Experimental Study Towards Efficient Interference Avoidance using Wi-Fi 6 OFDMA on SDR

Thijs Havinga, Xianjun Jiao, Wei Liu and Ingrid Moerman
IDLab, *Department of Information Technology*, Ghent University - imec
Ghent, Belgium
{firstname.lastname}@UGent.be

***Abstract*—Interference is an important problem that influences the performance of wireless communication systems in crowded environments. In this work, we experimentally examine interference avoidance using Orthogonal Frequency Division Multiple Access (OFDMA) as defined in IEEE 802.11ax (Wi-Fi 6). The concept involves leaving a portion of the available subcarriers, specifically a designated Resource Unit (RU), unused in the presence of interference—referred to as RU puncturing. This enables concurrent transmission with narrowband technologies while mitigating the effects of interference. We use *openwifi*, an open-source Field-Programmable Gate Array (FPGA)-based Wi-Fi transceiver on Software-Defined Radio (SDR), extended with Wi-Fi 6 compliant OFDMA feature to exploit this mechanism. We show by simulations and experiments that our method operates effectively when using two SDR devices. However, achieving end-to-end system-level improvement is challenging when connecting to a Commercial Off-The-Shelf (COTS) Wi-Fi 6 device. For efficient interference avoidance it is important to effectively schedule punctured transmissions to make most use of the gained spectral efficiency.**

I. INTRODUCTION

In a world with ever-increasing number of devices operating in unlicensed bands, wireless interference is an important problem to consider. The interference can either come from other networks using the same technology (Intra-Technology Interference (ITI)), or from systems using a different technology, so called Cross-Technology Interference (CTI).

Wi-Fi (IEEE 802.11) has evolved from using a small bandwidth of $20\,\mathrm{MHz}$, to up to $320\,\mathrm{MHz}$ now in the new 6GHz band. While a wider bandwidth provides higher throughput in theory, in practice this may be severely influenced by other traffic running anywhere in this wide bandwidth. This interference may be Wi-Fi, or any other electromagnetic source. The Wi-Fi 6 standard supports Multi-User (MU) preamble puncturing, where in a wide bandwidth subchannels of $20\,\mathrm{MHz}$ can be left unused, which is signaled via the preamble.

However, even using a small bandwidth, narrowband interference from wireless technologies using the $2.4\,\mathrm{GHz}$ band, e.g. Bluetooth (Low Energy) and IEEE 802.15.4 (ZigBee), may degrade the performance of Wi-Fi. Similarly, Wi-Fi may also impact the performance of these technologies. Recently the Thread [1] standard, a mesh network with IPv6 support built on IEEE 802.15.4 using the $2.4\,\mathrm{GHz}$ band, has gained attention. It will use a fixed channel of approximately $2\,\mathrm{MHz}$ that may overlap with Wi-Fi. The Matter [2] standard, which focuses on smart home applications, will utilize both Wi-Fi and Thread. Furthermore, the authors of [3] experimentally show that ultra-narrowband communication (using around 10-$200\,\mathrm{Hz}$ of bandwidth), which has potential for ultra low-power wireless sensors, can have a great impact on Wi-Fi links even with an interference power as weak as $-80\,\mathrm{dBm}$. It becomes clear that avoiding narrowband interference is still important in scenarios where multiple technologies coexist.

OFDMA was originally designed to serve multiple users sharing a wireless communication channel at the same time by allocating only a subset of the available subcarriers (tones) to each user. This reduces latency in contention-based networks, as on average less contention is needed to serve all users. Since the available bandwidth is still shared between all users, it does not directly increase the throughput, but eventually the overall network throughput is increased thanks to the reduced contention time.

We propose an alternative usage of OFDMA by not allocating an RU affected by interference to any user. This idea is similar to the Wi-Fi 6 feature of preamble puncturing, but applied to narrowband interference. However, to the best of our knowledge, currently available COTS Wi-Fi 6 devices do not expose interfaces for low-level control like OFDMA scheduling. Therefore, we utilize *openwifi* [4], an open-source FPGA-based 802.11a/g/n transceiver that is supported on Software-Defined Radio (SDR) boards in the Fed4FIRE+ testbed w-iLab.t [5]. We extended it with OFDMA functionality following the 802.11ax standard, to have full control of this feature. This way, we can experimentally examine the performance of RU puncturing and gain insights when applying it in complex coexistence scenarios.

In the remainder, we first list related work on both interference avoidance and detection using customized solutions, then our Wi-Fi 6 compliant solution is described. Following, we present the simulated performance and its corresponding experimental performance. Then we discuss the results and provide suggestions for future work, and conclude the paper.

II. RELATED WORK

A. Interference avoidance

Interference can be avoided using either Medium Access Control (MAC) layer mechanisms or physical (PHY) layer mechanisms. For instance, [6] and [7] discuss the importance of interference avoidance and propose ways to overcome this with mechanisms on the MAC layer. The limitations of these approaches is that they usually focus on sharing the medium in the time domain and cannot ensure concurrent access.

There are several other works that consider the PHY layer. The works in [8] (*Embedded ZigBee*, EmBee) and [9] (*COexist wiFi For zigbEE*, COFFEE) mitigate ZigBee interference for Wi-Fi in a similar way as can be done with OFDMA, namely by nullifying some of the subcarriers. While EmBee uses the difference in frequency offset of both technologies to detect interference, COFFEE continuously examines the spectrum of the received signal using an additional FFT module and interference is detected when the amplitude of a part of the subcarriers is significantly higher than the others. Both use a modified version of the physical layer of IEEE 802.11g; EmBee is implemented on the WARP v3 platform and COFFEE on the USRP platform. EmBee shows improved throughput of ZigBee, with limited throughput loss of 12% for Wi-Fi. COFFEE shows improved throughput for both Wi-Fi and ZigBee as compared to a Time Division Multiple Access (TDMA) scheme.

On the other hand, [10] discusses changes on the physical layer of IEEE 802.15.4 in order to detect CTI and overcome packet errors by an adaptive error recovery mechanism.

In [11], the authors discuss intra-technology interference detection and mitigation using Channel State Information (CSI) and OFDMA in cellular systems. Increased throughput is observed in the presence of inter-cell interference, both when base stations collaborate by exchanging CSI information in combination with a pricing based scheme, and when only a probabilistic approach without coordination is used.

B. Interference detection

In order to detect ITI, the Wi-Fi standard requires a device to perform CCA on each 20 MHz section of the used bandwidth. Several works consider the detection of CTI, and more specifically narrowband interference. This is done with traditional techniques and, more recently, using machine learning.

The authors of [12] used the receiver error codes provided by Wi-Fi chips to detect ZigBee interference. The works in [13] and [14] propose an enhanced Clear Channel Assessment (CCA) mechanism for Wi-Fi, which can detect ZigBee even when the interfering power is lower than the CCA threshold.

The authors in [15]–[17] show that narrowband interference can be effectively detected using deep learning either on raw IQ samples, or using high-level features like packet queue and CCA statistics.

In summary, there exists many solutions to avoid interference. Though some do show advantages in certain scenarios, due to the fact that they are customized solutions, they did not gain popularity in industry. This study, however, explores Wi-Fi 6 compliant features for interference avoidance. To date, we have not seen an in depth experimental study for this purpose. The main scope of this work is not enhanced interference *detection* —instead we assume interference is detected— but we focus on the measures we can take to avoid it using OFDMA, and examine its performance.

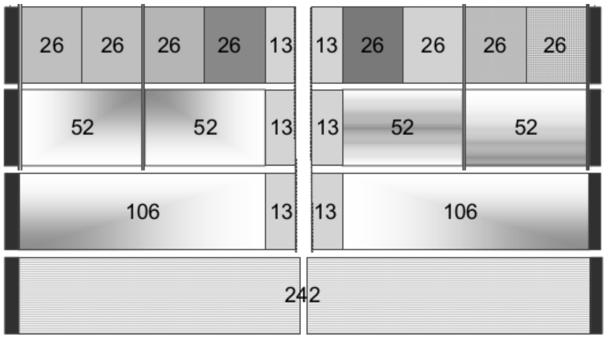

Fig. 1. IEEE 802.11ax RU locations for 20 MHz (adapted from [18]).

III. WI-FI 6 RU ALLOCATION MECHANISMS

The OFDMA extension we implemented on top of *openwifi* follows the IEEE 802.11ax standard. The RUs defined by the standard in the 20 MHz bandwidth are shown in Figure 1. The standard prescribes a procedure for establishing uplink OFDMA via Trigger Frames (TFs) sent by the AP. The TF contains common information about the High Efficiency Trigger-Based (HE TB) uplink packet for all users, such as its duration, and the RU width and index of each user. With our implementation on SDR boards, we can specify arbitrary configurations in the TF. Leaving some of the RUs unoccupied is allowed by the standard. On the other hand, a STA may also send an uplink HE MU packet spanning the full or half the bandwidth (106-tone RU)[1]. In this way, the STA does not have to be triggered to use partial bandwidth, since it can denote this in the HE MU preamble. For downlink traffic, the AP can send an HE MU packet that is assigned to only one user. Using these procedures the AP can control the bandwidth used in both uplink and downlink, thereby controlling the spectrum assigned to COTS Wi-Fi 6 STAs.

IV. THEORETICAL PERFORMANCE

Due to the complexity of establishing a controlled experimental setup while exploiting OFDMA, we limit our scope to a single Wi-Fi station (STA) and Access Point (AP). We focus on CTI by IEEE 802.15.4 since this is already a common technology used in proximity of Wi-Fi and its usage will likely increase with the rise of Matter and Thread.

In order to see the impact of CTI on the Wi-Fi performance with or without the RU puncturing feature, we simulate its performance using a bit-true model of the *openwifi* receiver in MATLAB. For this, the IQ samples of Wi-Fi (either HE SU (Single User) or TB packets) generated by a MATLAB model are superposed with IQ samples of an IEEE 802.15.4 packet captured over the air by an SDR. The 802.15.4 packet was sent at channel 11 (2404-2406 MHz), which interferes with Wi-Fi channel 1 (2402–2422 MHz). Using 802.11ax, the best option then is to use a 106-tone packet (with 102 data subcarriers) to avoid the interference. An HE TB packet with a 106-tone RU at index 2 spans from roughly 2413 to

---

[1] This follows from Section 27.1.1 in [18].

2421 MHz. This significantly reduces the available bandwidth, as only 43.6% of the 234 data subcarriers as used by an HE SU packet are occupied. The IEEE 802.11be standard (Wi-Fi 7), which is expected to be finalized in 2024, also supports multi-RU allocation to a single STA to improve spectral efficiency when the number of users is small [19]. Then, if only one 26-tone RU—which is roughly 2 MHz bandwidth—has to be punctured to avoid interference, 84.6% of the data subcarriers can still be used[2].

In order to test different Signal to Interference Ratios (SIRs), we scaled the IQ samples accordingly. Additive White Gaussian Noise (AWGN) is added to the Wi-Fi packets, resulting in the Signal to Noise ratio (SNR) of 23 dB, at which level the simulation produces 0% Packet Error Ratio (PER) without interference. This SNR is chosen for a realistic simulation condition to showcase the impact on PER caused by IEEE 802.15.4 interference.

Furthermore, an HE packet consists of two parts: the pre-HE modulated part and the HE modulated part. The partial bandwidth only applies to HE modulated part. Hence even for 106-tone HE TB packets, the Wi-Fi preamble and header fields occupy the full bandwidth. Meaning that with strong interference, the packet may not be detected or part of the header—although it is using the lowest Modulation and Coding Scheme (MCS)—cannot be decoded. Thus, HE TB packets can still be affected by narrow band interference. We therefore make a distinction between packets for which only the data part is interfered with IEEE 802.15.4 and where both the preamble and data are interfered.

The simulation result is shown in Figure 2. Note that for HE TB packets (red) where only the data part is interfered, the PER is 0% and is therefore not visible. It can be seen that the PER for TB when also the header is interfered even at MCS 7 (the highest that *openwifi* supports) drops sharply around -1 dB SIR, which is the point where the packet can be detected. The result for TB packets with other MCSs is similar and therefore omitted. There is a large difference in the SIR between SU and TB at 10% PER, namely 23 dB for MCS 7 (blue) and 6-8 dB for MCS 3 (magenta). For SU, there is limited difference in performance when the full packet is interfered compared to when only the data is interfered. At a low SIR, the latter is worse, which can be explained by the fact that the channel estimation performed during the preamble does not yet include the interference, while it affects the data part, which is therefore not correctly compensated for.

The theoretical throughput of a 106-tone packet at MCS 7 with 3.2 µs guard interval is 31.9 Mbps, whereas that of an 242-tone (SU) packet is 73.1 Mbps. For SU at MCS 4, this is still 43 Mbps. For MCS3, it is lower than a 106-tone packet, namely 29.3 Mbps. Therefore, if the client would send SU packets with a fixed MCS of 4 (cyan), it is expected to still outperform the TB packets with MCS 7, even when every packet would be interfered with an SIR higher than 15 dB. However, if not all packets are interfered, a rate adaptation algorithm will likely try higher MCSs as well, which will then occasionally fail, leading to a lower overall throughput.

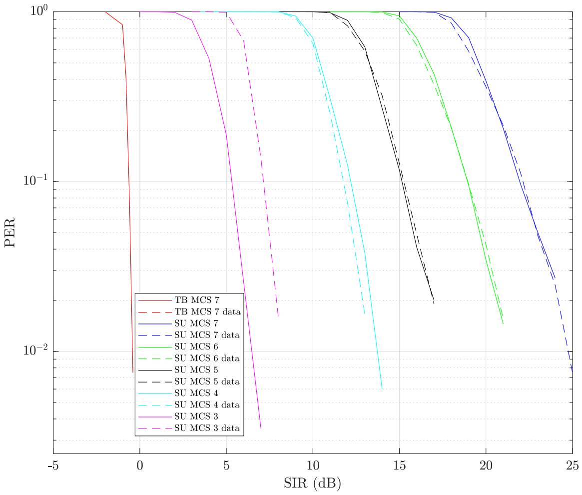

Fig. 2. Simulated Wi-Fi PER for different SIR with IEEE 802.15.4.

Furthermore, the spectral efficiency of SU and TB packets for the same MCS is roughly equivalent from a single user's perspective. Given a subcarrier width of 78.125 kHz, the data part of a 106-tone TB packet occupies 8.28 MHz bandwidth, while for a 242-tone SU this is 18.90 MHz. This results in a spectral efficiency of 3.85 and 3.87 (bit/s)/Hz, respectively[3]. If a lower MCS or retransmissions are needed due to interference for SU, this has a negative impact on its spectral efficiency.

V. EXPERIMENTAL PERFORMANCE

We implemented the OFDMA extension to *openwifi* on an AMD Zynq UltraScale+ MPSoC ZCU102 with AD9361 RF front-end and utilized the Zolertia RE-Mote running the RIOT operating system [20] as IEEE 802.15.4 transceivers, which are both available in the w-iLab.t testbed. With this we examined (*A*) the Wi-Fi PER between two SDR devices, (*B*) the Wi-Fi throughput under interference when communicating to a COTS STA, and (*C*) the PER of IEEE 802.15.4 when being interfered by Wi-Fi.

A. Wi-Fi PER between SDR devices

In order to have full control over the Wi-Fi configuration, we perform PER tests between two *openwifi* boards, when interference is generated using two IEEE 802.15.4 transceivers. In order to limit interference of other devices in the office environment, the devices are put relatively close to each other and the sensitivity threshold (the value below which the receiver will ignore an incoming signal) is set just above the Received Signal Strength Indicator (RSSI) for the Wi-Fi traffic. The experimental set-up is shown in Figure 3. In

---

[2] This is calculated as follows: $((106-4)+(52-4)+(26-2)+(13+13-2))/(242-8)\cdot 100\%$. Due to null subcarriers and pilots, not all subcarriers are used for data.

[3] This is calculated as the throughput mentioned above divided by the occupied bandwidth.

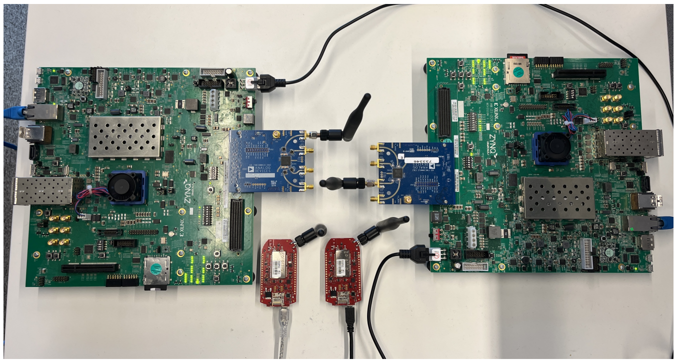

Fig. 3. Set-up consisting of two *openwifi* boards and two IEEE 802.15.4 transceivers.

order to create sufficient interference, the CCA threshold of the 802.15.4 transceivers is set to their maximum. Therefore, they won't back-off when sensing a busy channel, which mimics a hidden-node scenario. Furthermore, acknowledgments and retransmissions on the 802.15.4 transceivers are disabled to minimize waiting time, hence maximize the interferer's duty cycle. In this way, we send packets with a payload of 80 bytes at an interval of $15\,\mathrm{ms}$, where the receiver responds with a packet of equal size. This corresponds to an airtime of $3.83\,\mathrm{ms}$ per packet. As confirmed by capturing IQ samples when transmitting with the transceivers, there will be a delay of several microseconds between ping and reply. Meaning that during each $15\,\mathrm{ms}$, there is twice an idle time of around $3.67\,\mathrm{ms}$, giving a duty cycle of around 51%. The SIR is varied by changing the output power of the 802.15.4 devices and by relocating them. The SIR is measured as the difference between the RSSI of Wi-Fi and 802.15.4 packets, as determined by the RF front-end chip of the SDR. Given the slight deviation in SIR, the value is presented by a range.

Either HE SU or TB packets with MCS 7 are injected while both boards operate on Wi-Fi channel 1. The 802.15.4 transceivers are set to channel 11. In total for each SIR and packet format, five measurements with 50.000 Wi-Fi packets containing 600 payload bytes are executed. The results for SU (blue) and TB (red) with error bars depicting the standard deviation among the five measurements are shown in Figure 4. Infinite SIR here means there were no 802.15.4 packets sent.

It can be seen that in the worst case, there is around 38% PER for SU. According to the results in Figure 2, the PER at an SIR lower than $17\,\mathrm{dB}$ for MCS 7 should be 100%. However, since the duty cycle of the 802.15.4 traffic is around 51% and Wi-Fi still applies CCA, the actual PER is less than simulated, because the collision rate in the experiment is not 100%. As expected in the simulation, the PER of TB packets rapidly decreases to almost 0% (nearly invisible in the figure) for higher SIR. At 8.5-$11.5\,\mathrm{dB}$ SIR with 37% PER, considering that only 63% of the theoretical throughput can be reached, while failed packets still occupy the medium, the spectral efficiency of SU has dropped to around $2.44\,(\mathrm{bit/s})/\mathrm{Hz}$, while for TB it remains around $3.85\,(\mathrm{bit/s})/\mathrm{Hz}$.

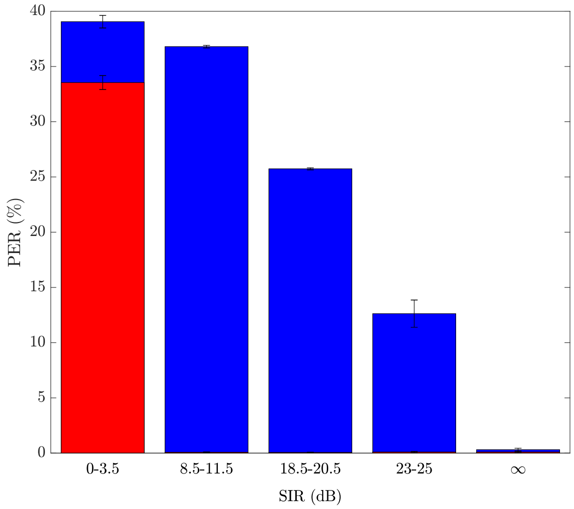

Fig. 4. Wi-Fi PER when sending HE SU packets (blue) or enabling RU puncturing by sending HE TB (red) frames with MCS 7 at different SIR.

B. *Wi-Fi throughput between SDR and COTS STA*

Next we investigate the end-to-end performance of RU puncturing to avoid interference using a COTS 802.11ax STA and an *openwifi* AP. The STA used is a Dell Latitude 5511 running Ubuntu 22.04 with Linux kernel version 6.2.0, which has an Intel AX201 Wi-Fi 6 chip.

During this experiment, the *openwifi* board acts as AP and the STA connects to it. The STA then sends UDP packets of 600 bytes using iPerf for 100MB in total. This is then repeated five times. Meanwhile, just like in Section V-A, 80-byte IEEE 802.15.4 packets were exchanged at an interval of $15\,\mathrm{ms}$. The measured SIR was in the range of 8-$16\,\mathrm{dB}$. When sending SU packets, the STA usually reserves $1158\,\mathrm{\mu s}$ of airtime using a request-to-send packet. We therefore set a similar uplink length in the TF, namely 6 times 600 bytes (to allow for aggregation), which results into $1118\,\mathrm{\mu s}$. This is about one third of the idle time in between the 802.15.4 packets. Since there is a known high traffic load, we let the AP send TFs at a minimum interval of $1\,\mathrm{ms}$, meaning as fast as possible when it has the chance.

The TF contains a field "CS required", which indicates whether the STA should perform carrier sensing before transmitting and cancel the transmission when the channel is busy. We set this to 0, meaning the STA is not required to do this. However, the AP still has to perform CCA before sending the TF. This is comparable to the procedure of sending HE SU PPDU, where the station first sends a CTS packet after performing CCA.

From traces captured using `tcpdump`, it becomes clear that the STA does not always respond with an HE TB after the AP sends a TF. Furthermore, since the AP has to apply CCA before sending the TF, it cannot reach the minimum interval we set. The result is that with interference, out of all correctly received packets, on average 25.5% is an HE TB packet, while the remaining are sent as HE SU.

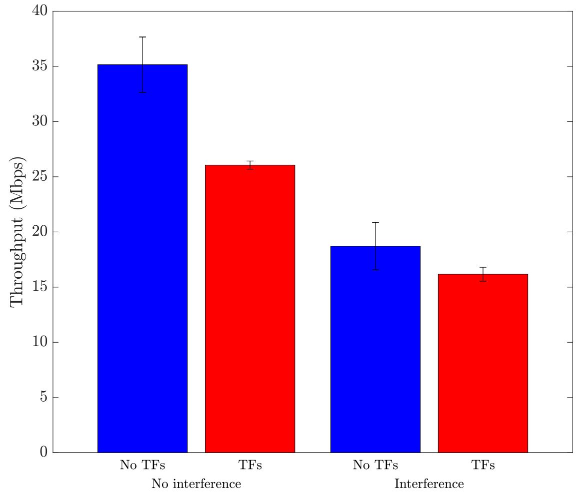

Fig. 5. Wi-Fi throughput with and without IEEE 802.15.4 interference, when RU puncturing is enabled (with TF) or disabled (without TF).

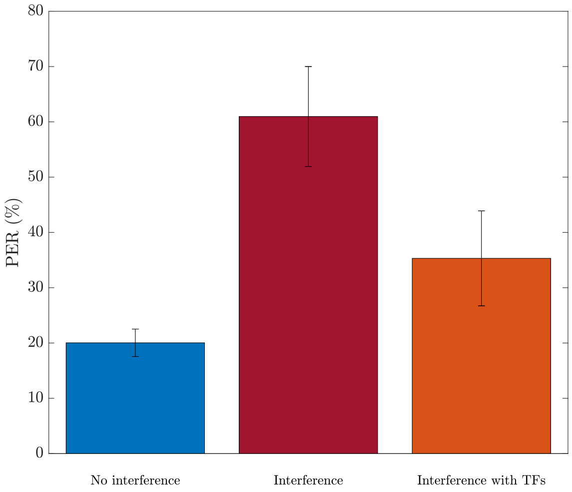

Fig. 6. IEEE 802.15.4 PER without Wi-Fi interference (left) and with interference, without RU puncturing (middle) and enabling RU puncturing with TFs (right).

The throughput results with error bars depicting the standard deviation among the measurements are presented in Figure 5. We observe the difference caused by interference in Wi-Fi throughput is significant when the traffic consists of only SU packets (it drops by around 50% when comparing the blue bars). In contrast, when RU puncturing is enabled on roughly 25% of the packets, the throughput only drops about 30% (when comparing the red bars). Still, the overall throughput is higher in all configurations if only using SU. This is because SU has higher throughput than TB, and our interference has a duty cycle of only 51%, hence there is sufficient chance for SU packets to be received without collision.

C. IEEE 802.15.4 PER when interfered by Wi-Fi

Lastly, we compare the PER when "pinging" between the two 802.15.4 transceivers on channel 11 in an office environment with identical configuration except for lowered transmit power. Additional Wi-Fi interference on channel 1 is created by the COTS STA which connects to the *openwifi* AP. The STA then continuously sends Wi-Fi packets to the AP. Ten measurements with 5.000 packets of 80 bytes at a 15ms interval between the 802.15.4 transceivers are performed with and without Wi-Fi interference, either when letting the AP send TFs for a 106-tone at index 2 or not.

The SIR over Wi-Fi is estimated by capturing the IQ samples by the *openwifi* board at the same location as the 802.15.4 devices. The recorded SIR is around $-25\,\mathrm{dB}$, which shows that 802.15.4 packets are strongly interfered.

The results of PER with error bars depicting the standard deviation between measurements is shown in Figure 6. It can be seen that even without induced interference, there is a PER of around 20%, which may be caused by other devices in the office environment and the lack of CCA and retransmissions. With induced Wi-Fi interference, the PER is significantly increased to around 60% when no RU puncturing is used (i.e. only SU packets are sent). While when RU puncturing is enabled on a portion of the packet, the PER drops to around 35% despite of the strong interference level. This shows that indeed also the 802.15.4 transceivers suffer from interference from Wi-Fi and this can be partly overcome when TFs are sent to induce TB packets.

VI. DISCUSSION AND FUTURE WORK

This experimental study consists of two parts. First we use two *openwifi* boards, where we fully control the types of packets as well as the MCS used in Wi-Fi traffic; then we replace one board by a COTS device, and observe that the COTS device only reacts to trigger frames with a TB packet about 25% time, and it uses SU packets with variable MCS. From the fully controlled experiment, we show great advantage in terms of PER and spectral efficiency for interference avoidance using the OFDMA feature of Wi-Fi 6. In the test with a COTS device, we can no longer estimate spectral efficiency easily, however we do observe with RU puncturing enabled on a portion of the packets, the Wi-Fi throughput is less impacted by IEEE 802.15.4, and vice versa from IEEE 802.15.4's perspective.

Though we also admit the throughput from a single Wi-Fi client is worse when enabling RU puncturing. This can to a great extent be explained by the fact that only 43.6% of the available subcarriers are used when avoiding interference. With Wi-Fi 7, this can be greatly improved as multi-RU transmissions are supported towards a single user. Also, in a multi-user scenario, the utilization of subcarriers can be improved, such that less bandwidth is wasted to avoid a narrowband interferer. If not all users experience the interference, even the full bandwidth can be used when allocating the RUs to the right user. However, then there needs to be a feedback mechanism for the state of interference such that the transmitter can schedule transmissions accordingly. For uplink OFDMA, STAs usually already send buffer status reports to inform the AP about the traffic in its queue. However, in case interference is only intermittently present, ideally the feedback should include very recent information or a predictive algorithm should be used. Optimizing when to apply RU puncturing, especially in a multi-user scenario, in combination with a suitable interference detection approach is left for future work.

Moreover, OFDMA scheduling is generally a hard problem and may not always result in better performance, as shown by [21], [22]. There are several configuration options in the TF, like uplink length, MCS, CS required, target receive power, etc. This makes it a multi-dimensional optimization problem. Various vendors likely have different scheduler implementations that perform differently.

Furthermore, the rate control algorithm used can be made interference-aware. Traditional algorithms will typically lower the rate when packet loss occurs, but this is not necessary if the source of packet loss was interference that can be avoided.

Lastly, this research did not consider the impact of interference avoidance with OFDMA on latency. Packet loss leads to retransmissions which can greatly impact latency and jitter. Since OFDMA already provides opportunities for low-latency networks, this remains an interesting aspect to explore for future work.

VII. CONCLUSION

In this work we have examined the performance of interference avoidance using OFDMA by RU puncturing. Our simulations show that the PER of Wi-Fi can be greatly reduced when leveraging this under narrowband interference. We verified this with over-the-air experiments on SDR boards using *openwifi*, an FPGA-based Wi-Fi transceiver, which we extended with the 802.11ax compliant OFDMA feature. Mitigating interference in this way improves the spectral efficiency, however, it should be exploited according to the interference scenario in order to have a positive impact on overall throughput. As we show by another experiment in a single-user scenario with a COTS 802.11ax device, this is not yet the case if interference is intermittent and has relatively low duty cycle. When utilizing OFDMA the impact of interference on other technologies can be reduced as well, as we experimentally show by the PER of IEEE 802.15.4 packets. Given these results, there is potential for efficient interference avoidance using OFDMA. Some considerations, like making most use of the unaffected subcarriers and applying RU puncturing at the right moment, need to be taken into account to make full use of it.

ACKNOWLEDGMENT

This work is partially funded by the Flemish FWO SBO S003921N VERI-END.com project and imec ICON project VELOCe - VErifiable, LOw-latency audio Communication (Agentschap Innoveren en Ondernemen project nr. HBC.2021.0657).

REFERENCES

[1] "Thread," https://www.threadgroup.org/, Thread Group, 2023, [Online; accessed 24-November-2023].

[2] "Matter Executive Overview," https://csa-iot.org/developer-resource/matter-executive-overview/, Connectivity Standards Alliance, 2022, [Online; accessed 24-November-2023].

[3] M. Z. Mahfouz, A. B. J. Kokkeler, A. Meijerink, and A. A. Glazunov, "Impact of Ultra-Narrowband Interference on Wi-Fi Links: An Experimental Study," *IEEE Transactions on Wireless Communications*, vol. 20, no. 5, pp. 3016–3030, 2021.

[4] X. Jiao, W. Liu, M. Mehari, M. Aslam, and I. Moerman, "openwifi: a free and open-source IEEE802.11 SDR implementation on SoC," in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*. IEEE, 2020, pp. 1–2.

[5] "OpenWiFi - How to use ZYNQ SDR in Linux mode," https://doc.ilabt.imec.be/ilabt/wilab/tutorials/openwifi.html, imec iLab.t, [Online; accessed 19-December-2023].

[6] A. M. Mamadou and G. Chalhoub, "Enhancing the CSMA/CA of IEEE 802.15.4 for better coexistence with IEEE 802.11," *Wireless Networks*, vol. 27, pp. 3903–3914, 2021.

[7] J. Bauwens, B. Jooris *et al.*, "Coexistence between IEEE802.15.4 and IEEE802.11 through cross-technology signaling," in *2017 IEEE INFOCOM WKSHPS*, 2017, pp. 529–534.

[8] R. Chen and W. Gao, "Enabling Cross-Technology Coexistence for Extremely Weak Wireless Devices," in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, 2019, pp. 253–261.

[9] P. Li, Y. Yan *et al.*, "Coexist WiFi for ZigBee Networks With Fine-Grained Frequency Approach," *IEEE Access*, vol. 7, pp. 135 363–135 376, 2019.

[10] A. Hithnawi, S. Li *et al.*, "CrossZig: Combating Cross-Technology Interference in Low-Power Wireless Networks," in *2016 15th ACM/IEEE IPSN*, 2016, pp. 1–12.

[11] E. Yaacoub and Z. Dawy, "Interference mitigation and avoidance in uplink OFDMA with collaborative distributed intracell scheduling," *AEU - International Journal of Electronics and Communications*, vol. 65, no. 11, pp. 937–941, 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1434841111000707

[12] D. Croce, P. Gallo *et al.*, "ErrorSense: Characterizing WiFi error patterns for detecting ZigBee interference," in *2014 IWCMC*, 2014, pp. 447–452.

[13] J. Yao, W. Lou *et al.*, "Mitigating Cross-Technology Interference Through Fast Signal Identification," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 2, pp. 2521–2534, 2023.

[14] L. Tytgat, O. Yaron *et al.*, "Avoiding collisions between IEEE 802.11 and IEEE 802.15.4 through coexistence aware clear channel assesment," *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, 04 2012.

[15] C. P. Robinson, D. Uvaydov *et al.*, "Narrowband Interference Detection via Deep Learning," in *ICC 2023*, 2023, pp. 6379–6384.

[16] K. Davaslioglu, S. Soltani *et al.*, "DeepWiFi: Cognitive WiFi with Deep Learning," *IEEE Transactions on Mobile Computing*, vol. 20, no. 2, p. 429–444, feb 2021. [Online]. Available: https://doi.org/10.1109/TMC.2019.2949815

[17] E. D. Salik, G. Görbilek *et al.*, "Non-WiFi Interference Detection and Throughput Estimation at the WiFi Edge for 2.4 and 5 GHz Bands with Machine Learning," in *IEEE BlackSeaCom 2023, Istanbul, Turkey, July 4-7, 2023*. IEEE, 2023, pp. 371–378. [Online]. Available: https://doi.org/10.1109/BlackSeaCom58138.2023.10299696

[18] "Enhancements for High-Efficiency WLAN," *IEEE Std 802.11ax-2021 (Amendment to IEEE Std 802.11-2020)*, pp. 1–767, 2021.

[19] E. Khorov, I. Levitsky, and I. F. Akyildiz, "Current Status and Directions of IEEE 802.11be, the Future Wi-Fi 7," *IEEE Access*, vol. 8, pp. 88 664–88 688, 2020.

[20] E. Baccelli, C. Gündoğan *et al.*, "RIOT: An Open Source Operating System for Low-End Embedded Devices in the IoT," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4428–4440, 2018.

[21] T. Oogami, H. Tamura *et al.*, ""Experimental Evaluation of Uplink Communication Performance in IEEE 802.11ax Wireless Local Area Network: OFDM vs. OFDMA," in *Advances in Intelligent Networking and Collaborative Systems*, L. Barolli, Ed. Cham: Springer Nature Switzerland, 2023, pp. 495–504.

[22] D. Weller, R. D. Mensenkamp *et al.*, "Wi-Fi 6 performance measurements of 1024-QAM and DL OFDMA," in *ICC 2020*, 2020, pp. 1–7.