloads is afftected by the upload traffic intensity.

As for the upload flows, their data streams are indirectly controlled by the LASACK scheduler when it schedules their TCP acks. The net result for those flows should be that their response time is close to the one of downloading flows.

6 Download traffic only

In this section, we investigate the performance of FIFO and LAS for the case where data packets flow from wired servers to the wireless stations. Note that LAS is identical to LASACK in the case of downloads only. We set the buffer size at the access point to 50 Kbytes (corresponding to approximately 33 MSS, with an MSS of 1.5Kbytes), in line with the discussion in Section 4.

We first focus on a baseline scenario, with a Bounded Pareto flow size distribution with parameters $(k, p, \alpha) = (3MSS, 2 \times 10^7 MSS, 1.4)$. Those parameters correspond to a CoV of approximately 5.5. We vary the input offered load so as to stress the wireless bottleneck.

In Figure 11, we present bar plots of measured load for an offered (input) load that increases from 1 to 7 Mbit/s - FIFO experiments are on the left, LAS experiments on the right. We distinguish for each experiment between unique (first transmission) and retransmitted bytes. We do observe that while FIFO achieves higher total rates, the rates of unique data is equivalent under FIFO and LAS for each experiment. This result underscores the ability of LAS to enable TCP to rip the full benefit of the network resources. In contrast, FIFO lets TCP exhaust the network ressources and do a lot of retransmissions.

Before analyzing further the results, we need to define when the network is in underload or not. Since the buffer sizes are finite, we cannot expect to observe unbounded response times. The increase in the number of retransmitted bytes as well as the stagnation of the number of unique bytes in Figure 11 indicate that saturation has been reached. To further quantify when one switches from underload to overload, we computed the number of active connections over time for some simulations of LAS and FIFO lasting 1000 seconds. We do observe that the number of active connections starts ramping up for input loads larger than 4 Mbit/s for both LAS and FIFO. However, since LAS favors the short connections that represent the majority of connections, we expect that when overload is reached, the number of active connections under FIFO ramps up at a faster rate than under LAS. This is confirmed by Figure 12 where we plot the ratio over time of the number of active connections under FIFO and LAS. After 1000 second of simulation, FIFO is left with 60 to 100% more unfinished connections than LAS.

of active flows over time is similar constant becomes LAS. load.

Let us now investigate in more detail the behavior of LAS and FIFO during each experiment. Due to space constraint, we focus on two input load values, 2 and 5 Mbit/s. The first input load value corresponds to an underload situation, while the second one corresponds to an overload scenario. For each offered load value, we present 4 graphs that depict:

- The conditional response time, computed as the time elpased between the first SYN packet and the last ACK (in response to a FYN-ACK) sent by the TCP sender. Those quantities are extracted from the tcpdump traces collected by Qualnet during the simulation on the TCP sender side. Tcpdump files are further processed (using tcptrace) to extract response times as well as the quantities used below;

- The conditional retransmission ratio computed as the number of bytes retransmitted divided by the total number of unique bytes transmitted by the

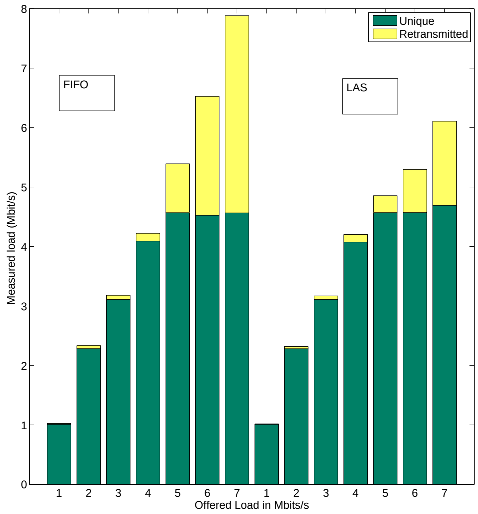

Figure 11: Average measured load during simulation

connection. The number of retransmission is an upper bound on the actual loss rate in the network as TCP might perform useless retransmissions.

- The conditional (time) weighted average of the congestion window per flow;
- The conditional average RTT per flow.

of flow file

We plot the above metrics w.r.t. the quantiles of the flow size distribution. The plots for an input load of 2 Mbit/s (resp. 5 Mbits/s) are presented in Figure 13 (resp. Figure 14).

Let us first focus on the response time. We do observe that the response time is smaller under LAS than under FIFO for the majority of flow sizes (using the quantile view of the data), under all load conditions. Of course, the largest flows pay a penalty in terms of larger response time under LAS for a load of 2 Mbits/s. Some of the largest flows do not even complete before the end of the simulation for an offered load of 5 Mbit/s under LAS.

More importantly, we do observe that under LAS, the majority of flows (say up the 50-th quantile) are only marginally affected by an increase in the offered load (from 2 to 5 Mbits/s) as compared to what happens under FIFO, where the response time of the smallest flow is increased by a factor of 100. This means that the user interactivity (measured as the response time of the smaller flows) is relatively independent of the load conditions under LAS as compared to FIFO. This is definitely a key feature of LAS. We will see in Section 6.1 that this property holds even if we change the distribution of the flow sizes.

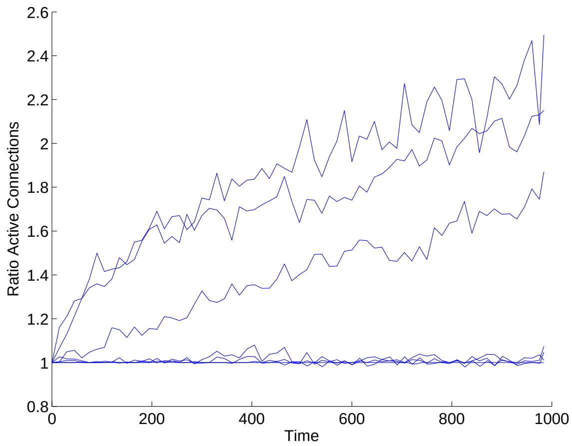

Figure 12: Ratios of number of active connections over time for FIFO and LAS for input loads between 1 Mbits/s (lower curve) and 7 Mbits/s (higher curve)

A closer look at the retransmission ratio and the round trip times sheds lights on the observations made on the response time. We do observe that under LAS the smallest flows experience little to no losses in combination with small round trip times. Those observations result from the behavior of the LAS scheduler. Note that in contrast to the case of persistent flows (Section 4), LAS can not “tune” the congestion window of the small values which is very low, 1 to 3 MSS, in general.

6.1 Queueing Models for the Response time

We now investigate the accuracy of the queuing models proposed in Section 5.3. For each of the experiment performed in the previous section, we computed the response time at the access point queue per flow. To do so, we first computed the one way delay between the source and the receiver for each TCP packet. We next remove the latency associated to the wired part for the groups of packets sent by the TCP sender (TCP sends packets in flights). To determine the size of those groups, we assume that TCP uses the slow start algorithm (delayed ack is used, hence the window is not double each time) until reaching the advertised window that corresponds to 65 Kbytes, i.e., 43 MSS packets in our simulations. Note that we do not account for retransmissions. The propagation delay between any sender (on the wired part of the path) and the access point is 2 ms. In addition, senders are connected with 10 Mbit/s access link. Thus, it takes 1.2 ms to emit a packet of 1.5 Kbytes (the MSS in our simulation).

In Figures 16 to 18, we present results for three different offered load values between 1 and 3 Mbit/s. For each experiment, we plot the conditional response time at the access point for each quantile of the flow size distribution as obtained from the simulation. We further plot the corresponding results for the queuing

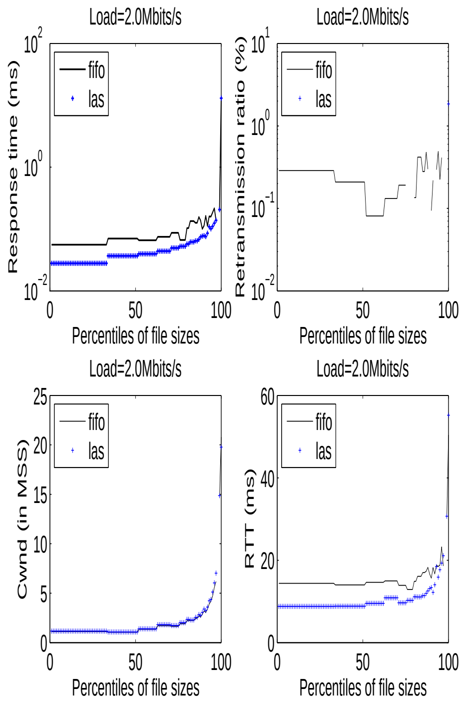

Figure 13: Download traffic only - Input load of 2 Mbit/s

models. We defer to Section 6.1.1 the discussion on how we choose the IP level bandwidth at the access point that we use in the queuing model. From Figures 16 to 18, we observe that:

- For the case of FIFO, the M/G/1/PS model tends to slightly underestimate the observed response time at the access point at low loads. At high load, close to 1, we observe, on the contrary an overestimation of the response time. The reason behind those observations is the depency of the reponse time of each flow on the actual load ($T(x)_{PS} = \frac{x}{1-\rho}$ for a flow of size x). Hence, any small error on the estimation of the capacity (hence on the actual load) can dramatically impact the fitting. This problem is particularly accurate in the case of wireless networks - see Section 6.1.1.

- For the case of LAS, we observe that the M/G/1/LAS model is quite accurate in general at all load values. for buffer

One of the objective of this work is to show that the M/G/1/LAS queue is a good model for an LAS access point. The results presented above hold for a specific distribution, namely a Bounded Pareto with parameters $(k, p, \alpha) = (3MSS, 2 \times 10^7 MSS, 1.4)$). We further evaluated the accuracy of the M/G/1/LAS model for

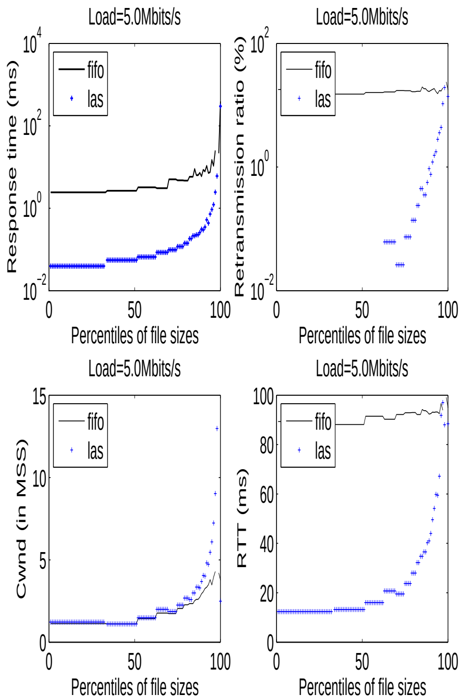

Figure 14: Download traffic only - Input load of 5 Mbit/s

another Bounded Pareto distribution with a CoV of 3, shifted exponential distributions[5] with CoV of 0.62 and 0.75 and some LogNormal distributions with CoV of 3 but different mean values. In all cases, we observed (graphs are not presented due to space constraints) that the M/G/1/LAS model is good at predicting the reponse time at the queue of an LAS access point. As for the M/G/1/PS model, the results turn out to be dependent on the exact distribution with sometimes some under estimations and sometimes some over estimations of the queue response time.

6.1.1 Setting the capacity

We have obtained in Section 5.2 upper bounds on the IP level capacity of a wireless network for the case of TCP transfers. As illustrated by Figure 10, those bounds are function of the level of contention on the wireless medium. We propose here a technique to assess the IP level capacity for operational cases. Our technique relies on the M/G/1/LAS formula that we discussed in the previous section. Let us consider a distribution with a minimum flow size k. Then, from Equation (4), we obtain that the response time of flows of size k is given by the following formula:

[5]The density function of a shifted exponential with minimum value k is $f(x) = \lambda e^{-\lambda(x-k)}$ if $x \geq k$ and 0 otherwise. A straightword computation shows that its coefficient of variation is below 1 due to the shift of the mean.

$$T(k)_{LAS} = \frac{\lambda(k/C)^2}{2(1-\lambda k/C)^2} + \frac{k/C}{1-\lambda k/C} \quad (9)$$

The simple form of Equation (9) follows from the fact that $m(k) = \int_k^k uf(u)du = 0$, $m_2(k) = \int_k^k u^2 f(u)du = 0$ and $F^c(k) = 1$. Note that Equation (9) is identical for any distribution with a minimum flow size k. Using Equation (9), we can obtain an estimator $\hat{C}$ of C by resolving the corresponding quadratic equation. The closed formula for $\hat{C}$ is given by Equation (10).

$$\hat{C} = \frac{k(2\lambda T(k)_{LAS} + 1) + \sqrt{k^2(2\lambda T(k)_{LAS} + 1)}}{2T(k)_{LAS}} \quad (10)$$

$\hat{C}$ values for a Bounded Pareto distribution with parameters $(k, p, \alpha) = (3MSS, 2\times 10^7 MSS, 1.4)$ correspond to the first line of results reported in Table 1.

We do observe that $\hat{C}$ values are below what was predicted in Section 5.2. There is not necessarily a contradiction here since the values predicted in Section 5.2 are upper bounds to the actual values. We can follow a reasoning similar to the one conducted in Section 5.2 for the specific case of flows of size k to explain the low values of $\hat{C}$. We assume that those flows experience no losses, which is in general the case with LAS as exemplified in Figures 13 and 14. For those flows, the TCP sender will emit one SYN, 3 data packets of size MSS and one final ACK (the FIN flag is set in the last data packet). The TCP receiver will emit one SYN-ACK, one FIN-ACK and 2 pure ACKs. Hence, we can form an ideal cycle consisting of the emission of 3 TCP data packets and 5 TCP control packets of 40 bytes each. The total duration T^k_{down} of the cycle is computed similarly to T_{down} in Equation (2). The corresponding IP level capacity is $\frac{3MSS}{T^k_{down}}$, which corresponds to the curve labeled C_k in Figure 10. We do observe that the values of C_k are in line with our estimation $\hat{C}$ since the initial value of C_k - for a contention window equal to 32 slots - is 3.5 Mbits/s.

A second striking observation concerning $\hat{C}$ is that it is increasing with the offered load. This is in apparent contradiction with the fact that an higher load should result in a higher contention on the wireless, and thus, as predicted in Figure 10, the IP level bandwidth should decrease. We however observed that the main effect of an increase in the offered load is the increase of the queue drops at the buffer of the access point and not an increase in the number of MAC retransmissions on the wireless medium. To illustrate this latter observation, we plot in Figure 15 the number of MAC retransmissions, the number of TCP retransmissions and the number of drops at the access point queue for the different input loads. We do observe from Figure 15 that the dominant effect of workload increase is the increase of queue drops rather than MAC retransmissions.

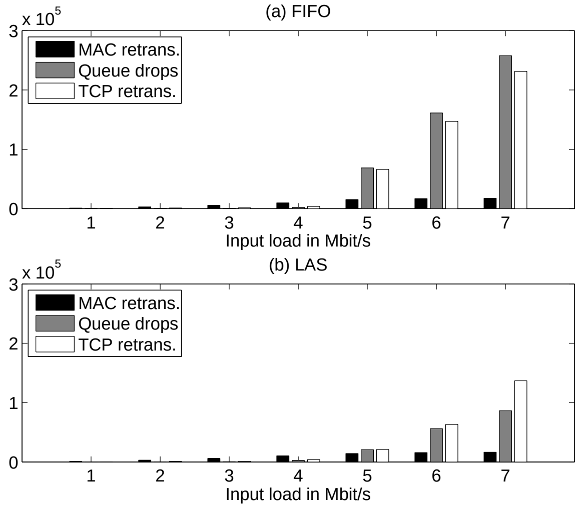

Figure 15: Impact of workload increase on the different layers of the protocol stack

			Load (Mbits/s)						
Dist	Mean	CoV	1	2	3	4	5	6	7
BP	9.67	5.5	3.0	3.1	3.2	3.3	3.4	3.6	3.9
S-EXP	9.67	0.75	3.0	3.2	3.2	3.3	3.3	3.6	3.9
LN	9.67	3	3.0	3.1	3.2	3.3	3.4	3.6	3.9
BP	6.67	3	3.2	3.4	3.6	3.7	3.9	4.1	4.3
S-EXP	6.67	0.62	3.2	3.5	3.6	3.7	3.9	4.1	4.3
LN	6.67	3	3.2	3.5	3.7	3.7	4	4.2	4.3

Table 1: Estimated IP level Capacity

We also checked the dependency of the estimator on the distribution of flow sizes. To do so, we considered a Bounded Pareto distribution with a coefficient of variation of 3, a shifthed exponential distribution with a minimum value of 3.

The results for all those distributions are reported in Table 1. We do observe that the results in Table 1 are independent of the distribution but depends on its mean. This is because for a smaller mean, the arrival rate λ, which appears in the $\hat{C}$ formula (Equation (10)) is increased so as to maintain offered loads between 1 and 7 Mbits/s for all distributions. While the latter argument explains the indirect relation between a change in the mean of the distribution and a change in $\hat{C}$, it does not tell us why $\hat{C}$ should increase when λ increases. Indeed, an increase in λ might be compensated by an increase in the response time $T(k)_{LAS}$ in Equation (10). We hypothetize that $\hat{C}$ increases because when the mean decreases, the fraction of flows of size k exactly increases, which means that those flows have to compete with less packets from other flows and thus obtain a larger share of the IP level bandwidth of the scheduler.

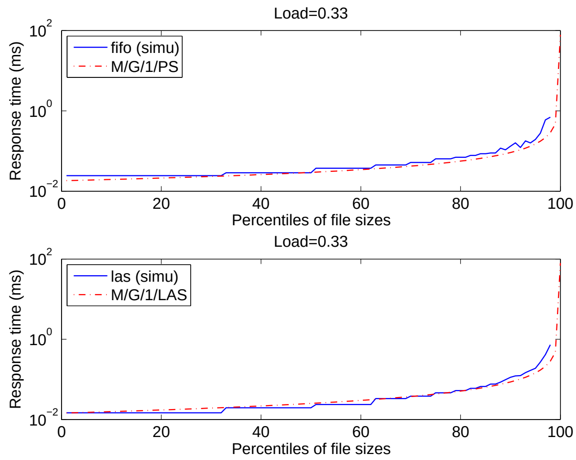

Figure 16: Response time at the access point for an offered load of 1 Mbits/s. Bounded Pareto distribution - CoV=5.5

7 Mix of upload and download traffic

We consider in this section the case where TCP connections convey data from wireless hosts to wired servers or from wired servers to wireless hosts. Following the notations introduced in Section 5.3.1, we denote λ_u and λ_d the arrival rates of TCP connections in each direction. We present here results for the case where $\frac{\lambda_u}{\lambda_d} = 0.5$. In addition, we assume that the distribution of file sizes are similar for both upload and download traffic. Overall, this means that the total upload traffic represents 33.3% of the whole traffic while the download traffic amounts for 66.6% of the total traffic. We present in Figures 19 and 20 the results for a total (download+upload) input load of 2 and 5 Mbit/s. The distribution of file size is a Bounded Pareto distribution with parameters $(k, p, \alpha) = (3MSS, 2 \times 10^7 MSS, 1.73)$. Those parameters correspond to a CoV of approximately 3.

The response time graphs in Figures 19 and 20 reveals that LASACK does a good job at keeping the response time low for most of the uploading and downloading flows as compared to FIFO. To tune the response times, LASACK controls the round trip times of the connections. As in the case of downloads only, there is less flexibility concerning the congestion windows that are very small for most flows in general.

Concerning the relative performance of downloads and uploads under FIFO and LASACK, we observe that FIFO favours uploads at the expense of downloads. This is inline with the results of Section 4 and also the results in [3]. As for LASACK, this is opposite as the uploads have slighlty higher response times than downloads. We do not have a clear explanation for this phenomenon at the moment. This might be related to the fact that the LASACK scheduler treats TCP acks from the uploads equivalently to TCP data packets from the downloads. This might be

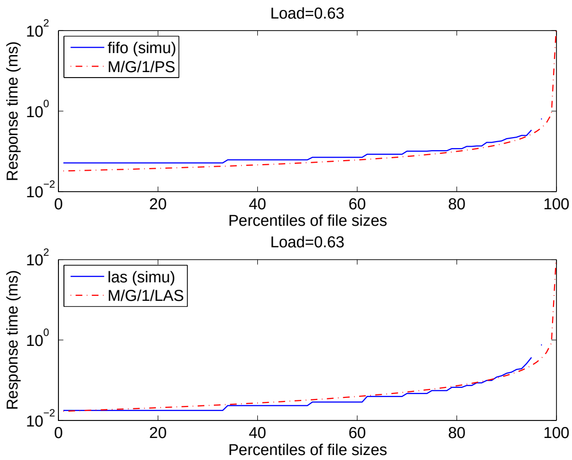

Figure 17: Response time at the access point for an offered load of 2 Mbits/s. Bounded Pareto distribution - CoV=5.5

slighlty unfair for the TCP acks which have to compete with packets whose service requirement is $\frac{S_{ack}}{MSS}$ larger, even if the unfairness is minimal as compared to what we observe for FIFO.

Overall, what is important to note is that LASACK is able to maintain low response times for the short flows when the global increases, while the short flows under FIFO experience almost a 10 fold increase (from 2 to 5 Mbits/s for instance).

7.1 Queueing Models for the Response time

We want to test in this Section the accuracy of the queuing model developped in Section 5.3.1. Our focus here is on the response time at the access point queue for the download traffic. A necessary first step to compare simulations and queuing results is to assign a capacity to the scheduler. We use the technique developped in Section 6.1.1 (Equation (10) with λ replaced by $\lambda_D + \frac{S_{ack}}{MSS}\lambda_U$). Using this technique, we obtain a capacity of 2.65 Mbit/s for both an input load of 3 and 5 Mbit/s (Figures 19 and 20). To check whether those results are in line with the approach presented in Section 5.2, we have to form the idle cycle for flows of size k when 2/3 of the traffic consists of downloads and 1/3 consists of uploads. Let T_{down} be the time necessary to transmit 3 MSS and the 5 control packet on the wireless medium for the download traffic. Obviously, the time T_{up} to transmit the same number of packets for the upload traffic is the same, i.e., $T_{down} = T_{up}$. To determine the IP level capacity C_{mix} from the point of view of the download flows, we have to account for the relative intensity of the traffic flowing in both directions. We simply obtain that to transmit 3 MSS, download flows have to wait for T_{down} two thirds of the time, and $T_{down} + T_{up}$ one third of the time. We eventually obtain the following formula for the IP level capacity C_{mix}:

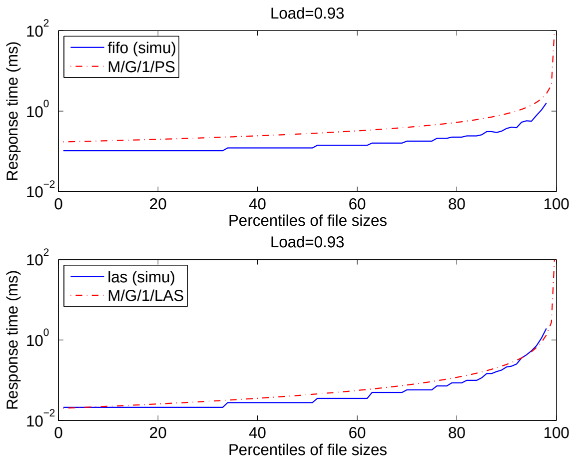

Figure 18: Response time at the access point for an offered load of 3 Mbits/s. Bounded Pareto distribution - CoV=5.5

$$C_{mix} = \frac{3MSS}{\frac{T_{down}}{3} + \frac{2(T_{down}+T_{up})}{3}} = \frac{9MSS}{4T_{down}} \tag{11}$$

Comparing C_{mix} with C_k (IP level capacity for downloads only - see Section 6.1.1), we obtain that $C_{mix} = 3/4C_k$. We should obtain that $C_{mix} \leq 2.6$ Mbit/s, since the largest value of C_k, which is obtained for the minimum contention window $Cw = 32$ is 3.5 Mbits/s. Those results are thus in line with our empirical approach to determine the IP level capacity. As for the reason why C_{mix} does not decrease with an increasing offered load, it is similar to the one obtained in Section 6: while contention increases, the dominant factor that explains the retransmitted bytes by TCP are the drops at the queue of the access point. However, since the smallest flows (of siwe k) do not experience any losses in our simulations when the offered load increases, the IP level capacity C_{mix} that they perceive is not affected by an increase in the offered load.

We present in Figures 21 and 22 the response time at the access point queue for the M/G/1/LASACK model with a server capacity of 2.6 Mbits/s. We provide results only for the LASACK policy as with a capacity estimate of 2.65 Mbits/s the M/G/1/PS queue is in overload. This is the case also for LASACK with the noticeable exception that under LASACK performance degrades gracefully when the load is above 1: these are first the largest jobs that have an infinite response time while the smallest still have a finite response time[6]. Especially, we can observe in Figure 22 that the M/G/1/LASACK model does not deliver a result for the last quantiles of the distribution as those largest flows experience an infinite response time in the queuing theory.

[6] see [5] for details about LAS in overload - results can be easily extended to the case of LASACK

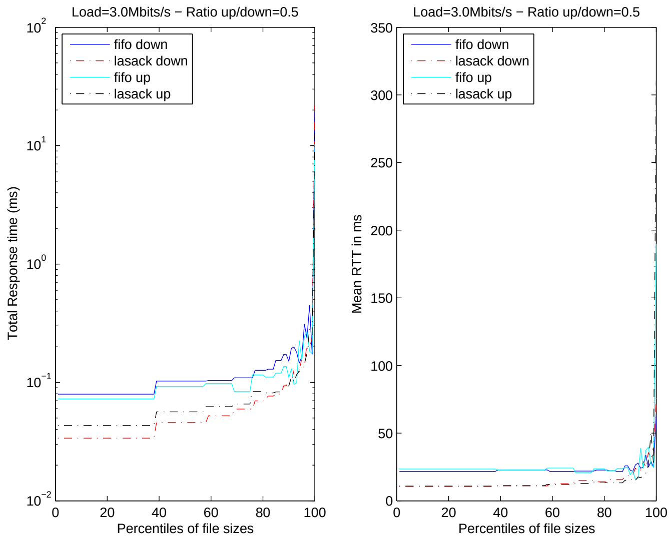

Figure 19: LASACK vs FIFO for a total input load of 3 Mbits/s

Overall, we observe that the M/G/1/LASACK queueing models is able to capture the behavior of an LASACK router. We checked, similarly to what we did in Section 6 that the M/G/1/LASACK offers good results under a variety of distributions.

8 Conclusion

In this paper, we have investigated the use of size-based scheduling policies to both inforce fairness and provide small response times to most of the flows in a WLAN. We first considered persistent TCP flows and show that the original LAS policy is able to improve the fairness when data flows in a single direction but degrades the performance as compared to FIFO when data is flowing in both directions. We demonstrate that a specific cross layer variant of LAS called LASACK, that inspects the acknowledgement number of TCP, improves fairness under all scenarios.

We next considered the most practical case of a mix of short and large flows, which constitutes a realistic model of users traffic. We back our choice of flow size distribution with some measurement studies performed on large WLANs. We further evaluated the use of single server queuing models to assess the response time at the bottleneck, i.e. the access point. This last point is challenging because the wireless medium is shared and one might question if a single queue model is enough to model an actual system. In addition, queuing models require to specify the capacity of the wireless medium seen at the IP layer. We presented a number of heuristics to estimate a priori the IP level capacity of the wireless channel and in ad-

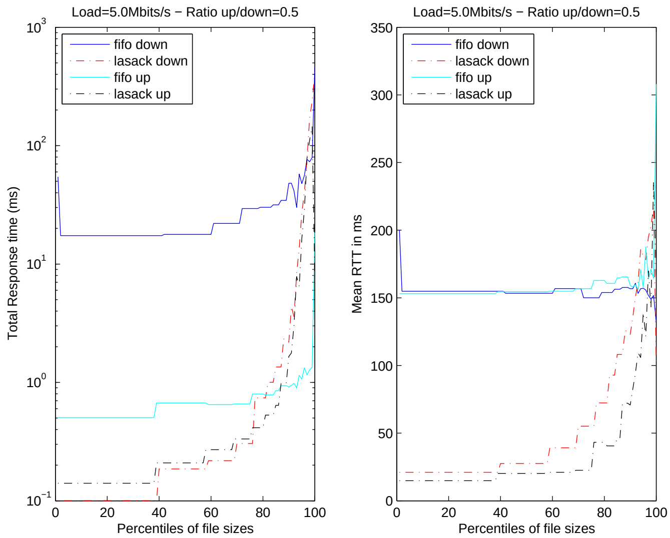

Figure 20: LASACK vs FIFO for a total input load of 5 Mbits/s

dition, to directly assess the capacity from the simulations. The preliminary results we have obtained on the capacity estimation are encouraging. Also, the queuing models for LAS and LASACK turn out to be accurate under many different input loads and flow size distributions.

From a networking point of view, LASACK fulfills its promises by improving the interactivity perceived by the users but most importantly by being resilient to load increase. The net result for a user is that only long transfers are penalized by a load increase; the longest of the long transfers being penalized first. Its short transfers, on the other hand, always receive the same quality of service (response time) from the network.

In terms of future work, many questions remain to be investigated. We focused on the conditional mean response time but the variance of the response time is also important. Also, our focus in this work was on TCP transfers only but more importantly, we were assuming that the applications on top of TCP were of elastic type. The next step is to consider the case of multimedia traffic flowing either over TCP or UDP.

The mobility of users might be also an important topic to focus on. More generally, variations of the channel quality due to the environement (exluding contention) needs also to be taken into account. To maintain the good results obtained with LASACK in the case of varying channel conditions, one might have to combine it with some mechanisms to improve fairness at the station level, e.g. [16]. Ultimeately, we would like to investigate the benefits of size-based scheduling policies in ad-hoc and wireless mesh networks where the performance of TCP flows are subject to wide variations in general.

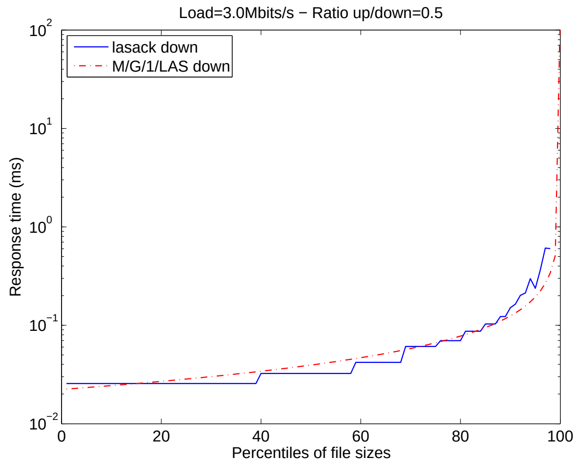

Figure 21: Response time of download traffic at the access point for an offered load of 3 Mbits/s

References

[1] F. Hernández-Campos, M. Karaliopoulos, M. Papadopouli, and H. Shen, "Spatio-temporal modeling of traffic workload in a campus wlan," in *WICON '06: Proceedings of the 2nd annual international workshop on Wireless internet*, 2006, p. 1.

[2] X. G. Meng, S. H. Y. Wong, Y. Yuan, and S. Lu, "Characterizing flows in large wireless data networks," in *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM Press, 2004, pp. 174–186.

[3] P. Sinha, S. Pilosof, R. Ramjee, D. Raz, and Y. Shavitt, "Understanding TCP fairness over wireless LAN," in *Proceedings Infocom 2003*, Apr. 2003.

[4] I. A. Rai, Urvoy-Keller, and E. W. Biersack, "Las scheduling approach to avoid bandwidth hogging in heterogeneous tcp networks," in *HSNMC'04*, July 2004.

[5] I. A. Rai, G. Urvoy-Keller, and E. W. Biersack, "Analysis of las scheduling for job size distributions with high variance," in *Proc. ACM SIGMETRICS*, June 2003, pp. 218–228.

[6] M. Bottigleliengo, C. Casetti, C. Chiasserini, and M. Meo, "Short-term fairness for tcp flows in 802.11b wlans," in *INFOCOM 2004*, vol. 2, 2004, pp. 1383–1392 vol.2.

[7] R. Bruno, M. Conti, and E. Gregori, "Analytical modeling of tcp clients in wi-fi hot spot networks." in *NETWORKING*, 2004, pp. 626–637.

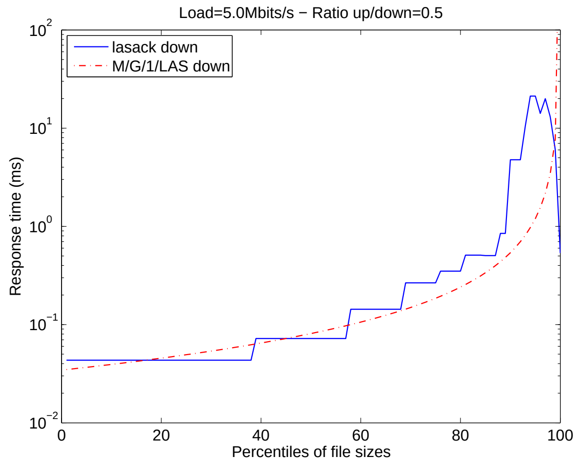

Figure 22: Response time of download traffic at the access point for an offered load of 5 Mbits/s

[8] S. Pollin, A. Motamedi, A. Bahai, F. Catthoor, and L. Van der Perre, "Delay improvement of ieee 802.11 distributed coordination function using size-based scheduling," vol. 5, 2005, pp. 3484–3488 Vol. 5.

[9] L. E. Schrage, "The queue m/g/1 with feedback to lower priority queues," *Management Science*, vol. 13, no. 7, pp. 466–474, 1967.

[10] M. Crovella and A. Bestavros, "Self-similarity in world wide web traffic: Evidence and possible causes," *IEEE/ACM Transactions on Networking*, pp. 835–846, December 1997.

[11] L. H. Gomes, C. Cazita, J. M. Almeida, V. Almeida, and J. Wagner Meira, "Characterizing a spam traffic," in *IMC '04: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*. New York, NY, USA: ACM Press, 2004, pp. 356–369.

[12] I. Rai, E. W. Biersack, and G. Urvoy-Keller, "Size-based scheduling to improve the performance of short TCP flows," *IEEE Network Magazine*, 2005. [Online]. Available: http://www.eurecom.fr/~btroup/BPublished/las_net.pdf

[13] R. Bruno, M. Conti, and E. Gregori, "Performance modelling and measurements of tcp transfer throughput in 802.11-based wlan," in *MSWiM '06: Proceedings of the 9th ACM international symposium on Modeling analysis and simulation of wireless and mobile systems*, 2006, pp. 4–11.

[14] J. Charzinski, "Observed performance of elastic internet applications," *Computer Communication*, vol. 26, no. 8, pp. 914–925, 2003. [Online].

Available: http://www.compscipreprints.com/comp/Preprint/char/20020716/1/compcomm02-col.pdf

[15] L. Kleinrock, *Queuing Systems, Volume II: Computer Applications*. New York: Wiley, 1976.

[16] M. Heusse, F. Rousseau, R. Guillier, and A. Duda, "Idle sense: an optimal access method for high throughput and fairness in rate diverse wireless lans," in *SIGCOMM '05*, 2005, pp. 121–132.