Passive Detection of Fat Users in WiFi Networks using Thompson Sampling

Lorenz Julius Pusch, Anatolij Zubow and Falko Dressler
Technische Universität Berlin, Germany
l.pusch@campus.tu-berlin.de, {zubow,dressler}@tkn.tu-berlin.de

***Abstract*—The user experience in large WiFi networks, such as those on university campuses, can be greatly enhanced by monitoring user activity. Early detection of high-bandwidth users, often called *fat users*, who consume substantial radio resources, can help prevent the formation of network hotspots. This enables both short-term measures, such as traffic shaping, and long-term solutions, like deploying additional access points. In this paper, we present a fully passive method for detecting fat users in WiFi networks that requires no modifications to existing WiFi infrastructure. Our approach employs multiple WiFi interfaces to passively capture WiFi traffic over the air, with channel hopping managed intelligently through a Multi-Armed Bandit framework using a modified Thompson Sampling algorithm. This technique allows effective monitoring of a broad radio spectrum, even with a limited number of interfaces. Simulation results and real-world experiments demonstrate the feasibility of this approach in both stationary and non-stationary network environments.**

***Index Terms*—Wireless, 802.11, passive sensing, user activity, Thompson sampling**

I. Introduction

The popularity of IEEE 802.11 (Wi-Fi) networks has surged over the past decade, thanks to their convenient connectivity available anytime and anywhere. However, the Access Points (APs) that facilitate network access are often deployed in a spontaneous and sometimes chaotic manner, leading to highly variable AP densities and an uneven distribution of load among them [1]. Hence, to improve the operation of next-generation WiFi networks, monitoring user activity over time is essential [2]. In particular, detecting so-called heavy or fat users is important, as this can help prevent the creation of hotspots, i.e., overloaded APs, by means of traffic shaping, load balancing [3] and reconfigurations like band steering [4] in the WiFi access. Fat users are users who have high resource consumption in the access network, caused by their high network traffic and unfavorable wireless conditions, i.e., low signal quality due to a significant distance from the nearest access point (AP), an outdated WiFi standard with low-order MCS and insufficient channel bonding capabilities, or ultra-low cost single-input single-output (SISO) systems. However, monitoring such users is not easy, as it requires comprehensive network monitoring, including both the core and access networks. A completely passive approach without active monitoring within the WiFi access or backhaul network would therefore be desirable.

Contributions: We present `PDF-WiN` which is a practical system for the passive detection of fat users in Wi-Fi networks. `PDF-WiN` utilizes multiple WiFi interfaces to passively sniff WiFi traffic over the air whereas the channel hopping is steered by an Multi-Armed-Bandit (MAB) based on a modified Thompson Sampler. Thompson sampling is a common heuristic for MAB and is used due to its efficient balance between exploration and exploitation. This enables the intelligent monitoring of a much larger radio spectrum even with small number of sniffing interfaces. It is also able to handle non-stationary environments with changing traffic load and radio channel conditions. Results from simulations and experiments with a prototype build from inexpensive off-the-shelf hardware show that `PDF-WiN` is able to accurately identify the fat users even in non-stationary environments and using only a few sniffing interfaces.

II. Related Work

Related work falls into two categories:

Passive Sniffing: Friess [5] proposed a multi-channel sniffing system for the identification of WiFi devices and analyzed the efficiency of channel hopping in comparison to a system without channel switching. Results for a low-traffic scenario show that the number of sniffing interfaces can be much lower than the total number of channels and still the rate of missed packets can be kept low. Li et al. [6] analyzed the performance of different channel hopping strategies for determining the number of users on a certain channel and their activity over time. They were able to show that the number of channel switches had a large influence as during channel reconfiguration no packets can be received. Hao et al. [7] targeted the problem of large scale crowd counting using passive WiFi sensing. Here fingerprints were created from sensing data mainly including timestamp, signal strength, frame type and MAC address and later used to detect devices. Song et al. [8] proposed a passive client-side approach which allows accurately derive metrics such as airtime and throughput with only a minimal amount of observed WiFi traffic. Passive Monitoring has also found applications in sensing human activities, as Liu et al. [9] presented in their work. Ge et al. [10] stated that this activity tracking through WiFi Sensing and Monitoring is also a relevant and important topic for future healthcare.

Thompson Sampling: The usage of Thompson Sampling for rate adaptation in 802.11ac WiFi networks was proposed by Qi et al. [11]. Bardou et al. [12] targeted IEEE 802.11ax networks and showed that a method based on Thompson Sampling can be used to adapt the Clear Channel Assessment (CCA) sensitivity threshold and its transmission power leading to improved spatial reuse and hence better performance. Another approach to efficiently select the next free channel to transmit using Thompson Sampling was conducted by Maskooki et al. [13]. Ye and Wang [14] proposed an improved Thompson Sampler (TS) method for Dynamic Spectrum Access (DSA) in Cognitive Radio (CR) with non-stationary environments. Results show that the proposed approach outperforms classical TS as outdated information are deleted after some time. The threshold method used by our approach was inspired by this work. Trovo et al. [15] proposed a sliding-window approach to TS and analyzed the impact of the window size in different scenarios.

Our work provides a cost efficient passive monitoring system. While previous studies have focused on binary rewards, our approach uses a modified version that utilizes the radio airtime as a continuous reward.

III. Multi-Armed-Bandit Primer

MAB is a framework for algorithms that make decisions over time under uncertainty. It focuses on optimizing rewards through exploration and exploitation. It is designed for scenarios in which multiple options, or "arms," must be selected in order to maximize the reward. The next action is always based on the current learning state and feedback of last selections [16], [17].

There are various versions of the MAB, which can be used in stationary but also in non-stationary cases, where probabilities of each bandit change over time. Specific algorithms e.g., are the simple ϵ-greedy, where a small percentage of choices are made randomly to explore new options. But also Thompson Sampling (TS) and Upper Confidence Bound (UCB) are common methods [16]. The TS is a widespread heuristic for the MAB problem and its basic version was designed for a stationary environment [17]. Imagine there are $\mathcal{K}$ bandits. Only one Bandit k can be selected at a time. Each of them has a fixed (stationary) probability θ_k to give a positive reward (1) and a probability of $1-\theta_k$ to give a negative one (0). To model the likelihood of the reward for each bandit it is common for TS to use beta distribution. Initially it begins with an independent prior belief. The parameters of the distribution, α and β, are both one. As a result, the distribution is constant within the interval [0, 1]. Afterwards, each bandit is then updated as follows:

$$(\alpha_k,\beta_k)=\begin{cases}(\alpha_k,\beta_k) & \text{, bandit } k \text{ is not chosen}\\ (\alpha_k,\beta_k)+(1,0) & \text{, chosen and reward is } 1\\ (\alpha_k,\beta_k)+(0,1) & \text{, chosen and reward is } 0\end{cases}$$

To decide which bandit will be visited next, an estimated reward is generated from each bandit and the one with the highest reward is chosen.

IV. System Model & Problem Statement

We consider a IEEE 802.11 WiFi infrastructure network consisting of a large number of APs and user devices (STA)

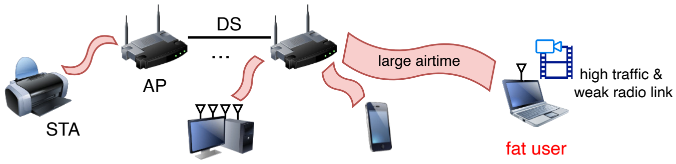


Figure 1. System model.

(Figure 1). The Basic Service Sets (BSS), i.e., APs with associated clients, are operating on either different or same radio channels (overlapping BSS) from either 2.4, 5 or 6 GHz band. The user devices generate network traffic in both downlink and uplink. We assume very heterogeneous users, i.e., from ultra-low cost legacy SISO systems to state-of-the-art large MIMO systems operating on wide channels (cmp. Figure 1). Moreover, no assumptions are made about the stationarity of the users with respect to the used radio resources. This means that both the traffic pattern and the radio conditions (due to mobility) can change.

The following specific model was used. The time is divided into T timeslots, $t \in [1,T]$, of equal duration τ. The spectrum is split into N channels with equal bandwidth, $\mathcal{N}=\{1,2,\dots,N\}, n\in\mathcal{N}$. The non-stationary environment is divided over the entire time T into M stationary segments $m\in[0,M]$. The segments are represented as slots $\varphi_0,\varphi_1,\dots,\varphi_{M+1}$ with $\varphi_0=0$ and $\varphi_{M+1}=T$. In each segment the network traffic load changes for each user on all channels:

$$a_{u_n}(t)=P(p_{u_n,m}),\ \varphi_m<t\leq\varphi_{m+1}$$

Here $a_{u_n}(t)$ is the state of the user u_n and specifies the airtime in a timeslot t. $P(p_{u_n,m})$ is a random variable with Poisson distribution. There are $\mathcal{U}_n$ users per channel n.

$$\mathcal{U}_n=\{1,2,\dots,U_n\}, u_n\in\mathcal{U}_n$$

And the available W monitor interfaces are specified by $\Omega=\{1,2,\dots,W\}$ with $\omega\in\Omega$.

Our objective is to find the fat users, i.e., those occupying the largest airtime which is $p_{u_n,m}$ in our model. Such users are characterized by either high network load and/or poor radio conditions, i.e., low SNR, no or small MIMO spatial multiplexing, no or low channel bonding (see Figure 1).

V. PDF-WiN Approach

Figure 2 shows the architecture of the envisioned PDF-WiN approach. The frames received over the W WiFi sniffing interfaces are decoded and their airtime is estimated using information including the PHY header. This information is passed to the adapted TS, which updates its internal state and decides on the channels to be used by each interface in the next time slot t. After the T timeslots have been completed the TS outputs the identified fat users together with their channels. The proposed adapted TS is described in next section.

A. Adapted Thompson Sampler

The core of `PDF-WiN` is the modified TS which differs from the plain version described in Section III as follows (Algorithm 1). First of all, the reward in our system is not binary. Hence, the Beta-distribution cannot be used since it is only defined in the interval $[0, 1]$. Moreover, the TS must be capable of visiting multiple bandits in each time slot, because it is equivalent to using more than one sniffing interface, i.e., $W > 1$. Additionally, to improve its adaptability in non-stationary environments additional modifications (Threshold method, Sliding Window) were made.

Algorithm 1 Adapted Thompson Sampling

Input: A prior for the reward distributions
$D \leftarrow \emptyset$ ▷ observed data from visited channels
for $t = 1, \ldots, T$ **do**
  $\theta^t \leftarrow$ draw sample from Posterior $P(\theta^t \mid D)$
  Compute expected rewards $\mathbb{E}_{P(r|n,\theta^t)}[r]$ for all channels n
  Rank all channels by expected reward
  $a \leftarrow \min(W, N)$
  if rand() $\leq 95\%$ **then**
    $\mathcal{N}^*(t) \leftarrow$ Select top a channels maximizing $\mathbb{E}_{P(r|n,\theta^t)}[r]$
  else
    $L \leftarrow$ Actions not in the top a
    $k \leftarrow \min(a, |L|)$
    Randomly select k channels from L
    if $k < a$ **then**
      Add $(a - k)$ channels from the top a to $\mathcal{N}^*(t)$
    end if
  end if
  Visit all channels from $\mathcal{N}^*(t)$ and observe rewards $r(t)$
  Append $(r(t), \mathcal{N}^*(t))$ to D
end for

A modified TS is used to estimate the fat users in the network. It controls the channels to be monitored (sniffed) by each of the W available interfaces in the next time slot. The reward for each channel n in each timeslot t is the airtime of the heaviest user, i.e., user with largest airtime, of this channel:

$$r_n(t) = a_{u_{\text{heavy}}}(t); \text{ with } u_{\text{heavy}} \text{ is in } \underset{u_n \in \mathcal{U}_n}{\operatorname{argmax}}(a_{u_n}(t))$$

To make it possible that the TS is capable of visiting multiple bandits in each time slot the adapted TS visits $\min(W, N)$ channels. This way it is possible to update in one timeslot multiple estimated distributions of the channels.

In order to operate in non-stationary environments outdated observations need to be discarded. Three methods are available:

1) Plain TS,
2) Window TS,
3) Threshold TS.

Plain TS is the TS as described above, where all observations even very old ones are kept. Hence, this method is only suitable for stationary environments. *Window TS*, is a method that uses a sliding window to construct the estimated distribution of each channel based on the rewards from the last n time slots. Older reward values are discarded. *Threshold TS* uses a method adapted from [14]. It stores the rewards of a channel for the first n visits. Afterwards a confidence interval is calculated from these initial rewards on the channels estimating distribution. For following visits, the estimated distribution is based on the most recent n rewards like a sliding window. Additionally, if a observed reward falls outside the confidence interval the entire learning state will be reset and the TS restarts with the initial learning phase for that channel. The last two methods are suitable for non-stationary environments.

Finally, there is an additional mechanism for adapting to changes on the channel. With a probability of 5% the TS does not visit the channels with highest estimated reward but randomly chooses the channels to be used for the next time slot. In case of fewer channels available then the number of chosen channels, then the chosen ones are included again in descending order regarding the estimated reward.

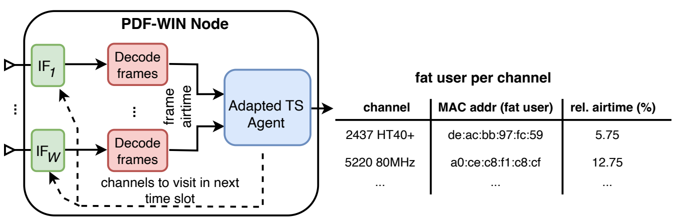


Figure 2. `PDF-WiN` architecture of a single node with W interfaces.

B. Prototype

As a proof-of-concept (PoC) `PDF-WiN` was implemented as prototype (Figure 3)[1]. The PoC uses commercial off-the-shelf hardware, i.e., multiple WiFi USB NICs connected via USB hub to Linux laptop. Specifically, we used EDUP AX3000 WiFi cards with MT7921AUN chipset which are compliant to IEEE 802.11a/b/g/n/ac/ax (6E) and support monitor/promiscuous mode. On the software side we used the Python library Scapy for parsing WiFi frames. The prototype is capable of handling a different number of NICs for both sniffing and sending. In the prototype shown in Figure 3, eight NICs are connected. During execution it is continuously displaying the current status in the terminal. For the sniffer component it is the observed heavy user of the current timeslot t for each monitor interface ω and for the sender this is the MAC address and corresponding airtime of the simulated heavy user for each channel. Additionally, the sniffer component displays the current state of probability distributions for each channel the TS works with in a live-updated Matplotlib window. The second plot shows how often each channel has been visited. Regarding performance requirements, the program was executed five minutes with eight NICs on a laptop with an Intel(R) Core(TM) i5-7300U CPU with 4 cores at 2.60 GHz. The uptime command reported load averages of 0.85 over the last 5 minutes. This value reflects efficient resource usage, as they stay below the available CPU core count, which implies that the program is capable of managing higher loads effectively.

[1] https://git.tu-berlin.de/lorenz.pusch/ts-passive-monitoring-system

Figure 3. The `PDF-WiN` prototype with 802.11ax NICs connected via USB.

VI. Evaluation

The proposed approach was analyzed both by simulations and real experiments using the prototype.

A. Methodology

For the evaluation the following performance metric was selected. It is the cumulative airtime over all time slots and channels of the selected approach divided by that of the oracle solution:

$$\mu = \frac{R}{R^*} \tag{1}$$

where R is defined as follows:

$$R = \sum_{t=1}^{T} \sum_{n \in C_t} r_{n,m}(t) \tag{2}$$

Here C_t represents the visited radio channels in time slot t:

$$C_t = \{c | c \in \mathcal{N}, c_{\omega_i} \neq c_{\omega_j}, \forall \omega_i, \omega_j \in \Omega, \omega_i \neq \omega_j, \text{ this set has a maximum of } W \text{ elements}\} \tag{3}$$

Note, that R^* is computed in the same way and represents the upper bound as it is computed using full information about traffic on all channels, i.e., oracle solution. Hence, in each timeslot all W sniffing interfaces are distributed optimally over the channels with the highest $p_{u_n,m}$. So, C_t becomes C_t^*:

$$C_t^* = \{c | c \in \mathcal{N}, c_{\omega_i} \neq c_{\omega_j}, \forall \omega_i, \omega_j \in \Omega, \omega_i \neq \omega_j, \text{ this set has a maximum of } W \text{ elements}, p_{u_c,m} \text{ is ranked among the highest } r_n(t), \forall n \in \mathcal{N}\} \tag{4}$$

B. Simulation

We present results from simulations starting with analysis of the selected distribution for TS, the performance of our proposed approach in both non-stationary as well as stationary environments and the comparison with simple random and sequential scanning strategies.

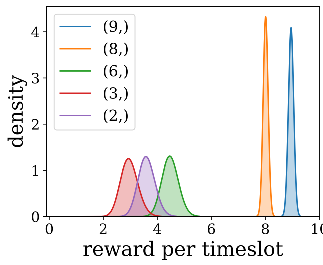
1) Impact of TS Distribution: In the following, we want to examine which distribution should be used in Thompson sampling. Specifically, we examined the Gamma, Poisson and Gaussian distribution. For the analysis we selected $W = 2$ sniffing interfaces, $N = 5$ radio channels, $\tau = 1\,s$ time slot duration and a total number of $T = 1k$ time slots. The number of users was five and each user used a different channel. The user traffic was Poisson distributed with different $\lambda = 2, 3, 6, 8, 9$ for the five users.

Figure 4 shows the different estimated distributions. From the results we can see that all three distributions are suitable for identifying the fat user, i.e., those with the highest traffic, although the number of sniffing interfaces was smaller than the number of channels, $W < N$.

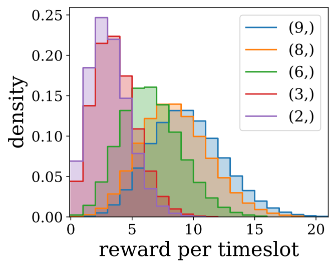
2) Stationary environment: To demonstrate the effectiveness of the proposed approach, it is compared with the following two strategies. In the random strategy at the beginning of a time slot, the channels for the W sniffing interfaces are randomly selected whereas the sequential strategy scans the N available channels in sequential order using the W interfaces. The following scenario was used: $W = 1$ sniffing interface, $N = 7$ channels, $\tau = 1\,s$ time slot duration, a total number of $T = 500$ time slots. `PDF-WiN` was configured to use plain TS with Poisson distribution.

As can be seen from Figure 5 both the random and the sequential strategies are unable to identify the fat users as μ remains low at ≈ 0.45. This is because all channels are visited equally frequently and therefore independently of the occupancy by users. In contrast μ of the proposed TS approach converges to one and hence is achieving a performance close to the oracle solution.

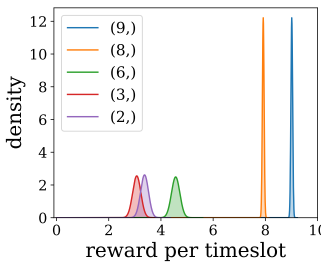
3) Non-stationary environment: Next, we analyzed the three different methods, i.e., plain, window and threshold TS, in a



(a) Gamma distribution



(b) Poisson distribution



(c) Normal distribution

Figure 4. Estimated distributions of TS with different distributions.

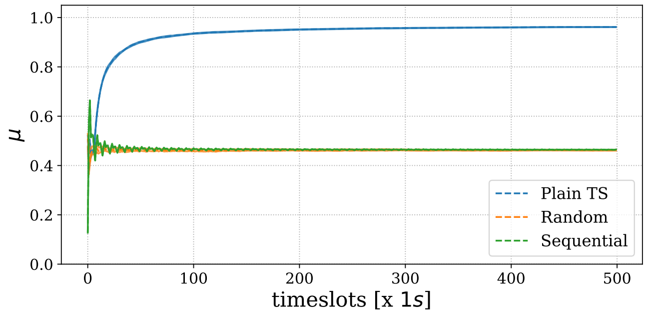


Figure 5. Comparison of plain TS with random and sequential strategies.

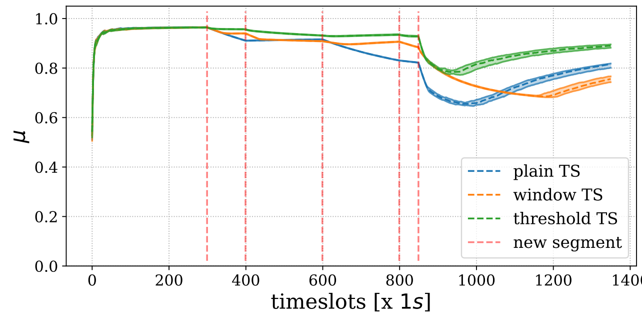


Figure 6. Comparison of different TS methods in a non-stationary environment.

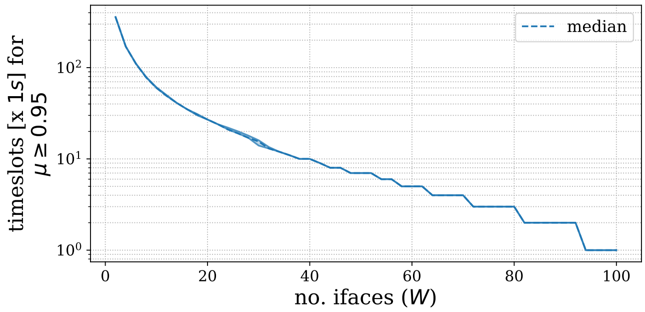


Figure 7. Impact of number of sniffing interfaces.

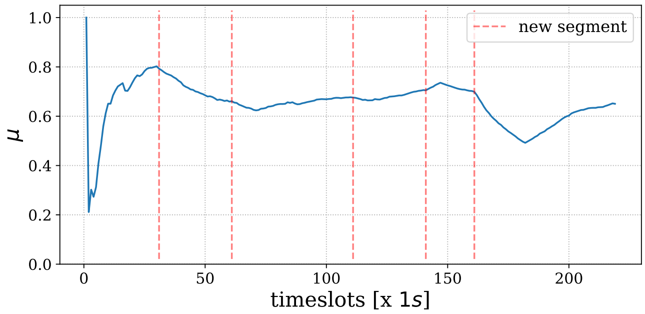


Figure 8. Results from over-the-air experiments with emulated traffic.

non-stationary environment. The following configuration was used: $W = 2$ sniffing interfaces, $N = 5$ radio channels, $\tau = 1\,s$ time slot duration, a total number of $T = 1350$ time slots, use of Gamma distribution for TS and $M = 6$ segments.

Figure 6 shows the results. The plain TS performs significantly worse than the other two methods whereas the threshold TS is able to adapt fastest. Note, that in last segment the channel utilization changed very drastically. The advantage of the threshold TS was highest here, 0.79 vs. 0.66 at $t = 920$.

4) Impact of number sniffing interfaces: The objective of this examination was to find out how many timeslots T are required to reach $\mu \geq 0.95$, i.e., to identify the heavy users with a high degree of certainty. We therefore varied the number of interfaces W, while the number of channels remained constant. The specific configuration was: $N = 100$ radio channels, $\tau = 1\,s$ time slot duration, use of Gamma distribution for TS, stationary environment ($M = 1$) and the use of threshold TS. The number of users and their utilization was randomized in each run and the results are averaged. From the result in Figure 7 we can clearly see the speedup in identification if sufficient sniffing interfaces are available.

C. Experiments

We conclude our evaluation with presenting results from real over-the-air experiments using our `PDF-WiN` prototype.

1) Emulated user traffic: The selected parameters were, $W = 1$ sniffing interface, $N = 5$ radio channels, $\tau = 1\,s$ time slot duration, a total number of $T = 220$ time slots, use of Gamma distribution with window TS and $M = 6$ segments. The user traffic was emulated using five additional WiFi interfaces by injecting WiFi frames via Scapy.

Figure 8 shows a single run where a sufficient high value for μ was achieved. Note, that for the measurement, we used the unlicensed 2.4 GHz ISM band, which was also used by other WiFi devices at the time of the measurement. Therefore, only the emulated traffic could be taken into account resulting in a slightly lower value for μ.

2) Measurements on campus: We used our PoC to identify the fat users in three different locations on our campus, namely: i) office, ii) lecture hall, iii) small library. The selected parameters were $W = 8$ sniffing interfaces, $\tau = 1\,s$ time slot duration, a total number of $T = 600$ time slots (10 min), use of Gamma distribution and Window TS with window size 20. An initial scan was used to find the channels on which APs are operating.

The results for the three environments are shown in Table I. We can observe that in the office location, the fat users are operating in the 2.4 GHz spectrum whereas it is the 5 GHz spectrum that is being used in the other two locations. Moreover, we see that the channel utilization was not particularly high in the three environments. Even the heaviest user consumed less than 2% of the total medium time. Finally, we see that channel bonding is being used and supported by our prototype.

3) Measurement over time: In a last measurement we analyzed how the heavy users change over time. For this the last experiment (VI-C2) was conducted repeatedly for 90 min in a lecture hall during a lecture. Hence, this is a real world measurement without knowledge about the ground truth. The selected parameters were $W = 4$ sniffing interfaces, $\tau = 1\,s$ time slot duration, a total number of $T = 120$ time slots (2 min), use of Gamma distribution and Threshold TS with

Table I
DETECTED FAT USERS (LOCATION: OFFICE)

Location	Channel	Rel. airtime (%)
Office	2462 HT20	0.953
	2412 HT40+	0.408
	2437 HT40+	0.401
Lecture hall	5500 40MHz	1.717
	2437 HT40+	0.899
	2437 HT40-	0.820
Library	5500 80MHz	0.939
	5220 40MHz	0.771
	5220 80MHz	0.708

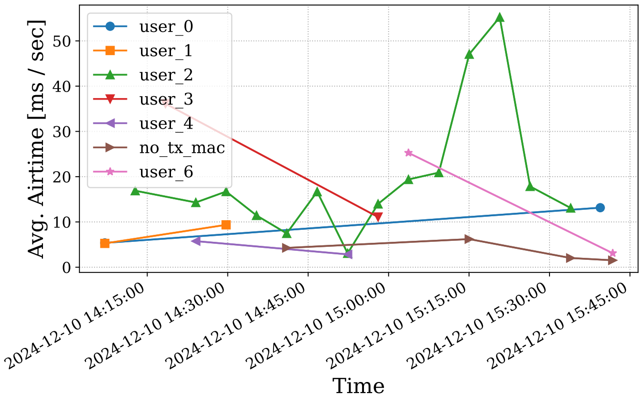


Figure 9. Results from measurements over time on the campus

window size 20. Also for this measurement, an initial scan was used to find the channels on which APs are active. As a result of this scan, the measurement was conducted on 7 frequencies with in total 24 different bandwidths.

Figure 9 shows the result. Each user refers to a MAC address, that is extracted from WiFi-frames. There are some frame-types without an entry for transmitter address, such as Acknowledgments (ACK) and Clear to Send (CTS). These are listed under `no_tx_mac` in the plot, but do not have a high airtime compared to the other users. We can observe that there are users that came up infrequently and spontaneously, like `user_0` or `user_6` in the plot. Besides this, `user_2` was present most the time and consumed on average the highest airtime.

VII. CONCLUSION

Early detection of high-bandwidth users, known as *fat users*, helps prevent network hotspots by enabling short-term traffic shaping and long-term solutions like deploying additional access points. We presented `PDF-WiN` which is using a Multi-armed Bandit based on a modified Thompson Sampler for efficient passive detection of fat users in Wi-Fi networks. Its feasibility in stationary and non-stationary environments was demonstrated by means of simulations and real experiments. As future work we plan to perform field tests in large-scale WiFi university campus networks which would allow us to validate our approach under real channel and network conditions.

ACKNOWLEDGEMENTS

This work was supported by the Federal Ministry of Education and Research (BMBF, Germany) within the 6G Research and Innovation Cluster 6G-RIC under Grant 16KISK020K as well as by the German Research Foundation (DFG) within the project ML4WiFi under grant DR 639/28-1.

REFERENCES

[1] C. Zhang, D. Qiu, S. Mao, X. Hei, and W. Cheng, “Characterizing Interference in a Campus WiFi Network via Mobile Crowd Sensing,” in *Collaborative Computing: Networking, Applications, and Worksharing (CollaborateCom 2015)*, Wuhan, China: Springer, Nov. 2015, pp. 173–182.
[2] E. Rozner, Y. Mehta, A. Akella, and L. Qiu, “Traffic-Aware Channel Assignment in Enterprise Wireless LANs,” in *15th IEEE International Conference on Network Protocols (ICNP 2007)*, Beijing, China: IEEE, Oct. 2007, pp. 133–143.
[3] S. Manzoor, Z. Chen, Y. Gao, X. Hei, and W. Cheng, “Towards QoS-aware load balancing for high density software defined Wi-Fi networks,” *IEEE Access*, vol. 8, pp. 117 623–117 638, Jun. 2020.
[4] Z. Han, J. Liao, Q. Qi, H. Sun, and J. Wang, “Band Steering Technology Based on QoE-Oriented Optimization in Wireless Network,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2020, pp. 1–13, Jan. 2020.
[5] K. Friess, “Multichannel-Sniffing-System for Real-World Analysing of Wi-Fi-Packets,” in *10th International Conference on Ubiquitous and Future Networks (ICUFN 2018)*, Prague, Czechia: IEEE, Jul. 2018, pp. 358–364.
[6] Y. Li, J. Barthelemy, S. Sun, P. Perez, and B. Moran, “A case study of WiFi sniffing performance evaluation,” *IEEE Access*, vol. 8, pp. 129 224–129 235, Jul. 2020.
[7] L. Hao, B. Huang, B. Jia, G. Xu, and G. Mao, “Toward accurate crowd counting in large surveillance areas based on passive wifi sensing,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 12, pp. 14 086–14 096, Dec. 2023.
[8] L. Song, A. Mohammed, and A. Striegel, “A passive client side control packet-based WiFi traffic characterization mechanism,” in *IEEE International Conference on Communications (ICC 2020)*, Virtual Conference: IEEE, Jun. 2020, pp. 1–7.
[9] J. Liu, H. Liu, Y. Chen, Y. Wang, and C. Wang, “Wireless Sensing for Human Activity: A Survey,” *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1629–1645, Jul. 2020.
[10] Y. Ge, A. Taha, S. A. Shah, K. Dashtipour, S. Zhu, J. Cooper, Q. H. Abbasi, and M. A. Imran, “Contactless WiFi Sensing and Monitoring for Future Healthcare - Emerging Trends, Challenges, and Opportunities,” *IEEE Reviews in Biomedical Engineering*, vol. 16, pp. 171–191, 2023.
[11] H. Qi, Z. Hu, X. Wen, and Z. Lu, “Rate adaptation with thompson sampling in 802.11 ac WLAN,” *IEEE Communications Letters*, vol. 23, no. 10, pp. 1888–1892, Oct. 2019.
[12] A. Bardou, A. Busson, and T. Begin, “Improving the Spatial Reuse in IEEE 802.11ax WLANs: A Multi-Armed Bandit Approach,” in *24th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM 2021)*, Virtual Conference: ACM, Nov. 2021, pp. 135–144.
[13] A. Maskooki, V. Toldov, L. Clavier, V. Loscrì, and N. Mitton, “Competition: Channel Exploration/Exploitation Based on a Thompson Sampling Approach in a Radio Cognitive Environment,” in *International Conference on Embedded Wireless Systems and Networks (EWSN 2016)*, Graz, Austria: Junction Publishing, Feb. 2016, pp. 285–286.
[14] S. Ye and S. Wang, “An Improved Thompson Sampling Method for Dynamic Spectrum Access in Non-Stationary Environments,” in *IEEE Applied Sensing Conference (APSCON 2023)*, Bengaluru, India: IEEE, Jan. 2023, pp. 1–3.
[15] F. Trovo, S. Paladino, M. Restelli, and N. Gatti, “Sliding-Window Thompson Sampling for Non-Stationary Settings,” *Journal of Artificial Intelligence Research (JAIR)*, vol. 68, pp. 311–364, May 2020.
[16] A. Slivkins, “Introduction to Multi-Armed Bandits,” arXiv, cs.LG 1904.07272, Apr. 2019.
[17] D. J. Russo, B. V. Roy, A. Kazerouni, I. Osband, and Z. Wen, “A Tutorial on Thompson Sampling,” *Foundations and Trends® in Machine Learning*, vol. 11, no. 1, pp. 1–96, Jul. 2018.