WiFox: Scaling WiFi Performance for Large Audience Environments

Arpit Gupta*, Jeongki Min*, Injong Rhee
Dept. of Computer Science, North Carolina State University
{agupta13, jkmin, rhee}@ncsu.edu

ABSTRACT

WiFi-based wireless LANs (WLANs) are widely used for Internet access. They were designed such that an Access Points (AP) serves few associated clients with symmetric uplink/downlink traffic patterns. Usage of WiFi hotspots in locations such as airports and large conventions frequently experience poor performance in terms of downlink goodput and responsiveness. We study the various factors responsible for this performance degradation. We analyse and emulate a large conference network environment on our testbed with 45 nodes. We find that presence of asymmetry between the uplink/downlink traffic results in backlogged packets at WiFi Access Point's (AP's) transmission queue and subsequent packet losses. This traffic asymmetry results in maximum performance loss for such an environment along with degradation due to rate diversity, fairness and TCP behaviour. We propose our solution *WiFox*, which (1) adaptively prioritizes AP's channel access over competing STAs avoiding traffic asymmetry (2) provides a fairness framework alleviating the problem of performance loss due to rate-diversity/fairness and (3) avoids degradation due to TCP behaviour. We demonstrate that *WiFox* not only improves downlink goodput by 400-700 % but also reduces request's average response time by 30-40 %.

Categories and Subject Descriptors

C.2 [**Computer-Communication Networks**]: Local and Wide-Area Networks–Access schemes

General Terms

Design, Experimentation, Measurement, Performance

Keywords

WiFi, large audience environments, goodput, traffic asymmetry

*Authors' names in alphabetical order

CoNEXT'12, December 10–13, 2012, Nice, France.

1. INTRODUCTION

WLANs are the most popular means of access to the Internet. The proliferation of mobile devices equipped with WiFi interfaces, such as smart phones, laptops, and personal mobile multimedia devices, has heightened this trend. The performance of WiFi hotspots serving locations such as large conventions and busy airports has been extremely poor. In such a setting, more than one WiFi access points (APs) provide wireless access to the Internet for many user devices (STAs). The followings are the commonly cited causes of this problem.

- **Contention and collision**. When many STAs are competing for channel resources using CSMA/CA, the overhead of handling high contentions, such as carrier sensing, back-off and collisions, can be very high.
- **Rate diversity and fairness.** In an access network, different STAs may have different channel conditions. WiFi typically adopts an automatic rate adaptation scheme where poorer channel STAs use lower-rate modulation schemes which has a side effect of occupying longer channel time for transmission of the same size packets. As poorer STAs will use more channel resources, the overall throughput is reduced.
- **Random losses and TCP performance.** TCP is the most commonly used transport protocol. But it treats all packet losses as congestion related and reduces its transmission rates for each packet loss. Unfortunately, wireless channels are prone to random packet losses unrelated to congestion. TCP throughput in WiFi can be low when many losses occur due to low channel quality and collisions.
- **Traffic asymmetry.** In an access network, a single AP serves all associated clients. Moreover our analysis of network traces [27, 28] shows that downlink (AP to STAs) traffic is much greater than uplink traffic (4-10 times), attributable to commonly used client-and-server based applications (e.g., web and email). This asymmetry of data traffic combined with IEEE 802.11 DCF providing equal opportunity for channel access to both APs and STAs results in congested APs and subsequent packet losses.

There have been many proposals to fix or ameliorate these problems. However, the existing solutions do not sufficiently address the problem. Major limitations of existing solutions are as follows.

First, the existing approaches [3, 36, 31, 14] to the performance optimization of WiFi are highly atomistic, focusing only on fixing one or a subset of these problems. In general, performance optimization must take a holistic approach with careful considerations of complex interactions among various control "knobs" of optimization. Tweaking an unsuitable combination of knobs may not bring sufficient performance improvement and may even negatively impact the performance.

Second, many existing solutions are not amenable to practical deployment because nearly all cases require changes in both APs and clients and sometimes also in the MAC layer coordination that can be realized only by modifying the firmware of wireless interface cards. While deploying a new (modified) AP is relatively easy in a hotspot, deploying client solutions is difficult because of the diversity of client devices. For practical and incremental deployability, a proposed modification must be limited to AP.

Third, many existing solutions are not tested in real networks with realistic network workloads. Most of them are based on simulation or theoretical analysis [18, 17, 9]. There is a significant gap among the results predicted from real network experiments, simulation and theoretical analysis. Therefore, it is hard to predict the actual performance of these proposed solutions.

All the above factors deter the deployment of these solutions in production networks. In this paper we demonstrate that solving the problem of traffic asymmetry results in maximum performance improvements for large audience environments. We find correlation between the presence of asymmetry in network traffic and instantaneous transmission queue at the WiFi AP and develop a *mechanism* where traffic asymmetry is inferred in real time, prioritizing the AP accordingly for channel resource access over competing STAs. For large audience environments, the prioritization of AP's traffic enables efficient realization of AP-only fairness solutions. The key contribution of our work is the empirical study of the performance implications of these solutions in order to optimize the performance of busy WiFi hotspots. To add realism to our results, we implemented our solutions in an off-the-shelf commercial IEEE 802.11g AP, constructed a real network testbed of 45 WiFi nodes and tested the performance of various optimization settings in network traffic loads emulating the traffic patterns captured from real traces [27, 28].

In the ensuing sections, we start with understanding the characteristics of wireless networks in Large Audience Environments (LAE) in section 2. We discuss the attributes of a desired solution based on these observations and in section 3 discuss design and implementations of our solution. Description of testbeds and experimental procedures are provided in section 4 and experimental results are provided in section 5. Later we discuss related works in section 6 and finally conclude our contributions in section 7.

2. TRAFFIC CHARACTERISTICS

In this section, we characterize the nature of WLAN traffic patterns for large audience environments. In past Maier et al. [20] analysed the traffic characteristics for residential broadband Internet traffic while Raghavendra et al. [23] and Rodrig et al. [27] analysed the same for large conference environments. Of the two Rodrig et al. [27] analysed network traces for SIGCOMM 2004 and reported presence of the asymmetric traffic patterns with around 80 % of total consisting of downlink traffic. To further confirm the similar trend in a relatively more contemporary data set, we analysed the SIGCOMM 2008 traces [28] which capture the WLAN traffic occurring during the event.

Figure 1a & 1b show the number of active STAs and aggregate network throughput measured each minute. Figure 1a shows the difference in the number of active clients sending uplink requests and receiving downlink responses. Figure 1b shows relatively low wireless throughput for such an environment. These two figures show that for large number of associated clients, many STAs couldn't receive any response from servers resulting in degraded throughput attributable mainly to the bottleneck for downlink traffic at the AP. We also observe that downlink throughput is significantly higher than uplink most of the time. Figure 1c shows the CDF of the ratio of the downlink traffic volume over the total traffic volume measured from all traces. More than 90% of the traces have the ratio greater than 50%. These figures clearly show the presence of high asymmetry between uplink and downlink traffic which is further quantified in Table 1.

		% of Protocol	% Out of Total
Downlink	TCP	91.2	83.4
	UDP	8.8	
Uplink	TCP	82.5	16.6
	UDP	17.5	

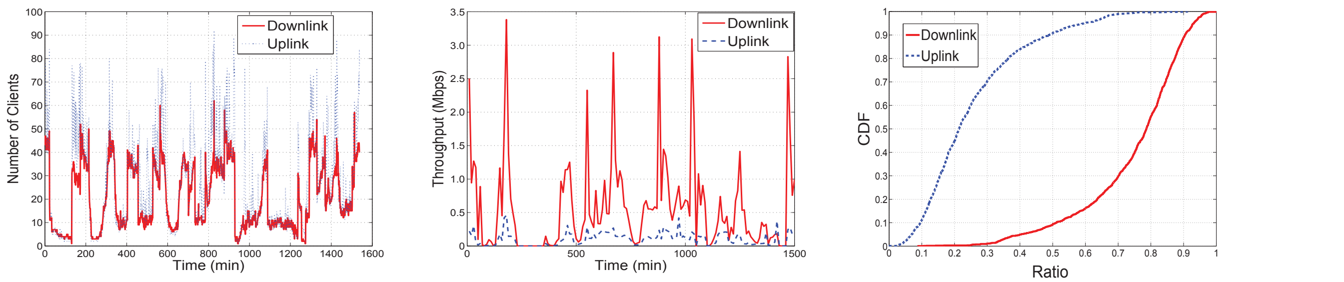
Table 1: Ratio of TCP and UDP of Downlink and Uplink Traffic in terms of number of bytes

Application Type	TCP (%)	UDP (%)
Web	56.54	0.00
IPSec	0.00	59.46
Email	12.99	0.00
Chat	0.59	0.00
Service Discovery	0.00	1.36
File Sharing	0.50	0.00
DNS	0.00	8.60
NetBIOS	0.72	9.44
Secure Shell	5.62	0.00
Streaming	0.69	0.00
Remote Desktop Service	0.42	0.00
Network Configuration	0.00	1.71
Others	21.93	19.43

Table 2: Types of applications in TCP and UDP traffic.

We classified the application types of the captured packets in Table 2, based on their port addresses. Among identified TCP packets, 57% of TCP traffic is Web traffic, 13% is email, and 6% is SSH. For UDP traffic, 60% is for IPSec NAT traffic, 10% is for NetBIOS, and 9% is for DNS. These data indicate that most of the TCP and UDP traffic are for client-and-server applications (Web, email, NAT, DNS, and NetBIOS) whose servers are running in the Internet.

Figure 2 shows the inter-arrival times for uplink UDP and TCP packets. They can be fit to exponential distributions using the maximum likelihood estimation. The average inter-arrival times are about 47 ms for TCP and 88



(a) (b) (c)

Figure 1: (a) Number of STAs sending Uplink/Downlink requests per minute (b) Downlink/Uplink throughput averaged over one minute (c) Distribution of Downlink/Uplink over total traffic volume ratio

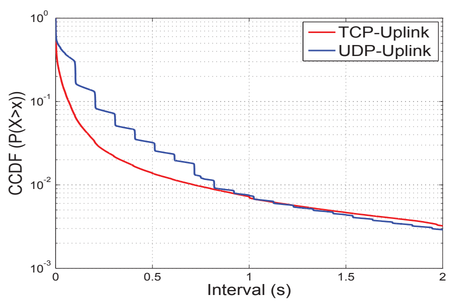


Figure 2: Distribution of Interval of TCP and UDP Uplink

ms for UDP. Note that these rates are conservative because not all STAs are associated with the measurement AP.

Rodrig et al. [27] illustrates that the ratio of STAs experiencing network failure over all STAs increases as the number of STAs increases in a wireless network. Further our analysis of traffic also validates the results obtained by Rodrig et al. [27]. We highlight two major points from our analysis.

- WLAN traffic is highly asymmetric in nature. On average, about 80% of the total traffic is downlink.
- The majority of the traffic (80 to 90%) is TCP. A majority of the TCP traffic ($\sim$70%) is from web and email, both known to generate heavy-tail traffic patterns [10, 29].

We leverage these results to design our testbed emulating such an environment. These network characteristics also guide the design choices of our solutions. It is evident from this analysis that the asymmetry in network traffic intensifies with the increase in the number of associated STAs. This results in performance bottleneck at the AP for downlink traffic and thus motivates the need for a mechanism to prioritize medium access for the AP over STAs under heavy traffic load. In next section we will discuss design and implementations of our solution.

3. DESIGN AND IMPLEMENTATION

Previously we discussed various factors that result in performance degradation for WIFi in LAE. In this section we will discuss how we tackled each of those problems. We describe our novel method for AP's adaptive priority control (APC) which solves the problem of traffic asymmetry. We then present our implementation of AP-only fairness framework which addresses the problem of unfair resource allocations due to rate diversity. We further discuss our implementations of TCP Proxy/ECN to resolve the problems of TCP performance for LAE. Combining all these solutions in an optimal manner we propose our solution called ***Wi***Fi ***Fo***r Large ***C***onference Environment (*WiFox*).

Class	CWmin	CWmax	AIFS	TXOPLimit
AP	1	5	1	64
STAs	5	10	N/A	N/A

Table 3: An example of parameter value setting enabling the highest priority channel access by the AP. TXOPLimit is denoted in terms of the number of 5 *us* slots. The default value of AIFS in IEEE 802.11e is 2. The settings for STAs is the default setting of IEEE 802.11g which uses DIFS instead of AIFS and does not define TXOP (so its value is set to 0). DIFS is much larger than AIFS.

3.1 Priority Control

The asymmetric nature of WLAN traffic as discussed in Section 2 causes congestion at the AP which becomes a bottleneck under heavy traffic load. The DCF-based IEEE 802.11 MAC is designed to give the AP the same opportunity to the wireless medium as the STAs in the basic service set even though the AP has the greatest amount of the traffic to transmit. Under heavy downlink load coupled with high contention from many active STAs, the AP cannot flush its traffic quickly, thus becoming a performance bottleneck and suffering a high rate of packet losses from both transmission queue overflow and collisions due to high contention. This motivates the need for a mechanism enabling a controlled preferential treatment to the overloaded AP over STAs for medium access.

We cannot give the AP high priority over STAs by default. It has an adverse effect on network performance: because the uplink traffic in the form of client requests from the STAs will be stifled, it will lead to a decreased downlink traffic which in light network load, can reduce the network goodput. Therefore a fine balance is required between uplink and downlink traffic in order to optimize network throughput.

We propose an APC scheme wherein the percentage of

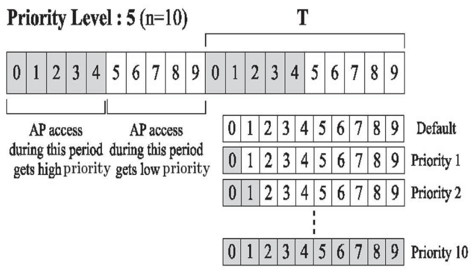


Figure 3: The illustration of the WiFox priority control model. For ease of illustration, the random k slots of high priority is chosen to be slot 0 to $k-1$ in each cycle of T.

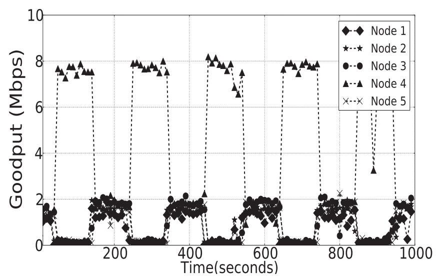


Figure 4: Effect of a priority setting on UDP Throughput. Node 4 runs with the HIGH priority setting specified in Table 3 while the other nodes run at the DEFAULT setting

downlink traffic being given priority is proportionately controlled based on the dynamic traffic load at the AP. This ensures that at low load the STAs get an equal opportunity as the AP to transmit requests and at higher loads the AP have a higher access priority proportional to the amount of downlink traffic. APC is designed in two steps. First, we deal with a priority model to define fine-grained MAC-level priority levels which are easy to control. Second, we develop an algorithm for adaptive control of priority levels that adjusts the channel access priority of the AP according to dynamic downlink traffic load.

3.1.1 Linear Scaling Priority Model

A common way to control the priority of channel access is to reduce the inter frame space (IFS) of WiFi and backoff duration values of each packet transmission so that the packet transmission starts before all other nodes start checking for the availability of the wireless medium. The following four parameters of IEEE 802.11e are commonly available from software WiFi drivers: (1) the minimum contention window size $2^{CW_{min}}$; (2) the maximum contention window size $2^{CW_{max}}$; (3) the Transmission Opportunity Limit (TXOPLimit), defined as the maximum duration that a node can transmit without contending for the wireless channel with Short Inter Frame Space (SIFS); and (4) inter frame spaces (DIFS or AIFS, PIFS and SIFS).

Striking good balance between uplink and downlink traffic in the presence of dynamically varying traffic load requires a fine-grain control of channel access priority. This requires (a) a set of fine-grained priority levels, and (b) an estimation of the impact of each level on the probability of channel access by the AP and STAs. But these requirements are very difficult to meet by simply adjusting the MAC parameters, especially because the impact of each particular parameter value, let alone their combinations, on the channel access probability of the AP and STAs is not well defined. For instance, given two value settings: (CW_{min} = 2, CW_{max} = 6, $TXOPLimint$ = 32, $AIFS$ = 3), and (3,5,0,2), which one has higher priority? If one has higher priority than the other, then how much do they differ in terms of channel access probability?

To address this problem, we take a novel approach that allows for a linear scaling of the access probability. The linear scaling permits convenient and accurate control of the channel access probability of the AP. We first define a setting of the MAC parameters that assigns the highest priority for the AP – when the AP competes for channel access with that setting, it wins the access most of the time. We call such a setting *HIGH*. The default value setting of these parameters is called *DEFAULT*. Table 3 shows one particular example of such a setting. Note that some of these parameters are defined only in IEEE 802.11e. However, this setting is compatible with STAs supporting any of IEEE 802.11 a/b/g/n. We only need the AP software to support IEEE 802.11e. The MADWIFI driver of WLAN provides the interfaces for controlling these parameters.

To get a set of fine-grained priority levels satisfying our requirements, we then divide the channel time of the AP into continuous intervals of time T. Each unit of T is further divided into n slots of duration T/n. If the AP has a priority level $k \leq n$, then k random slots out of the n slots within each T are *high priority slots* and the remaining slots are *low priority slots*. When the AP has a packet to transmit, if its current real-time indicates it is in a high priority slot, it accesses the channel with the HIGH setting; otherwise, it accesses the channel with the DEFAULT setting. Figure 3 illustrates this model. The random choice of the high priority slots with each cycle of T is intended to avoid collision among WiFox APs running in proximity. Because CW_{max} of HIGH is set to 5, even if multiple WiFox APs are competing, they have a room for random backoffs (up to 32 slots). With proper planning of LAE, a situation where many WiFox APs are competing in the same interference range can be avoided. Note that this model does not require time synchronization with STAs, nor with other APs. because it controls the access priority of the AP – the STAs simply use their default settings governed always by whatever IEEE 802.11 standard they are currently following.

This way, n fine grained priority levels can be obtained by controlling the number of slots (in each interval T) in which the AP receives very high priority. This scheme supports a nice linear scaling property where the average throughput of the high priority node increases in linear proportion to its priority level. This linear scaling makes it very convenient to design an adaptive control algorithm.

To verify that the parameter value setting in Table 3 increases the priority of packet transmissions correctly, we conducted a simple experiment consisting of 5 senders transmitting an Iperf UDP flow to one receiver. All nodes uses Netgear IEEE 802.11b/g card with Atheros chipset. We modified the MADWIFI driver in one of the senders to implement our priority model which is activated in alternating periods of 100 seconds. The other nodes always use the default IEEE 802.11 standard. We measure the throughput of each sender. Figure 4 shows the alternating periods of

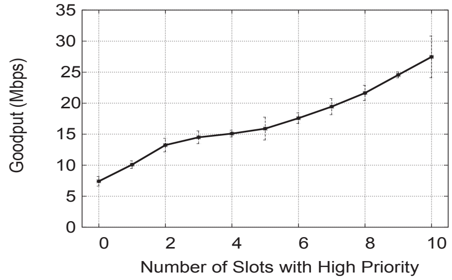


Figure 5: Linear scaling property of our priority model. The average throughput of the high priority node increases more or less linearly proportionally to its priority level

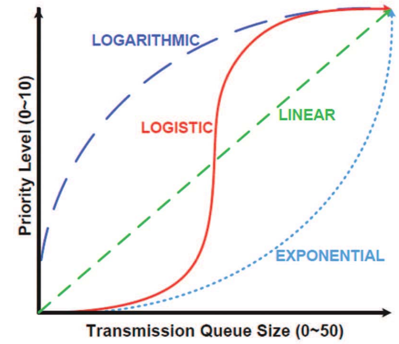


Figure 6: Four representative priority control functions.

100s where the high priority node (node 4) gets much higher throughput than the other nodes. In the remaining periods, the throughput of all nodes is comparable because all the nodes use the same default IEEE 802.11g setting.

To verify the linear scaling property of our priority model, we set n to 10 and T to 100ms, and vary the number of prioritized slots from 1 to 10 in the experiment taken in Figure 5. We measure the average throughput of node 4 for each priority level only during the intervals of prioritized accesses. Figure 5 shows the linear scaling property of our priority model being closely approximated in terms of the throughput achieved by the high priority node with various priority level settings.

3.1.2 Adaptive Priority Control

It is rational to assume that the provision of priority to the downlink traffic at the AP needs to be closely related to the dynamic traffic load at the AP. Dynamic downlink traffic load at the AP can be reliably estimated by the instantaneous transmission queue size of the AP where the maximum queue size is limited by an upper bound, Q_{max}. Therefore, we design adaptive priority control(APC) models determining the priority level depending on the transmission queue size at the AP for WiFox. Here, we map 10 priority levels into the slotted queue size whose maximum, Q_{max} is 50. We apply two intuitive criteria in designing APC models. First, APC models should have the lowest priority (e.g., no downlink) at zero queue size and the highest priority at max queue size. Second, the priority level of APC models should be monotonically increasing as the queue size increases. There may exist uncountably many models satisfying the criteria but amongst them, we choose 4 representative models which lead to totally different behaviors in the priority control and in the queue size variation. The models we choose are depicted in Fig. 6 and their characteristics are described below.

Logarithmic APC model (PC-LGA) provides a steep growth of the priority level for a small increase in the low queue size. Then, the growth speed diminishes as the queue size approaches the maximum. PC-LGA shows the most aggressive priority control among the 4 models as it maintains downlink priority unless the queue size becomes too low. PC-LGA serves more downlink traffic even under low queue size however this eventually tends to empty the queue because choking the data request in the uplink may result the lack of data arrival to the downlink queue of the AP in future.

Exponential APC model (PC-EXP) provides an exponential growth of the priority level at the high queue size. From a different point of view, it rapidly backs off the priority of the downlink when the queue size starts to reduce. It loses the chance of serving downlink traffic for low and medium queue size but this conservative back-off will bring about a large amount of data arrivals, hence the queue has a very low probability of becoming empty.

Linear APC model (PC-LIN) provides a non-special growth of the priority level. It balances the downlink and uplink proportionally. There will be no aggressiveness in serving downlink traffic nor rapid back-off.

Logistic APC model (PC-LGI) provides a combinational priority control of the exponential growth and the logarithmic reduction in the middle level of queue size. Thus, PC-LGI is aggressively serving downlink traffic at the high queue size and rapidly backs off when the queue size drops below a certain level. (e.g., half of Q_{max}) PC-LGI can be considered as a threshold policy in controlling priority since it determines only on and off according to the threshold value.

3.2 Fairness Control

WiFox integrates a fairness control with APC. It does not advocate one particular notion of fairness over another. Instead it offers a framework in which the system designer can plug in his own implementation of a control algorithm that best suits his needs.

WiFox can help realize the potential of AP-only fairness control. During the period of heavy downlink traffic, WiFox assigns a high priority level to AP, and thus, AP packets will always get high priority over uplink packets. Since the channel time will be consumed mostly by AP with its prioritized accesses, this ensures that channel time allocation asymptotically follows whatever notion of fairness the implemented control strives to accomplish.

WiFox offers a framework where the AP-only fairness algorithm is implemented as a kernel module of the AP which functions in the IP layer just above the MAC layer (where APC runs). The module contains a separate transmission queue for each active destination STA. It uses *Netfilter* architecture to capture outgoing packets using the *POST ROUTING* hook before they reach the MAC layer. If the captured packets are destined for the wireless interface, they are enqueued in their queues corresponding to their destinations. Queues are dynamically created and deleted on an as needed basis.

Figure 7 illustrates the architecture of the WiFox frame-

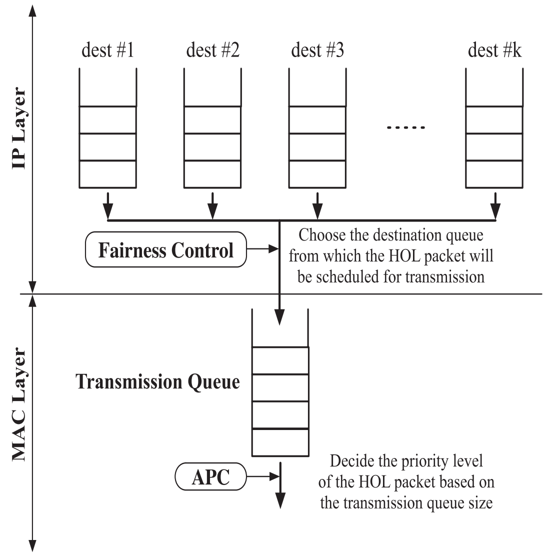


Figure 7: The architecture of the WiFox framework

work. The system designer can plug in his own fairness control algorithm here. For instance, time fairness can be implemented as follows. The kernel module maintains a channel occupancy time table of which entries maintain exponentially weighted moving averages of channel occupancy time for all destination queues it current holds. The module periodically monitors the instantaneous transmission queue size in the MAC layer. If space is available, the scheduler picks a non-empty queue with the minimum channel occupancy time and sends a packet from that queue to the MAC layer for transmission. If two or more queues have the same minimum channel occupancy time, it selects one in a round robin manner. The channel occupancy time of a transmitted packet is computed based on its size and the estimate of the current data rate. The channel occupancy times of all the queues are periodically updated by taking a moving average with the total channel occupancy time that their packets transmitted since the last update. It should be noted that our post-routing netfilter hooks filter out TCP data packets and queue them in appropriate destination queues and all other packets traverse directly to the AP's TxQ. This ensures that WiFox does not interferes with the network stack traversal of other packets and can easily support functionalities like pure link layer forwarding etc.

3.3 TCP Proxy/ECN

Apart from prioritization and fairness solutions, we will discuss briefly how we dealt with performance degradation due to *Random Losses and TCP Performance* under our design constraints. In the past, the problem of random losses interpreted as packet loss due to congestion has been solved using *Explicit Congestion Notifications (ECN)* [25]. To enable ECN, we modified AP's MAC driver and TCP stack to send congestion notification to the server with STA's acknowledgements when its instantaneous transmission queue (TxQ) exceeds a predetermined threshold. *TCP proxy* can further avoid performance degradation by bringing the server closer to the STA virtually and enabling faster reactions to its fluctuating network conditions [19]. TCP Proxy (Fig 8) is

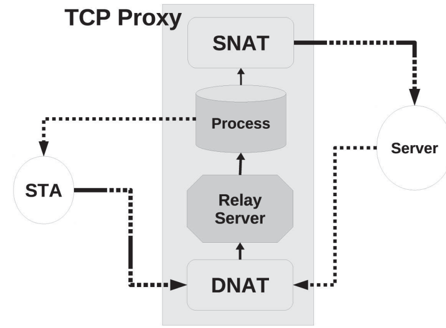


Figure 8: TCP Proxy implementation at AP

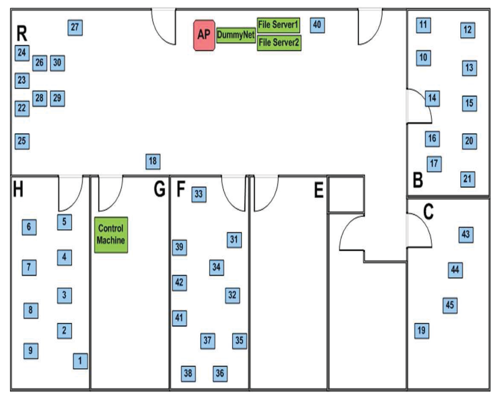


Figure 9: Placement of various nodes in our Testbed. Distance between AP and STAs ranges between 10-40 meters

implemented at AP using IPtables and NAT and it receives all requests from the clients as a virtual server and transmits new requests as a virtual client to real Internet servers. In next section we will discuss the experimental set-up under which we tested the performances of these proposed solutions.

4. EXPERIMENTAL SETUP

In this section we present our detailed experimental methodology for evaluating WiFox on our testbed. We will discuss how we built our testbed, emulated the traffic patterns similar to those of SIGCOMM 2008 traces [28] and conducted experiments to quantify performance gains of our solutions.

4.1 Testbed

Our experimental set-up consists of a testbed with multiple APs and 45 STAs deployed at our research lab (about 2600 sqft area). The architecture and layout of the testbed are shown in Figures 10 and 9 respectively. Scaling the layout from Figure 9, we can see that distance between STAs and AP ranges between 10-40 meters. The testbed consists of 24 STAs and 2 File Servers (FSs) of Type A and 21 STAs and an AP of Type B. Type A machines have an AMD Dual Core processor with 2 GB RAM, 160 GB Hard Disk, Netgear IEEE 802.11b/g wireless card with Atheros chipset and Debian OS with kernel version 2.6.27.12-1. Type B machines have a VIA processor with 1 GB RAM, 60 GB Hard Disk and the rest of the configuration is the same as Type A. It should be noted that machines we used as APs are not

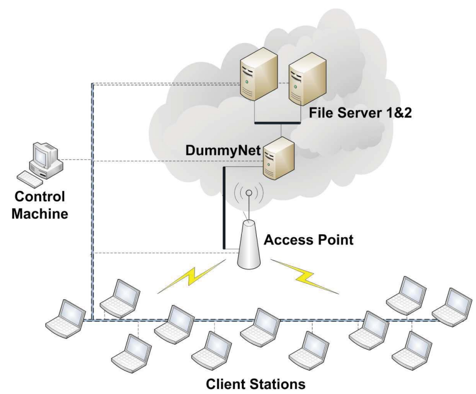
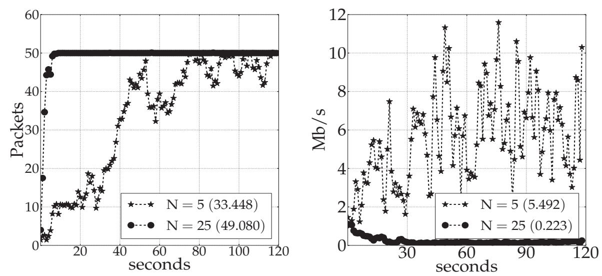


Figure 10: Test Bed Architecture



(a) TxQ Length (b) Goodput (down)

Figure 11: (a) We observe that traffic asymmetry is correlated with TxQ length (b) Downlink goodput for entire network drops significantly when number of associated clients increase from 5 to 25

computation intensive ones and are comparable to ones commercially used for such network environments. Also usage of netfilter hooks should not limit the scalability of our solution as they can be efficiently embedded with basic routing module itself for the APs.

The wireless interface card uses the MADWIFI [2] Linux Device Driver, an open source project under GPL. We have two File Servers (FSs) which act like remote servers over the Internet. All STAs can access these FSs through an AP. To give more realistic network delay between all STAs and FSs, we place a DummyNet [1] using FreeBSD. DummyNet assigns different delays per tcp traffic flow. We configure the DummyNet to give flow delay values between 5-200 ms.

4.2 Traffic Modeling

We use SURGE [7], a web traffic generator, and *Iperf*, for traffic generation. To reflect more contemporary web traffic patterns consisting of large file sizes, we adopt a modified form of SURGE [12] which makes available a number of parameters for tuning the traffic patterns. In our setup, SURGE generates web requests and the file servers generate replies by sending random sized files. The inter-arrival time of requests in SURGE follows an exponential distribution and can be controlled using the SURGE *rate* parameter. By default, we set the average inter-arrival time of TCP requests to 2 seconds, with which 5-45 active STAs each running between 5-25 threads, gives the aggregate inter-arrival time of about 40-200 ms which is close to what we observed for the uplink rate of TCP measured from the SIGCOMM 2008 traces [28]. By varying the number of active STAs, we can evaluate WiFox under various work loads.

From the traffic analysis [28], we can see that UDP traffic from STAs or APs is constantly generated. This UDP packet transmission in uplink traffic affects the downlink TCP traffic from APs. To emulate this phenomenon, we generate uplink UDP traffic from each STA in our experiments. So, we also use UDP traffic as well as the TCP requests for the uplink traffic. By doing so, we also expect that the UDP traffic could play a role of other interferences which prevents APs' packet transmission [9], thus we refer this UDP traffic as *Background Traffic*. Most of the experiments we performed are tested for background traffic in the range 0-30 Kbps per STA. However, even with the different UDP traffic rate, overall pattern for most of our results remains unchanged and thus until specified otherwise we present results with 25 Kbps only.

4.3 Experimental Procedure

Most experiments in this paper involve one AP and a set of STAs associated with it. All machines are connected to a server from which we can control all testbed components. We use *tcpdump* to gather every trace file at the file servers and STAs and uses *tshark* to extract the relevant data from the tcpdump files. UDP data is analysed with our own tool. We have several kernel modules of the modified MADWIFI driver at the AP for our implementations of various schemes which are dynamically loaded according to testing requirements. Each data point has been obtained by averaging the results from experiments repeated for 5-30 rounds with the duration of each round in the range of 120-240 secs.

We implement various different fairness control algorithms for WiFox like throughput maximization (TM), time fairness (TF) and round robin (RR) etc. Among them we chose time-fairness (TF) for our evaluation. We used 10 prioritization levels as it was a good number to implement our dynamic adaptation scheme. We tried various other quantization levels but there were no benefits in terms of network performance. We evaluate the following protocols for comparison.

- NPC: the default IEEE 802.11g with no priority control.
- WANG: the protocol proposed in [36]. The only existing work on solving traffic asymmetry in WiFi networks that does not require change in STAs.
- NPC-{TF}: NPC is combined with TF.
- PC-{LIN,EXP,LOG,LIG}: APC with one of the linear, exponential, logarithmic, and logistic priority control functions.

To corroborate our understanding of traffic asymmetry problem in real LAE, we run an experiment with default NPC module on our emulated LAE testbed. Figure 11a, shows variation of TxQ length for the duration of experiment. For lesser number of associated STAs, problem of TxQ saturation and subsequent packet losses is not significant, but as associated STAs increase we observe immediate TxQ saturation. Figure 11b shows severe goodput degradation as AP saturates, which conforms with our observation for LAE from SIGCOMM 2008 traces [28].

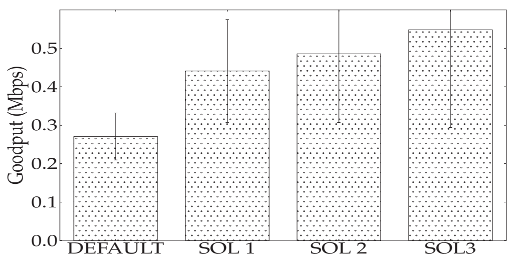


Figure 12: Here default is NPC, TCP Proxy/ECN is solution 1, AP-only TF realization is solution 2 and their combination is solution 3. Clearly these solutions result in 70-100 % improvement in downlink goodput for 25 associated clients

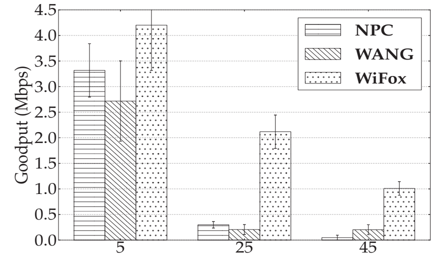


Figure 14: Network downlink goodput benefits significantly with prioritization and performance gains upto 700 % are achievable

5. EVALUATION

In this section we present a detailed analysis and results of the experiments carried to test the performance of WiFox. We summarize the major results as follows.

- AP only TF, TCP ECN & Proxy solutions result in improvements of around 70-100 % for downlink goodput.
- Among all the performance degradation factors, traffic asymmetry results in maximum performance loss for LAE. WiFox adaptively prioritizes AP's channel access and results in 400-700 % goodput improvement for downlink traffic and 50-150 % total goodput improvement.
- WiFox enhances user experiences with a faster requests serve rate. It results in the reduction of response time by 40-60 %.
- All variants of APC perform significantly better than existing solutions.

5.1 Non-APC Solutions

Since we strive to ensure that our solution will be easily deployable, we limit our evaluation to existing solutions that require modifications only to APs and require no further changes in STAs or 802.11 MAC protocol. Figure 12 shows the performance comparison for TCP/ECN and AP-only fairness realizations for 25 associated clients. We observe that downlink goodput performance improves by 70-100% when these solutions are applied. Clearly solution 2 does not uses TCP Proxy/ECN and yet shows performance improvements as compared to NPC, thus we should expect performance improvements for solutions without TCP Proxy (NAT boxes) also. All the solutions presented here are practical realizations of existing works [33, 25, 19] focusing on one particular performance degradation factor. We expect even better performance gains by solving the problem of traffic asymmetry.

5.2 WiFox

WiFox combines implementations of TCP Proxy/ECN and AP-only fairness with adaptive AP priority control (APC). We will now analyze the performance of WiFox with other existing schemes.

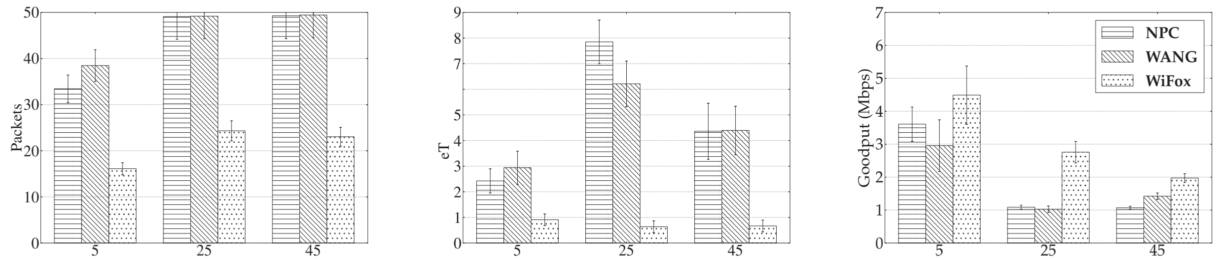
5.2.1 *Performance*

We compare the performance of WiFox with default NPC and WANG. For brevity of discussion we chose PC-LGA from among many APC variants for comparison in this section. Later we will compare the performance of all APC variants in Section 5.2.3. Figure 13a shows that dynamically prioritizing AP over STAs avoids queue saturation and packet losses due to buffer overflow and increased contention. Average TxQ length for 45 clients is slightly less than that for 25 clients which is attributable to the logarithmic nature of the mapping function. In Figure 13b, we observe that for WiFox TCP retransmission rate is not affected by the increased number of associated clients as it results in reduced contention by prioritizing AP traffic. Figure 13c strongly demonstrates significant goodput performance gains achieved through priority control. As the amount of traffic load increases with the number of associated clients we observe a slight decrease in uplink traffic which is compensated with significant improvement for downlink goodput (Figure 14) for *WiFox*.

Figure 15 shows that downlink goodput improvements are not achieved at the cost of fairness. We compared time fairness performance of various schemes using Proportional Fairness Index (PFIndex), which is defined as $PF = \frac{\sum_{i=1}^{N} X_i \log X_i}{N}$, where X_i is the goodput of node $i \in [1, N]$ and N is the number of clients. We also observe that fairness performance for WiFox is marginally better than NPC-TF because AP gets more airtime for WiFox enabling effective realization of fairness schemes [33]. To improve our understanding of whether prioritizing AP impacts link rate adaptation, we observed the corresponding average link rates for each station. Figure 16 shows that no such correlation exists and we have similar average link rate with or without priority control.

5.2.2 *User Experience*

So far we have discussed performance in terms of network goodput and fairness. WiFox results in significant goodput performance improvement, but that does not always correlate with better user experience. Reduced response time for a request is considered as more critical than improved goodput for activities such as web browsing [8]. In this section we probed whether WiFox results in reduced response time. We developed an experiment *Fixed and Inflated*, where each client sends a fixed number of requests (25), and observation duration is inflated from two to four minutes. Sending fixed number of requests for a longer duration ensures reception of all sent requests by the server. Figure 17, shows the cu-



(a) AP's Transmission Queue Length (b) % Retransmission (c) Network Goodput

Figure 13: All three graphs use the same legend. (a) Dynamically prioritizing AP traffic reduces TxQ length by 40-50 % (b) Number of retransmissions are reduced by 70-80 % (c) thus consequently resulting in aggregate goodput improvements by 75-125 %

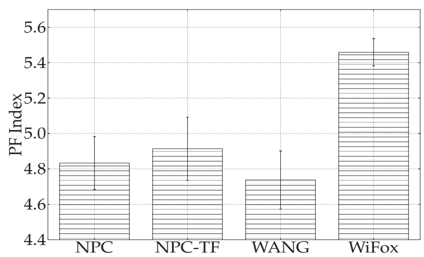


Figure 15: Proportional Fairness Index for 25 clients shows marginal improvement of 10-20 % in fairness among contending STAs

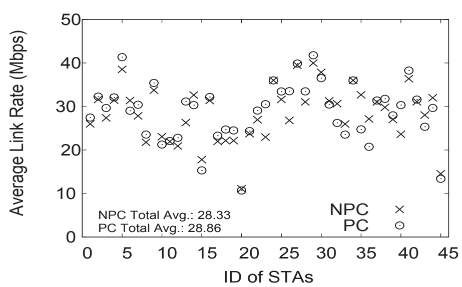


Figure 16: Prioritization does not affects the link rates

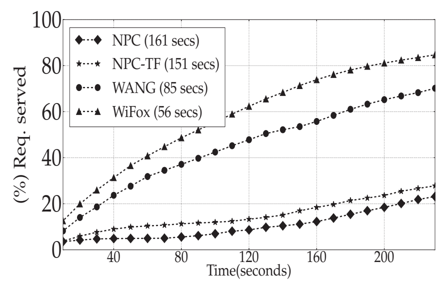


Figure 17: WiFox requests serve rate is 4 times faster than NPC and it reduces response time by 40-60 %. Values in parenthesis give the average response time for each scheme

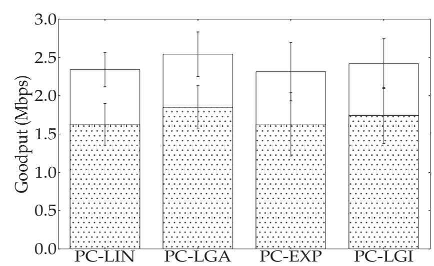


Figure 18: Goodput for APC variants with 25 associated clients

mulative distribution of completed requests with respect to time. Here a steeper slope ensures faster requests serve rate. We observe that WiFox results in reduced response time, as backlogged packets in TxQ are promptly served. As WiFox ensures higher channel access priority to APs, thus delay at its TxQ for response packets is lesser as compared to other schemes. 70-80 % of sent requests are not served within the observation interval for AP's without priority control. Shorter response time for WiFox not only improves the user experience, but with the philosophy of "race to sleep" will save energy for smart devices [24].

5.2.3 *APC Variants*

So far we have discussed results with PC-LGA scheme for brevity. In this section we will compare performance differences among all the APC variants. We carried out an experiment with all variants on similar lines as explained in section 4.3. Figure 18 shows their goodput performance for 25 associated clients. In terms of goodput, there is little difference, though PC-LGA performs slightly better than the others. We observed earlier that for 25 or more associated clients, the queue saturates instantly to the maximum value. PC-LGA is aggressive in assigning higher priority to downlink traffic, but under a heavy load as emulated from real LAE, it outperforms other APC variants. Aggressive priority assignment of PC-LGA is evident in Figure 19 as it has the smallest TxQ length of all the variants. As PC-EXP is the least aggressive its TxQ has more packets backlogged at any given time, similarly, as expected PC-LGI is bounded by exponential and logarithmic mappings. Figure 20 shows the

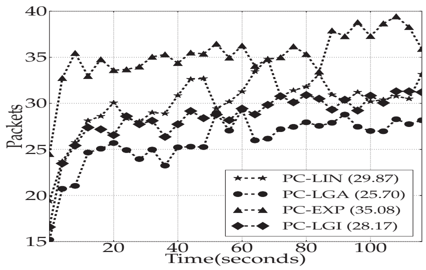


Figure 19: TxQ length for AP with 25 clients for APC variants. Values in parenthesis gives the average TxQ length for each scheme

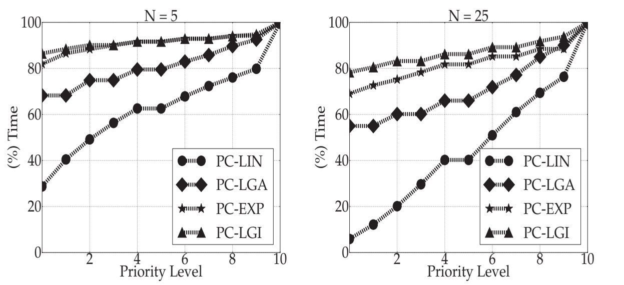


Figure 20: Distribution of Priority Level for APC Variants for 5 and 25 associated clients respectively

cumulative distribution of time spent at each priority level by APC variants for two disparate traffic loads. PC-LIN is uniformly distributed for all priority levels, with a steeper slope for 25 clients. With fewer associated clients PC-LGI follows PC-EXP, contending with default priority for most of the time and swiftly moves towards PC-LGA as the traffic load increases.

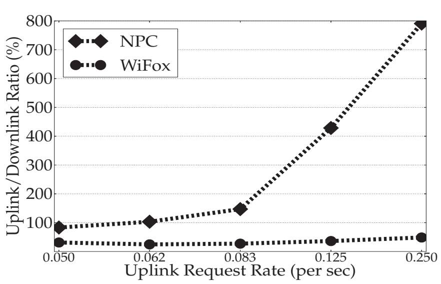


Figure 21: Performance with varying Uplink TCP Traffic for 25 STAs. For entire experiment downlink request rate for each client was 0.50 per second

5.3 Test for Robustness

To evaluate the performance of our scheme for robustness, we tested WiFox for following scenarios.

5.3.1 *Impact of varying Uplink Traffic*

In all our previous experiments tcp uplink traffic is generated by requests sent by STAs to web servers. To account for TCP uplink traffic generated by activities like photo uploads, cloud synchronization etc, we test WiFox for varying uplink traffic. Wireless stations respond to the requests generated by the web server at different rates. Figure 21 shows the performance of WiFox compared to NPC for varying additional uplink traffic. For default NPC, we observe significant improvement in uplink goodput at the cost of downlink traffic. Conversely for WiFox we observe a slight increase in uplink/downlink goodput ratio. WiFox prioritizes downlink traffic and ensures service of downlink requests is not affected with increasing uplink traffic. Such a behaviour is highly desirable for a large audience environment, where excess uplink activities of a few clients can impact the downlink performance of an entire network.

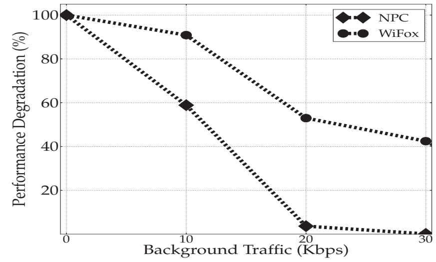
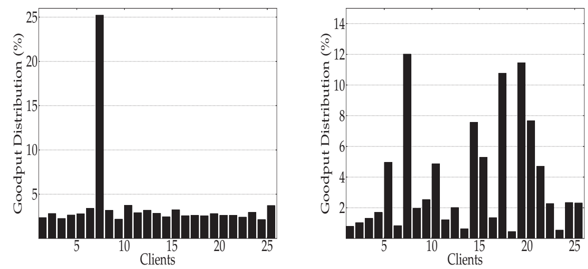


Figure 22: Goodput for varying UDP background trafic with respect to zero background goodput of each



(a) NPC-TF (b) WiFox

Figure 23: Normalized distribution of downlink goodput for each STA. STA_{mischv} (STA 7) requests files of large size and impacts the performance of other clients for NPC-TF scheme

5.3.2 *Impact of Background UDP Traffic*

In this experiment, all associated clients generate TCP traffic using SURGE as usual, but they vary the amount of uplink UDP traffic from 0 to 30 Kbps. As previously discussed, increase in background traffic, results in intensified channel contentions for AP. Also more UDP uplink requests imply fewer TCP requests sent to the server and thus results in reduced TCP downlink traffic. In figure 22, we show downlink goodput degradation for varying UDP traffic with respect to the goodput without any UDP traffic for both the schemes. We observe widening of the performance degradation gap with increasing background traffic. For background traffic of 30 Kbps per STA, we observe TCP downlink traffic for other schemes is nearly zero, whereas WiFox still manages to serve TCP requests.

5.3.3 *Impact of bulky downloads*

So far we considered all users requesting short transfers when using the web. In this experiment we want to test what happens if a client downloads large files. All the other 24 STAs send TCP requests as usual, except for one STA (STA_{mischv}) which requests large sized files (150 Mb). In order to ensure that TxQ has maximum packets for STA_{mischv}, we start its transfer two seconds before the others. Figure 23 shows the distribution of downlink goodput achieved by each node for NPC-TF and WiFox. It is evident that STA_{mischv} dominates the TxQ at the AP and achieves maximum goodput without prioritization. Whereas WiFox ensures that downlink airtime is fairly shared among all STAs avoiding unfairness due to bulk download. With prioritization, AP has maximum access to channel resources and ensures effective realization of AP-only fairness schemes. It should be noted that Figure 23 shows normalized downlink goodput distribution for each associated STAs and thus the two figures are not directly comparable to each other in terms of actual downlink goodput observed.

5.3.4 *Impact of Multiple APs*

The application domain for WiFox are LAEs which implies presence of multiple APs. This necessitates the analysis of WiFox's behaviour in presence of multiple APs. At the macro level we discussed in section 3.1 that high priority slots are assigned randomly. This ensures that multiple APs operating in proximity of each other are not much affected by their neighbour's priority assignment. At micro level, WiFox requires that all APs operating in proximity of each other should use same set of 802.11e parameters. This ensures that even when two APs are concurrently assigned high priority they can compete with each other for channel access as usual. Clearly for such an event the APs downlink performance improvement will be lesser but since prioritization works as usual thus we can ensure better performance than NPC APs for similar network environment. Moreover random priority assignment ensures that such events do not have much impact over longer run.

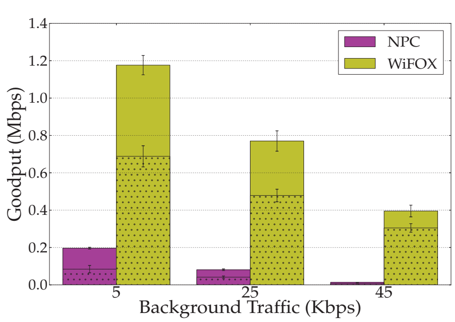


Figure 24: For Multi AP scenario, aggregate downlink goodput for WiFox APs is significantly better than NPC ones. Downlink network goodput for AP2 is stacked over AP1's to enable aggregate goodput comparison

To corroborate our understanding for multiple AP scenario we conducted an experiment with two APs on our testbed. Each AP is associated with 20 STAs and operate in close proximity of each other. The placements of APs and nodes are similar to our original configuration and STAs are randomly selected to connect to either of the two APs. Figure 24, shows the downlink goodput performance of each AP for WiFox and NPC. We observe that APs with NPC suffer significantly as compared to those with WiFox. We observe around 10 times improvement in overall network goodput for 25 Kbps background traffic. Clearly the performance improvements are more than single AP scenario as we discussed earlier which supports our understanding that NPC APs suffer more in multiple AP scenario as compared to WiFox.

6. RELATED WORK

WLAN performance enhancement has received a considerable amount of attention in the research community. Research in the area explores centralized solutions for appropriate channel assignments [21, 26], evaluating capacity for each AP [35, 13], or concurrent connections with multiple APs [16, 32] etc. to improve the overall network performance. These solutions can be orthogonally combined with our proposed solution to improve the performance of a large scale WiFi network.

There have been many proposals to resolve the problems associated with the asymmetric access patterns of WiFi to improve the overall network throughput (e.g., [3, 36]). Most of these proposals either require modifications in both APs and STAs [18, 22, 17], require all nodes to support 802.11e [17], or focus on equalizing the downlink and uplink traffic volume [22, 36]. Since downlink traffic is 4 to 10 times greater than uplink traffic in LAE, these solutions are not effective and even detrimental to the overall performance. Though Bruno et al. [9] makes modification only at APs but requires knowledge of optimal collision probability. Estimating this optimal value for diverse network traffic is non-trivial and limits the robustness of the solution.

Some of the related work on performance enhancement of wireless LANs [15, 33, 6, 30], focuses on the issue of the existence of rate diversity. They propose different methods of providing fairness amongst the STAs as a measure to alleviate overall network performance. Tan et al. [33] proposed a *Time Based Fairness* algorithm (TBR) to achieve a significant gain in the throughput as compared to the *Throughput Based Fairness* provided by IEEE 802.11 DCF in multi-rate WLANs. This paper proposes an AP-only scheme for allocation of equal channel occupancy time to all STAs to improve the aggregate throughput in infrastructure networks. Banchs et al. [6] also propose to solve the *performance anomaly* [11] problem in IEEE 802.11 DCF by the use of proportional fairness for throughput allocation. Aziz et al. [4], attempts to solve the problem of jointly providing efiňĄciency and fairness in wireless networks in general as a utility maximization problem. Our approach of prioritizing AP for channel access enables efficient realization of these fairness schemes([30, 33, 11, 4, 34, 5]).

7. CONCLUSION

In this paper we have examined the factors responsible for poor performance of WiFi for large audiences. We have emulated a large conference like network environment on our testbed with 45 nodes. We have give the reasoning how presence of traffic asymmetry is one of the major factor for performance degradation for such an environment and proposed *WiFox* which solves the problem of traffic asymmetry along with performance degradation due to rate-diversity/fairness and TCP behaviour. On our testbed we have demonstrated that it improves downlink goodput by 400-700 % and enhances user experience by reducing average response time for a request by 30-40 %. We have tested *WiFox* for robustness and have demonstrated that under various test conditions it outperforms the existing WiFi implementations. WiFox requires modifications to AP's software only and is adaptive to various traffic loads, making it a suitable candidate for wide scale deployments.

There are few open problems related to design of WiFox: exploring relationship between different traffic patterns and appropriate APC variants, characterizing performance of WiFox for MIMO based 802.11n APs and developing a priority aware scheduler enabling QoS support. WiFox improves the overall network goodput achieved thus we expect it to improve the performance for real time applications also for LAEs. We believe that it should be possible to ensure QoS and support real time applications with appropriate tuning of prioritization parameters and we intend to pursue this direction in future. Our testbed currently supports evaluation for 802.11g only and we plan to further extend our analysis for 802.11n also.

8. ACKNOWLEDGEMENTS

We would like to thank the following people. Our shepherd, Bhaskar Raman, for his helpful feedback, as well as Suman Banerjee, Venkata N. Padmanabhan, Kyunghan Lee for their valuable comments. We would also like to thanks Marhn Fullmer for his help in setting up the testbed. This work is supported in part by NSF awards CNS-1016216 and IIS-0910868. Any opinions, findings and conclusions expressed in this work are those of the authors and do not necessarily reflect the views of NSF.

9. REFERENCES

[1] DummyNet: http://info.iet.unipi.it/ luigi/dummynet/.

[2] MADWIFI Project: http://madwifi-project.org/.

[3] E. Aguilera, J. Casademont, J. Cotrina, and A. Rojas. Perf. enhancement of wlan IEEE 802.11 for asymmetric traffic. In *IEEE PIMRC*, 2005.

[4] A. Aziz, J. Herzen, R. Merz, S. Shneer, and P. Thiran. Enhance &; explore: an adaptive algorithm to maximize the utility of wireless networks. In *International Conference on Mobile computing and networking*, MobiCom '11. ACM, 2011.

[5] A. Babu, L. Jacob, and V. Brijith. A novel scheme for achieving time based in IEEE 802.11 multirate wireless LANs. In *IEEE ICON, 2005.*

[6] A. Banchs, P. Serrano, and H. Oliver. Proportional fair throughput allocation in multirate IEEE 802.11 e wireless LANs. *Wireless Networks*, 2007.

[7] P. Barford and M. Crovella. Generating representative Web workloads for network and server performance evaluation. In *ACM SIGMETRICS Perform. Eval. Rev.*, 1998.

[8] M. Belshe. More bandwidth doesn't matter (much),

[9] R. Bruno, M. Conti, and E. Gregori. Design of an enhanced AP to optimize TCP perf. in Wi-Fi hotspot networks. *Wirel. Netw.*, 2007.

[10] M. Crovella and A. Bestavros. Self-Similarity in www Traffic: Evidence and Possible Causes. *IEEE/ACM ToN*, 5, 1997.

[11] M. H. Franck, F. Rousseau, G. Berger-sabbatel, and A. Duda. Performance Anomaly of 802.11b. In *IEEE INFOCOM*, 2003.

[12] S. Ha, L. Le, I. Rhee, and L. Xu. Impact of background traffic on performance of high-speed TCP variant protocols. *Computer Networks*, 2007.

[13] N. Hegde, A. Proutiere, and J. Roberts. Evaluating the voice capacity of 802.11 wlan under distributed control. In *Local and Metropolitan Area Networks, 2005. LANMAN 2005. The 14th IEEE Workshop on*, pages 6–pp. IEEE, 2005.

[14] J. Jeong, S. Choi, and C. Kim. Achieving weighted fairness bw uplink and downlink in IEEE 802.11 DCF-based WLANs. In *QShine*, 2005.

[15] L. B. Jiang and S. C. Liew. Proportional fairness in wireless LANs and ad hoc networks. In *IEEE WCNC*, 2005.

[16] B. T. Kandula S., Lin K. and K. D. FatVAP: Aggregating AP Backhaul Capacity to Maximize Throughput. In *USENIX NSDI*, 2008.

[17] F. Keceli, I. Inan, and E. Ayanoglu. Weighted Fair Uplink/Downlink Access Provisioning in IEEE 802.11 e WLANs. In *IEEE ICC*, 2008.

[18] S. W. Kim, B. Kim, and Y. Fang. Downlink and uplink resource alloc. in IEEE 802.11 wlans. *IEEE TVT*, 2005.

[19] S. Kopparty, S. Krishnamurthy, M. Faloutsos, and S. Tripathi. Split tcp for mobile ad hoc networks. In *GLOBECOM*, 2002.

[20] G. Maier, A. Feldmann, V. Paxson, and M. Allman. On dominant characteristics of residential broadband internet traffic. In *Internet Measurement Conference*. ACM, 2009.

[21] R. Murty, J. Padhye, R. Chandra, A. Wolman, and B. Zill. Designing high perf. enterprise wi-fi networks. In *NSDI*, 2008.

[22] N. Nandiraju, H. Gossain, D. Cavalcanti, K. Chowdhury, and D. Agrawal. Achieving Fairness in Wireless LANs by Enhanced IEEE 802.11 DCF. In *IEEE WiMob*, 2006.

[23] R. Raghavendra, E. Belding, K. Papagiannaki, and K. Almeroth. Unwanted link layer traffic in large ieee 802.11 wireless networks. *Mobile Computing, IEEE Transactions on*, 9(9):1212 –1225, sept. 2010.

[24] C. Raiciu, D. Niculescu, M. Bagnulo, and M. Handley. Opp. mobility with mptcp. In *MobiArch*, 2011.

[25] K. K. Ramakrishnan, S. Floyd, and D. Black. The Addition of Explicit Congestion Notification (ECN) to IP, 2001.

[26] S. Rayanchu, V. Shrivastava, S. Banerjee, and R. Chandra. Fluid: improving throughputs in enterprise wlans through flexible channelization. MobiCom '11, 2011.

[27] M. Rodrig, C. Reis, R. Mahajan, D. Wetherall, and J. Zahorjan. Measr. based char. of 802.11 in a hotspot setting. In *ACM E-WIND*, 2005.

[28] A. Schulman, D. Levin, and N. Spring. On the fidelity of 802.11 packet traces. In *PAM*, 2008.

[29] S. Shah and B. Noble. A study of e-mail patterns. *Software, Practice & Practice*, May 2007.

[30] J. Shin, H. Roh, D. Lee, and S. Kim. Generalized proportional fair rate allocation schemes for IEEE 802.11e wireless LANs. In *APCC*, 2008.

[31] S. Shin and H. Schulzrinne. Balancing uplink and downlink delay of VoIP traffic in WLANs using Adaptive Priority Control (APC). In *ACM QShine*, 2006.

[32] H. Soroush, P. Gilbert, N. Banerjee, B. Levine, M. Corner, and L. Cox. Concurrent wi-fi for mobile users: analysis and measurements. In *ACM CoNEXT 2011.*

[33] G. Tan and J. Guttag. Time-based fairness improves performance in multi-rate WLANs. In *USENIX ATC*, 2004.

[34] I. Tinnirello and S. Choi. Temporal Fairness Provisioning in Multi-Rate Contention-Based 802.11e WLANs. In *IEEE WoWMoM*, 2005.

[35] H. Velayos, I. Mas, and G. Karlsson. Overload protection for ieee 802.11 cells. In *Quality of Service, 2006. IWQoS 2006. 14th IEEE International Workshop on*, 2006.

[36] X. Wang and S. A. Mujtaba. Perf. enhancement of 802.11 wlan for asymm. traffic using an adaptive MAC layer protocol. In *IEEE VTC*, 2002.