Latency-Sensitive Networked Control Using 802.11ax OFDMA Triggering

Karl Montgomery, Mohamed Kashef (Hany), Richard Candell National Institute of Standards and Technology (NIST), Gaithersburg, MD, USA Emails: {karl.montgomery, mohamed.kashef, and richard.candell}@nist.gov

Abstract—Orthogonal Frequency-Division Multiple Access (OFDMA) is a feature implemented in IEEE 802.11ax (Wi-Fi 6) that aims to improve performance by sending packets simultaneously to multiple users. This paper highlights performance issues when using OFDMA with a real-time control application based on the commonly used transmission control protocol (TCP). OFDMA is designed to send data to multiple users simultaneously with reduced latency by grouping many subcarrier frequencies into resource unit (RU) blocks for each user. However, we found that when using a TCP data stream subject to a variable update rate from late packets, the OFDMA algorithm under test was not sufficient to improve performance, and instead increased latency. This surprising finding may pose a concern for industrial applications that rely on TCP transport. Experiments were performed to compare the latency and physical performance results of OFDMA enabled, OFDMA disabled, and our previous works which utilized wireless time-sensitive networking (TSN) features implemented in software. The previous wireless TSN results demonstrated close to deterministic latencies for a TCP link subject to inter-network interfering traffic.

Index Terms—OFDMA, TSN, Factory communications, IEEE 802.11, IEEE 802.1Qbv, WLAN

I. INTRODUCTION

The demand for wireless connectivity is increasing in many sectors of the industry. The ease of installing wireless nodes and mobility gains provide advantages over legacy wired communication systems. With these advantages comes the cost of latency and reliability issues, which typically arise when interfering traffic is transmitted in overlapping frequency bands. For many industrial applications, communications must arrive on time, with low latency and jitter. Thus, wireless has not been adopted for many real-time control and other latencysensitive applications. To remedy this, technologies such as wireless time-sensitive networking are being developed for use in Wi-Fi. With the introduction of commercially available IEEE 802.11ax devices and access points (APs), OFDMA has been purported to improve performance for latency-sensitive data. However, as described in this work, we discovered evidence to the contrary.

A. OFDMA in Wi-Fi 6

OFDMA is implemented in commercially available IEEE 802.11ax (Wi-Fi 6) APs and user devices, where multiple stations connected to the AP may send packets simultaneously on RU blocks that occupy different portions of the available frequency band. The AP signals stations by sending OFDMA

trigger frames, indicating which users can send or receive data packets on their allocated RU blocks for the next transmission period. In practice, OFDMA allows for simultaneous transmissions across frequency, reducing the need for sequential transmissions to multiple stations. An illustration of previous Orthogonal Frequency Division Multiplexing (OFDMA) compared to the OFDMA mechanism, published in [1], can be seen in Fig. 1. In this previous work, we discuss details regarding OFDMA operation, resource units, and the OFDMA trigger conditions with traffic loads using the same wireless equipment used here. With our equipment, we found the OFDMA trigger condition occurs when two or more stations transmit over 100 packets per second (pps).

B. Dual-Lift Industrial Robotic Use Case

The Dual-Lift industrial robotic use case is a real-time control application, in which leader and follower robots coordinate to lift an object. The dual-lift use case components of the NIST Industrial Wireless Testbed are shown in Fig. 3. The use case is implemented using the robot operating system (ROS) with three nodes: the supervisor, leader, and follower. The supervisor node coordinates the leader and follower robots to pick up the bar and implements safety measures to stop the robots in the case of a communication loss from the leader or follower node. The leader and follower nodes communicate using Ethernet to each respective robot controller. The physical performance of the use case was determined by measuring the position of the follower and comparing it with the position of the leader on the circular path. The difference in positions, after calibrating for the bar length, was named the leaderfollower "Cartesian error" metric. Under ideal communications, the Cartesian error of the leader and follower spans the range of 13-15 mm when the leader is moving at 150 mm/s. At this speed an additional communications delay of just 8 ms corresponds to an increase of Cartesian error of 1.2 mm. The follower lag is by design and a result of the proportional velocity controller at the follower's ROS node. The ROS topic, "desired pose" is published by the leader and subscribed to by the follower. The follower moves towards the latest update to the desired pose data containing the targeted position and orientation for the follower. For a more detailed description of the real-time control use case and messaging flows, refer to [3], which also discusses our first iteration of wireless TSN with the same use case.



Fig. 1: Diagram of OFDM compared to OFDMA. OFDMA allows for simultaneous transmission across different RUs.



Fig. 2: The wireless TSN schedule, from [2], designed to accommodate the protected leader-follower TCP-based traffic for the dual-lift use case. The time critical (TC) window represents the protected window for only the leader and follower position traffic, the best-effort (BE) window for the background and interfering traffic, and the guard band (G) used for blocking all traffic that protected the next TC window from the last BE transmission.



Fig. 3: Picture of Leader-Follower Dual-Lift robotic use case. This is a real-control application with strict latency and jitter requirements. An overhead (not shown) infrared camera tracking system records the position of the robots' grippers as a ground truth measurement.

C. Review of Previous Works

There has been work showing close to deterministic latency through a software-based wireless TSN implementation. Ethernet-wireless adapters were used to apply an optimized time-aware schedule to the traffic [2], and the schedule that was used there is reproduced here in Fig. 2. The schedule had some limitations as the queuing for network packets was gated before the physical (PHY) layer of Wi-Fi. Therefore, there was a measurable overlap in time as the best-effort traffic violated the protected window. Despite the overlap issue, the work achieved a promising result for applying wireless TSN for reliable communications. The TSN schedule was tuned by varying the protected window period and cycle time for the schedule to better accommodate the acknowledgment (ACK) from the leader-follower TCP communication link. Also, in [2], tuning by limiting the token and limit buffer sizes of the traffic queues greatly reduced an issue with best-effort traffic overlap.

OFDMA in IEEE 802.11ax has been introduced to offer various access advancements over previous WiFi generations. These advancements in OFDMA include refining resource allocation, allowing lower channel access delay compared to carrier sense multiple access (CSMA), and improving the overall multi-user performance. The analysis of OFDMA performance in multi-user scenarios with latency requirements can be found in [4]–[7], and the references therein. In [4], a computational model for the OFDMA throughput and multiuser efficiency is presented to minimize the number of required retransmissions to make OFDMA more suitable for real-time applications. In [5], a simulation-based study of the uplink multi-user OFDMA performance is presented to demonstrate the impact of the number of users on the average network delay in dense and dynamic deployments. In [6], the system efficiency and delay performance of uplink random access OFDMA is studied analytically using simulations. In [7], the impact of the uplink scheduler on the delay and throughput

performance was highlighted. The scheduler allocated the size of the resource units while keeping a contention-free medium for real-time application latency and jitter guarantees.

The challenges of using trigger-based OFDMA for real-time applications was discussed and demonstrated in [8], where an uplink resource allocation algorithm is proposed to satisfy the real-time applications requirements of a 1 ms delay with the probability of 99.999% using IEEE 802.11ax. Another resource allocation strategy is introduced in [9] where an advanced OFDMA-based uplink random access scheme is simulated to optimize real-time responsiveness in industrial wireless networks. In [10], a cooperative resource scheduling scheme is presented to allow the users to share medium access control information to minimize latency for time-sensitive applications. Further optimization approaches and challenges for using OFDMA in real-time applications can be found in [11] and the references therein. However, all these OFDMArelated works discuss results that were obtained analytically or used various simulation tools; however, the analyses do not measure OFDMA's impact on the operational performance of real-time applications.

D. Motivation and Contributions

This work is primarily motivated to examine claims regarding performance increases with OFDMA. In previous works, we have demonstrated the dual-lift use case as a latencysensitive real-time control application, for which a reliable and low-latency communications system is required to maintain the desired performance. For wireless to be more widely adopted for control, we believe OFDMA should be vetted using actual hardware, not just in simulation. Our contributions to this paper are as follows:

1) Technology Demonstration: It demonstrates how OFDMA may be utilized for a typical industrial control process, with a leader-follower dual-lift use case as a latency-sensitive example.

2) *Protocol Asynchrony:* An asynchrony is exposed between the 802.11ax trigger frame control and the high-layer protocols of the host computing devices.

3) Comparison to Wireless TSN: Results obtained from the OFDMA experiments are compared with previous wireless TSN results using the same use case presented here. We demonstrate that wireless TSN achieved close to deterministic performance.

4) 802.11 Recommendations: Recommendations are made to improve future iterations of IEEE 802.11, such as adding OFDMA scheduler timing designed for TCP traffic loads and allowing for tunable OFDMA trigger conditions.

II. IEEE 802.11AX OFDMA PROTOCOL

OFDMA is a significant milestone in the evolution of the IEEE 802.11 protocol. It enables the simultaneous transmission of a physical layer protocol data unit (PPDU) on the downlink (DL) from the AP to two or more stations. Similarly, OFDMA allows PPDUs to be transmitted simultaneously from two or more stations to the AP on the uplink (UL). The



(a) Downlink



(b) Uplink

Fig. 4: The PPDU message flows for the IEEE 802.11ax OFDMA uplink (a) and downlink (b) message flows.

real-time control application presented here relies on both uplink and downlink messaging. The uplink triggering is of particular concern since the control communications sequence is initiated by the leader sending the desired pose to the follower. The follower then responds with a TCP acknowledgment (TCP_ACK). Thus, we have a physical layer uplink frame, a downlink frame, another uplink frame, and the final downlink frame to accommodate a single TCP transaction. Since the TCP network protocol makes the leader wait to send the next data packet until the last ACK is received, the entirety of the four-link OFDMA exchange must occur within the 8 ms scan window of the robotic application to maintain the 125 Hz update rate.

A. OFDMA Modulation and Coding Schemes in 802.11ax

OFDMA is a variant of orthogonal frequency division multiplexing (OFDM) that groups orthogonal sub-carriers together and assigns each group to a specific station at any given time. In OFDMA, the allocation of sub-carriers is managed by the AP and assigned to stations through control messages called trigger frames. Wi-Fi 6 operates in the 2.4 GHz and 5 GHz ISM bands, while Wi-Fi 6e extends channels to 6 GHz. In Wi-Fi 6, the physical channel size can be configured as 20, 40, 80, or 160 MHz. The physical channel can then be subdivided into 78.125 kHz sections, with 26, 52, 106, 242, 484, or 996 frequency resource units (RUs) available for carrying user data. The number of RUs is determined by the physical channel size. Each OFDM symbol has a duration of 12.80 μ s, and the guard interval can be 0.8, 1.2, or 3.2 μ s, yielding an effective bit rate that depends on the number of users and the modulation and coding scheme. In OFDMA, multi-user PPDU (MU-PPDU) frames use these RUs to carry data and control instructions for each station being addressed. The resulting protocol behaves very similarly to a broadcast. Each user encodes or decodes its data to or from the MU-PPDU frame.

B. 802.11ax OFDMA Message Flow

The Wi-Fi 6 downlink protocol message flow is illustrated in Fig. 4(a). In an uplink message flow, the AP broadcasts a Multi-user (MU) request-to-send (MU-RTS) addressed to client stations indicated by the PPDU. Each station responds with a clear-to-send (CTS), to which the AP responds with the MU-PPDU carrying the user data to the stations, followed by a block acknowledgment request (BAR). Each station then responds to the BAR with an acknowledgment (ACK). Each frame is separated by inter-frame spacing (IFS) of various types to allow for the processing of messages by receiving devices before being required to act. In 802.11ax, the short IFS (SIFS) following a frame is 16 μ s, which is defined as the amount of time required for a wireless device to process a received frame and begin to respond. The arbitration interframe spacing (AIFS) is a variable IFS used to indicate the time to arbitrate message queue priorities before the next transmission.

Similarly, the corresponding uplink trigger frame protocol is shown in Fig. 4(b). The MU-OFDMA frame sequence begins with a buffer status report poll (BSRP) from the AP. Each station responds with a buffer status report indicating that data is pending in its transmission queue. The AP sends a MU-RTS, and each station responds together with a CTS. The AP sends an uplink trigger that signals the stations to transmit their data, and each station responds with an OFDMA UL-PPDU utilizing their allocated RUs. The AP finalizes the transaction with an MU-Block ACK (MU-BA). This MU-BA is used for analysis in Section IV-A2.

C. Activation of OFDMA Trigger Frames

Our testing of two individual AP devices indicated that OFDMA is activated when an aggregate of 100 PPS traverses the AP with at least two stations (i.e., spatial streams) active. If the aggregate packet rate drops below this threshold, OFDMA is deactivated and OFDM with CSMA behavior is resumed. The exact threshold that determines when OFDMA is activated is not explicitly mentioned in the literature or within the specification. Since both vendors' APs evaluated in this work activated OFDMA with the same trigger condition, it is safe to assume that this OFDMA activation threshold could be an agreement adopted by other Wi-Fi equipment manufacturers.

III. EXPERIMENTAL METHODOLOGY

A. Measurement Methods

The leader-follower dual-lift use case experiments were measured by the physical robot positions and network captures. The physical positions of the robot's end effectors were captured by four Optitrack 13W infrared cameras, which



Fig. 5: Network diagram of Dual-Lift Robotic use case, with the wired and wireless operational networks. The leader ROS node captured packets on its wireless interface for latency measurements. The wireless sniffer, which captured the OFDMA trigger frames is shown.

comprise the 3D tracking vision system above the testbed, mounted to the laboratory's ceiling on a custom-built and fixed gantry system. An uncertainty analysis was performed to obtain the uncertainty of the Cartesian error between the leader and follower robots, and was shown to be 0.027 mm at a 99% confidence level in [12], a greater than a 50x improvement from using the positions reported by both robot controllers. The robot controllers' reporting rates could not be synchronized, which previously led to a much higher uncertainty for the Cartesian error metric. The Cartesian error, as defined in Section I-B, was calculated using the circular sequence portion of the use case. The calculation involved calibrating the positions of the robots when the bar is initially picked up and then computing the Cartesian Error by taking the Cartesian magnitude, in millimeters, between the leader's calibrated position on the circular path and the follower's current position.

The network captures mentioned consist of the leader ROS node's wireless interface and wireless sniffer captures. These devices are shown in the operational network diagram, seen in Fig. 5. The leader node locally captured its network data from its Intel AX210 Wi-Fi 6 interface to later compute the round-trip time (RTT). In previous works, Ethernet-wireless adapters served as the ROS nodes' wireless interfaces, with network taps inserted in Ethernet links for network capture. In this work, our ROS nodes are directly connected wirelessly to the Wi-Fi 6 AP. Since there is no wired tap in the network, packets are captured locally at the leader's wireless interface. Therefore, the uncertainty of the measurement of the RTT depended on the clock used by the leader node. The leader node was synchronized over Ethernet to a local grand leader (GL) clock using the Precision Time Protocol (PTP). The offsets were measured for both the physical hardware clock to the GL clock on the Ethernet port and the internal system clock to the physical hardware clock. The uncertainty of the RTT measurement, from the PTP synchronization error reports was calculated to be 270 ns at a 99% confidence level. The wireless sniffer was used to capture OFDMA trigger frames, the associated "request to send" and "clear to send" messages, all of which were sent from the AP and stations (STAs). The sniffer captures were used to measure OFDMA-related cadences and to confirm when OFDMA was activated by observing the OFDMA trigger frames.

The Key Performance Indicators (KPIs) in this paper are the same as previous work utilizing the leader-follower duallift use case, in [2] and [3]. The KPIs for this paper are the Cartesian error and the round trip time (RTT) metrics. Note that the Cartesian error is directly impacted by the RTT, as higher latencies from the leader and follower lead to higher Cartesian error. The sniffer data is used to extract the time differences between MU-Block ACKs sent from the AP, corresponding to data sent from the TCP leader-follower transaction. We filtered for the type of packet and destinations being "Broadcast" or the leader's address. The time differences from the filtered sniffer captures yield the cadence of the Block ACKs related to the uplink for the leader ROS node data, which is shown in the OFDMA uplink messaging diagram, Fig. 4b as "MU-BA".

B. Experimental Methodology

For the results in this paper, an IEEE 802.11ax AP was operated with and without OFDMA enabled. We used the 2.4 GHz band at channel 11. To activate OFDMA, the channel bandwidth was set to 40 MHz on the AP. Theoretically, there is no reason, from the protocol perspective, that 20 MHz should not work; however, we were not able to activate the conditions for OFDMA at 20 MHz despite using up to three stations that sent high rates of traffic. For the experiments, we used a TPlink Archer AXE300, which is a Wi-Fi 6E-capable router. We also experimented with a different AP, an Asus RT-AX82U, to confirm the latency issues observed were not hardwarespecific. The Asus and TP-Link APs yielded the same trigger condition for OFDMA: two or more stations must send greater than 100 pps, regardless of packet size [1]. We believe this was an agreement between vendors, as we did not observe any specification for this trigger condition within the IEEE 802.11ax protocol. With the OFDMA trigger condition stated, our use case activates OFDMA without any other traffic on the wireless network, as the leader and follower nodes qualify as two stations that send a TCP-based 125 Hz traffic stream. However, as discussed later, if the TCP-ACK from the follower node is received by the leader node after the 8 ms threshold, the update rate would be lower than 125 Hz and would likely fall below the 100 Hz OFDMA trigger condition. We postulate the fluctuating update rate as one hypothesis for the latency increases observed.

As described in the following results, we experimented with varying uplink user datagram protocol (UDP) traffic levels from a single iPerf [13] source, at 0, 8, 16, 32, 64, 80, 128, 160, 200, and 240 Mbps with 1000 Byte length packets. For the OFDMA results, we set the AP to "OFDMA only" mode without multi-user multi-input multi-output (MU-MIMO), as

MU-MIMO would prevent OFDMA from activating. When OFDMA was off, the IPerf source was only able to transmit up to 200 Mbps when the use case was also running. This was due to the contention for the channel. Without the use case in operation, the maximum supported bit rate through the iPerf stream was 270 Mbps. We decided to use a maximum of 240 Mbps for the uplink bitrate using OFDMA, due to the high physical jitter observed from the follower robot's movement.

IV. RESULTS

In this section, we present the results from our experiments utilizing the Dual-Lift use case. We then discuss how the TCP traffic in the use case is hindered by OFDMA, compared to when OFDM with CSMA is used. We infer several reasons for the observed latency increase with OFDMA. Previous results from [2] are presented, which achieved close to deterministic performance using wireless TSN scheduling. Our experiments were conducted using the 2.4 GHz frequency band, as well as the 5 GHz band. Also, additional stations carrying background traffic were experimented with. However, no significant performance changes were observed using the 5 GHz band or when multiple background traffic sources were used, compared to the case of a single background traffic source at 2.4 GHz. Therefore, the results presented here represent the leader-follower use case with a single interfering station communicating through a single 2.4 GHz Wi-Fi channel at 40 MHz.

A. Exposition of Measurements

The Cartesian error and RTT plots are presented in the form of cumulative distribution function (CDF) plots. Under ideal communications, both the Cartesian error and RTT plots should have close to a vertical rise with short tails. The gentler the slope of the CDF curves, the greater the standard deviation in the data (jitter), which correlates to an increased Cartesian error between the leader and the follower. It was calculated that every 8 ms of latency added in the leader-follower communications loop corresponded to an additional 1.2 mm of error for the follower. Under ideal communications, the error should fall between 13-15 mm. We chose a threshold for the ideal performance of the use case such that 95% of Cartesian error falls below 15.7 mm, as described in [2], marked with a vertical dashed line in the Cartesian error plots.

1) OFDMA Mode Comparisons: In Fig. 6, four cumulative distribution function (CDF) plots show the Cartesian error and RTT results for both OFDMA enabled and disabled at the AP. The CDF curves from the Cartesian error and RTT plots for the OFDMA enabled (on) cases represent increased latencies. The slopes for the OFDMA enabled case are shallower than OFDMA disabled (off), representing a larger deviation in the tracking performance of the follower. The larger deviation manifested as increased jitter and lag in the follower robot's movement while tracking the leader on the circular path. For OFDMA enabled, the minimum latency and error performance



Fig. 6: Cartesian Error and Round Trip Time for the OFDMA On and Off Cases. Cartesian error with a vertical dashed line for the 15.7 mm error threshold is shown in both the (a) and (b) graphs. The RTT is shown in the lower graphs (c) and (d). Note the RTT scale differences in (c) and (d), where (d) has significantly higher RTT values.

was with 128 Mbps of uplink traffic. The reason that additional traffic improves performance is correlated with the increased rate of OFDMA trigger frames measured, which could lower the latency of the leader-follower ROS transmissions.

One might assume that the lowest latency case should occur without any interfering traffic; however, without additional uplink traffic, the use case experienced abnormally high RTT latencies and Cartesian error. As expected, when OFDMA is disabled, the best performer was the baseline case without background traffic. As the traffic increases, the error and RTT curves shift from the left to the right. However, for OFDMA, the baseline case without the interfering traffic is surprisingly second from the right. As traffic increased, except for the 240 Mbps case, performance improved. The Cartesian error and RTT results are correlated, as shown in Fig. 7, using data points from the 95th percentiles from the charts in Fig. 6. When OFDMA is active, RTT is lowered and Cartesian error is improved as uplink traffic is introduced, up to 128 Mbps. After 128 Mbps, performance degraded again, as expected, as the aggregate bit rate approached its theoretical channel limit. When OFDMA is turned off, we saw improved performance for all traffic cases here and performance steadily degraded as more traffic was sent.

2) Regarding Increased Latency with OFDMA: The data presented thus far supports the hypothesis that the AP could be switching between OFDMA on and off for the leader-follower traffic. Since the TCP rate fluctuates, RTT delays above 10 ms reduce the effective rate below the 100 pps threshold of the

AP's OFDMA trigger condition. Then, adding an uplink traffic stream could increase activation of OFDMA. This hypothesis explains why adding a traffic stream improved performance, due to an increased amount of OFDMA trigger frames.



Fig. 7: Statistical summary of all 95th percentile CDF values of Cartesian Error and RTT for uplink traffic.

The OFDMA trigger-switching hypothesis is further supported by the cadence of the MU-Block ACKs to the leader. For example, without the uplink traffic stream, the average time between MU-Block ACKs was 8.55 ms with a standard deviation of 8.22 ms. For the case of a moderate traffic load, injecting 64 Mbps of uplink traffic, the average time improved to 6.28 ms with a standard deviation of 6.51 ms. Thus, there appeared to be more uplink trigger frames to carry the leader's uplink data. Although we do not have direct evidence of the OFDMA switching on and off during experiments due to the black box nature of the AP, the data presented corroborates the hypothesis.

Another hypothesis is that there could be a wireless driver or kernel issue causing delays at the leader and follower ROS nodes. While unlikely, this possibility has not been eliminated as other machines functioning as the ROS nodes were not experimented with. It is worth noting that the interference source station did not experience as high latencies as the leader and follower ROS nodes. A fundamental difference between the traffic source station and the leader/follower stations was that the uplink traffic station sent UDP packets at a much higher rate than 100 pps, compared to the leader and follower nodes, which attempted a TCP rate of 125 Hz.

The final hypothesis regarding the OFDMA latency issues presented in this paper is that the AP itself could be buffering downlink data frames through frame aggregation to optimize channel efficiency. The buffering could be introducing additional delays to the TCP traffic stream. In [14], a work on TCP-aware OFDMA transmissions depending on traffic load, the authors wrote that the IEEE 802.11ax does not specify a method to synchronize OFDMA transmission timing with the user application.



Fig. 8: Previous results using 2.4 GHz and TSN with 6x protected window multiplier subject to 0-80 Mbps of traffic. The vertical dashed line shows 15.7 mm application threshold for the horizontal 95th percentile. The stair-stepping is due to the overlap of best-effort traffic into the protected time-critical window.

B. Comparison to Wireless TSN

The previous results, from [2], are shown in Fig. 8. The Cartesian Error and RTT plots show close to deterministic performance, as the general shape of the Cartesian error is more consistent. The wireless TSN results show increased determinism with the smaller tails compared to the OFDMA on and OFDMA off cases. Note that the wireless TSN results used a 20 MHz bandwidth with a software-based AP, which had a lower maximum channel capacity of 104 Mbps using iPerf with 1000 B length packets. This is the reason why 80 Mbps was chosen to be the limit for the background traffic. To achieve the improved performance, a time-aware schedule was implemented. The schedule was designed to efficiently accommodate the TCP payload and ACK messages,

which protected the leader-follower traffic stream. Along with the protected traffic, the interfering (best-effort) traffic was scheduled to prevent interference with the protected traffic. There was overlap observed from the background traffic into the protected window, which was theorized to be due to the software implementation of the schedule at the medium access control (MAC) layer. This issue, along with the schedule's implementation and timing specifications, were further discussed in [2]. It was proposed that if wireless TSN were adopted at the physical layer, the overlapping issue could be greatly reduced.

V. CONCLUSION

In this paper, we presented the performance impacts of 802.11ax OFDMA frame triggering for a latency-sensitive control application. The experiments highlighted the latency issues of OFDMA with a 125 Hz TCP traffic stream for a latency-sensitive real-time control application. With OFDMA enabled, the average TCP transmission rate was transmitted below the 100 pps OFDMA trigger condition observed for the leader and follower ROS nodes. Since the application was based on TCP, the actual rate fluctuated based on increased latency conditions. When the TCP ACK was delayed, the transmission rate fell below the 100 pps activation threshold.

It was shown that moderately increasing the background interfering traffic within the same wireless network improved the latencies compared to the case without injected traffic. We, therefore, hypothesize that maintaining the background traffic rate above the OFDMA activation threshold sustains OFDMA triggering and, thus, reduces average OFDMA data frame latency and jitter as supported by our measured results. This paper highlights an opportunity for both improvement of the IEEE 802.11 standard and for the AP manufacturers to expose tunable parameters of OFDMA activation in the AP. Based on the experiments in this work, OFDMA requires improvements if it is to compete with CSMA channel access and TSN, as examined in the previous works [2] and [3]. Under the conditions examined in this paper, OFDMA was not shown to support the 8 ms round trip time required by the robotic application, whereas a software-based wireless TSN implementation had close to deterministic latency performance, allowing the robots to operate within acceptable limits.

The failure of OFDMA to exceed the performance of wireless TSN for our use case was a surprising result given that OFDMA activation had double the channel bandwidth from 20 MHz to 40 MHz as compared to the experiment in [2], despite the throughput requirement of the robot application being below 1 Mbps. From a high-level perspective, the wireless network had ample resources to service all frame transmissions better than a CSMA system using TSN. This points to a further need to refine AP implementations such that frames are scheduled for transmission commensurate with the triggering cadence. Indeed, as shown in [2], TSN greatly reduces jitter in this particular application by moderately increasing the average delay to schedule the TCP data and TCP ACK traffic deterministically. This result, therefore, merits further investigation as we discussed in Section IV-A2. Future

work will involve exploring the performance of OFDMA with UDP applications, investigating the hypotheses proposed here regarding the OFDMA latency issues, and testing the next generation of IEEE 802.11be OFDMA implementations.

DISCLAIMER

Certain commercial equipment, instruments, or materials are identified in this paper in order to specify the experimental procedure adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the materials or equipment identified are necessarily the best available for the purpose.

REFERENCES

- K. Montgomery and R. Candell, "A Characterization of OFDMA Uplink Activation for Industrial Applications," National Institute of Standards and Technology, Tech. Rep., 2024. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8506.pdf
- [2] R. Candell, K. Montgomery, M. Kashef Hany, S. Sudhakaran, and D. Cavalcanti, "Scheduling for time-critical applications utilizing tcp in software-based 802.1qbv wireless tsn," in 2023 IEEE 19th International Conference on Factory Communication Systems (WFCS), 2023, pp. 1–8.
- [3] R. Candell, K. Montgomery, M. Kashef Hany, S. Sudhakaran, J. Albrecht, and D. Cavalcanti, "Operational impacts of ieee 802.1qbv scheduling on a collaborative robotic scenario," in *IECON 2022 – 48th Annual Conference of the IEEE Industrial Electronics Society*, 2022, pp. 1–7.
- [4] G. Naik, S. Bhattarai, and J.-M. Park, "Performance analysis of uplink multi-user ofdma in ieee 802.11ax," in 2018 IEEE International Conference on Communications (ICC), 2018, pp. 1–6.
- [5] H. Yang, D.-J. Deng, and K.-C. Chen, "Performance analysis of ieee 802.11ax ul ofdma-based random access mechanism," in *GLOBECOM* 2017 - 2017 IEEE Global Communications Conference, 2017, pp. 1–6.
- [6] L. Lanante, H. O. T. Uwai, Y. Nagao, M. Kurosaki, and C. Ghosh, "Performance analysis of the 802.11ax ul ofdma random access protocol in dense networks," in 2017 IEEE International Conference on Communications (ICC), 2017, pp. 1–6.
- [7] D. Magrin, S. Avallone, S. Roy, and M. Zorzi, "Performance evaluation of 802.11ax ofdma through theoretical analysis and simulations," *IEEE Transactions on Wireless Communications*, vol. 22, no. 8, pp. 5070– 5083, 2023.
- [8] E. Avdotin, D. Bankov, E. Khorov, and A. Lyakhov, "Ofdma resource allocation for real-time applications in ieee 802.11ax networks," in 2019 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom), 2019, pp. 1–3.
- [9] Q. Zhang, P. Schulz, and G. Fettweis, "Optimizing real-time responsiveness in iiot: A dynamic approach for wifi ofdma uplink transmissions," in 2023 IEEE 98th Vehicular Technology Conference (VTC2023-Fall), 2023, pp. 1–7.
- [10] J. Li, S. K. Bose, and G. Shen, "Cooperative resource scheduling for time-sensitive services in an integrated xgs-pon and wi-fi 6 network," *IEEE Communications Letters*, vol. 26, no. 6, pp. 1338–1342, 2022.
- [11] E. Mozaffariahrar, F. Theoleyre, and M. Menth, "A survey of wi-fi 6: Technologies, advances, and challenges," *Future Internet*, vol. 14, no. 10, p. 293, Oct. 2022. [Online]. Available: http://dx.doi.org/10.3390/fi14100293
- [12] K. Montgomery, R. Candell, and M. Hany, "Uncertainty analysis of the dual-lift use case for the industrial wireless testbed," 2023-10-02 04:10:00 2023. [Online]. Available: https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id = 956530
- [13] "A tool for network performance measurement and tuning." [Online]. Available: https://iperf.fr
- [14] T. Uemura, Y. Tanigawa, and H. Tode, "Tcp-aware ofdma transmission based on traffic intensity in downlink and uplink directions in ieee 802.11ax wireless lans," in 2021 IEEE 32nd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), 2021, pp. 1024–1029.