# An Experimental Study of Triggered Multi-User Uplink Access with Real Application Traffic

Vinicius Da Silva Goncalves
*dept. of Electrical and Computer Engineering*
*Rice University*
Houston, USA
vinicius@rice.edu

Edward W. Knightly
*dept. of Electrical and Computer Engineering*
*Rice University*
Houston, USA
knightly@rice.edu

***Abstract*—The 802.11ax amendment introduced Triggered Uplink Access (TUA) to Wi-Fi to support uplink Multi-User (MU) MIMO. TUA coordinates simultaneous transmission of uplink users via an AP-transmitted trigger that gives an AP-selected group of users permission to transmit simultaneously for an AP-selected duration of time. Thus, TUA promises performance gains by enabling multi-user transmission and reducing contention overhead for access. In this paper, for the first time, we experimentally study the role of real application traffic on the performance of TUA. In particular, while TUA gains for fully backlogged traffic are well established, we show that bursty closed-loop traffic radically transforms performance. Using a real-time emulator, we experimentally evaluate the empirical limits of triggered uplink multi-user access with traffic from a real file transfer application and different uplink triggering strategies. Our results show that TUA significantly reduces file transfer latency compared to legacy single-user uplink, but unfortunately the standardized method for low-overhead backlog reporting leaves substantial benefits unrealized. Moreover, we show that unlike a single-user uplink, TUA has non-monotonic performance with respect to the frame aggregation limit.**

***Index Terms*—Medium access control, TUA, Reliable transport, Multi-user, MIMO, WLAN, IEEE 802.11ax**

## I. INTRODUCTION

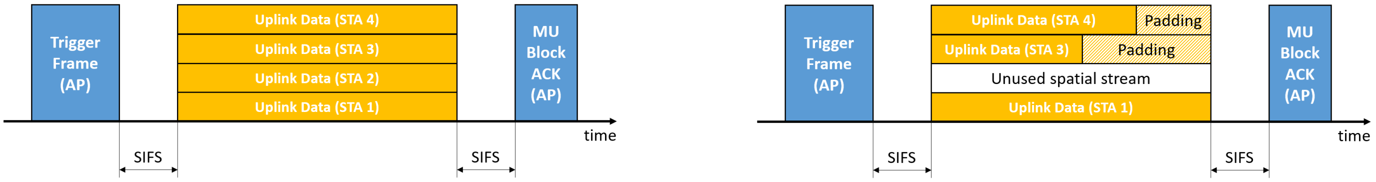
The 802.11ax amendment brings new features and modifications to the physical layer (PHY) and medium access control (MAC) sublayer of the 802.11 protocol for high efficiency operation in frequency bands between 1 GHz and 7.125 GHz. A key component for the operation of multi-user (MU) uplink in 11ax is the newly introduced Triggered Uplink Access (TUA) mechanism, which allows an AP station to trigger multiple non-AP stations to start a simultaneous transmission to the AP. TUA is the first mechanism in the 802.11 standard to allow multiple non-AP stations to simultaneously transmit independent data streams in the uplink direction. Thus, TUA not only enables spatial multiplexing, but also targets to do so efficiently, by using a single message to identify a group of stations and coordinate both their start and finish times for channel access.

Previous experimental studies have shown that near full-rank multiplexing gains are possible in uplink MU-MIMO Wi-Fi, with linear gains as the number of simultaneous streams increase [1]. However, despite such potential for physical layer gains, in this paper, we for the first time, experimentally study the key impact of real application traffic driven by closed-loop TCP congestion control dynamics on multi-user uplink WLANs, and present the following contributions.

First we define and implement two uplink strategies for TUA with MU-MIMO: *(i)* Piggy-backed Backlog status Report (PBR), which is an 11ax-based simple and practical uplink strategy in which the only mechanism for a client to report backlog status information to the AP is piggy-backing the backlog status (number of bytes currently backlogged) on the header of an uplink data transmission, whether MU or single user (SU). In this strategy, the AP will only trigger TUA transmission for stations that have a non-zero backlog report from previous uplink channel accesses. *(ii)* Real-Time Backlog status information (RTB), in which the AP always has perfect real-time knowledge about the uplink backlog status from all associated stations. While not realizable in practice, we use this strategy as an empirical upper-bound for the performance of TUA. We implement both schemes on a flexible MU-MIMO real-time WLAN emulator [2] that supports real application traffic using existing operating system implementations of TCP.

Next, we experimentally explore the performance gap between PBR and RTB while using a WLAN with a single-user uplink as a baseline. We find that for file transfers with average latency of 100 msec, MU with TUA can support significantly higher loads when compared to SU (63% and 98% for the standard PBR and the empirical bound, RTB perfect backlog knowledge). Likewise, under a fixed normalized application load of 1.18 (102.3 Mbps), PBR and RTB reduce average file transfer latency compared to SU by 74% and 82% respectively. However, despite these improvements compared to a SU uplink, the standardized PBR method leaves significant performance gains unrealized: For example, RTB attained a delay-constrained throughput up to 2.5 times greater than PBR in these experiments. The key reason is that with PBR, the AP is often not aware that a client that previously reported zero backlog has now become backlogged. The AP then fails to trigger the station for multi-user uplink transmission. This uplink slow down can subsequently cause TCP to throttle its congestion control, leading to even fewer opportunities for

This research was supported by Cisco, Intel, and by DOD: Army Research Laboratory grant W911NF-19-2-0269.



(a) Ideal TUA MU transmission

(b) TUA MU transmission with inefficiencies

Fig. 1: Example timelines of a TUA multi-user transmission: (a) ideal case in which all stations have enough backlog to fill up the total duration set by the AP, and (b) inefficiencies in the form of a lack of backlog in certain stations to fill the total duration with data and the lack of AP knowledge of backlog in stations to fully utilize spatial multiplexing.

simultaneously backlogged stations to be triggered.

Finally, we study the interplay between *frame aggregation*, TCP congestion control and TUA and *(i)* discover a non-monotonicity not present in prior studies: In contrast to a SU uplink network, we show that end-to-end file transfer latency and throughput vary non-monotonically with the frame aggregation limit for multi-user uplinks using TUA. Thus, the best choice for the frame aggregation limit must balance efficiency gains from transmission of back-to-back frames with TUA's need for sufficient feedback of non-zero backlog feedback information to help the AP trigger multi-user TUA transmissions. *(ii)* We show that the PBR reporting mechanism can be very efficient and accurate in keeping the AP backlog information updated if frame aggregation is not used. However, with the maximum level of frame aggregation limit, clients often report zero backlog after a transmission such that the AP is unaware that stations are backlogged in about two-thirds of the uplink channel access events. *(iii)* We study the channel air time distribution and show that by enabling MU uplink transmission, TUA improves channel efficiency, yielding a higher saturation throughput and improving performance overall. In addition, when combined with frame aggregation, TUA reduces contention overhead by more than 5 × . Last, we find that, combined with frame aggregation, TUA reduces average file upload latency by more than 2×. However, if frame aggregation limits are set too high, the increased usage of channel time for the uplink and the resulting reduction in channel air time available for the downlink can offset efficiency gains from frame aggregation and cause the overall latency of the network to rise (and the throughput to drop).

## II. TUA AND TRAFFIC DYNAMICS

In this section we introduce the key concepts and technologies of uplink multi-user transmissions for WLANs.

### A. *Background on TUA and Wi-Fi multi-user MIMO*

Triggered Uplink Access enables an AP station to identify and trigger multiple non-AP stations to start a simultaneous transmission to the AP by broadcasting a trigger frame. Figure 1 shows example timelines of TUA with the subsequent MU-MIMO data and ACK transmissions. The Trigger Frame, sent by the AP, contains the list of stations that can participate in the subsequent uplink channel access, information about the resource units that are allocated to each station, and the duration of the transmission, among other mandatory and optional fields. After the uplink transmission, the AP broadcasts an acknowledgment to all participating stations. In the current Wi-Fi standard, TUA is the only mechanism to initiate an uplink multi-user transmission.

The AP will only trigger stations that it infers to be backlogged. However, if a station is backlogged but the AP is not aware that it is backlogged, that station will not be triggered and the AP might forgo part of the available spatial multiplexing gains. Likewise, if the AP selects stations with backlog requiring different air times, padding will be used to align transmissions reducing efficiency. Both of these non-ideal cases are depicted in Figure 1b.

### B. *Backlog status and reporting*

We define backlog status as the number of packets and total bytes which a station has buffered in its outgoing queue at a given time. For the distributed nature of wireless networks, each station has direct access only to its own backlog status information, and exchanging this information requires extra resources. In TUA, since the AP is responsible for triggering one or more stations for uplink access, and must set the duration of the transmission, the AP must estimate the backlog status of its associated stations.

One straightforward way for the AP to acquire the backlog status of all associated stations is for stations to send their status to the AP after a poll. However, the overhead of such an operation increases with the number of associated stations and the reports could quickly become stale. An alternative reporting strategy to help reduce the cost of each report is to piggy-back a station's backlog status in previous uplink frame transmissions. The 11ax protocol defines a mechanism in this class called Unsolicited Buffer Status Report which can be used by any non-AP station, and involves the implicit report of backlog status in control fields of any uplink data (or null data) frame transmission (but not 802.11 ACK frames). It is therefore a low overhead mechanism that only adds the overhead of the Buffer Status Report field, and does not require separate contention and transmissions. Nonetheless, piggy-backed backlog reports have limitations. First, the reports can only be sent by stations that already have an active traffic flow. In other words, any station that receives traffic after some idle time will not be able to report its backlog through

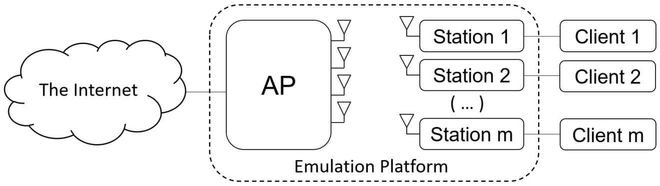


Fig. 2: System Diagram.

this mechanism. Second, the time in which reports are sent is necessarily different from the moment of a TUA transmission event. During the interval between the two events, the backlog status report may become stale.

TUA's performance will be impacted by the accuracy of the APs backlog estimates. While the often-simulated case of "fully backlogged" renders TUA perfect, bursty traffic poses a backlog estimation challenge for the AP. Thus, we study application traffic utilizing TCP congestion control.

## III. EXPERIMENTAL PLATFORM AND METHODOLOGY

In this section, we present the experimental platform we used for implementation and evaluation of TUA policies, together with the system design and all the common parameters used in the experiments.

### *A. Scenario and wireless network emulator*

We use a WLAN emulator and end-to-end testbed that supports real-time network traffic from commercial devices connected to the internet and running any application [2]. At the same time, it allows for flexible implementation and evaluation of MAC protocols and policies, including advanced features such as multi-user MIMO transmissions, user grouping, triggered uplink access, buffer status report and more. The platform achieves real-time emulation of a wireless network via a faster-than-Wi-Fi Ethernet LAN to interface with clients and the Internet coupled with an emulation engine that realizes precise controlling the time of each packet transmission according to the 802.11ax specifications.

Figure 2 shows a block diagram of the emulated WLAN. The system is composed of one multi-antenna AP associated with multiple non-AP single antenna stations. The AP is also connected to the backbone Internet via a wired link, and traffic can flow from each station to remote servers connected to the AP.

The PHY is emulated with full spatial multiplexing gains as reductions in rate due to PHY factors are studied elsewhere (see the Related Work discussion). The MAC sublayer is fully compliant with the 802.11 standard and uses enhanced distributed coordination function (EDCF) for contention and channel access. The downlink uses MU-MIMO with spatial multiplexing up to the configured maximum number of streams. If the number of backlogged stations exceeds the number of spatial streams available, the downlink randomly selects a maximal group for transmission. The uplink MAC strategy is selected from the options described in section III-B.

### *B. Uplink strategies*

We implement and analyze three uplink strategies. We start with a legacy baseline strategy in which all uplink transmissions are single-user (MU uplink is turned off). In the SU uplink network, each station must contend for channel access using the 801.11 EDCF mechanism. Upon winning contention, each station transmits an uplink data frame, which may include a number of aggregated data frames up the the limits of the protocol for a singe transmission.

Next, we define two backlog reporting mechanisms to study MU networks with TUA. First, the piggy-backed backlog information report (PBR) includes the backlog status in each uplink transmissions from each station. This way, the AP can receive the information about the amount of data left at the station for a future TUA transmission. The AP uses this information to trigger uplink transmissions via TUA, including the group selection and duration allocated for each access. However, between the time of the report and the triggered uplink access each station may receive more backlogged frames which would not be reported to the AP.

Second, we define the real-time backlog information (RTB) strategy as an ideal (but not implementable in practice) reporting strategy in which the AP has perfect knowledge about the backlog of all associated stations at all times. In other words, the stations are able to report their backlog information in real time. The practical implementation of such strategy is not in the scope of this work. We use the RTB strategy as an upper-bound on the TUA mechanism, to isolate the effects of backlog reporting on the end-to-end performance of the system with TUA.

It is important to note that in both PBR and RTB TUA strategies, some SU transmissions will still occur. Namely, even in a MU network, stations always set a contention countdown timer when they become backlogged. If a station's timer expires before a TUA trigger is received, it will transmit SU. In contrast, if the TUA trigger is received before the timer expires, it will transmit MU.

### *C. Traffic*

All traffic is generated by applications running full network stacks on commodity devices. The traffic generator uses the built-in implementation of TCP from the device's operating system, and all the reliability mechanisms and congestion control comes from the default implementation of the protocol.

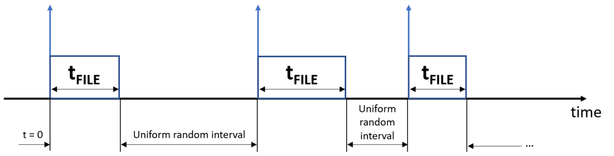


Fig. 3: Example timeline of the traffic application.

We implemented a file transfer application in which each station uploads or downloads a sequence of files initiated at times that are statistically independently among stations. Figure 3 shows the an exemplary high-level timeline of the

TABLE I: Base parameters for Experiments

| | |
|---|---|
| SISO PHY rate (MCS 8) | 86.7 Mbps |
| AP CW$_{min}$ | 12 |
| Stations CW$_{min}$ | 48 |
| File size | 300 kB |
| Associated stations | 32 |

traffic generated by this application. Each host downloads or uploads a series of files (randomly selecting the direction of transmission between download or upload) with a predefined file size. The interval between the end of a file transmission and the start of the following transmission is drawn from a uniform random distribution with fixed mean. By default, the file is an upload or download with equal probability. The file size and interval between files define the total load in the network.

### *D. Measuring performance with traffic from real applications*

We select file transfer latency as our primary metric for performance. The congestion control mechanism in TCP enforces a limit in the traffic generation rate, which is affected by the performance of the end-to-end link, and, in turn can slow down the transmission of each segment of TCP transaction. By measuring the time to transfer each TCP file, we can evaluate the overall performance of the wireless link and the efficiency of the TUA mechanism for the proposed scenario.

We further report the network's aggregate application-layer saturation throughput. This is the application-layer throughput attained when the uniform time duration between file transfers is set to zero.

Unless otherwise noted, all experiments in this paper use the configuration parameters presented by Table I.

## IV. THE PERFORMANCE GAP BETWEEN PBR AND RTB

**Research Question.** A key to achieving the multiplexing gains of TUA is that the AP must trigger stations with enough backlog to transmit during most of the access time. With PBR, the 11ax standard solution for low-overhead piggy-backed backlog reports, the AP must contend with delayed and missing backlog information. In this first set of experiments, we study the performance gap between TUA with PBR and RTB, the real-time backlog scheme that provides an experimental upper-bound on performance via full backlog knowledge. Likewise, we study SU uplink network as a baseline.

**Experimental setup.** The experimental setup consists of a single AP equipped with 8 antennas and 32 single-antenna user stations. Application traffic is generated independently in each of the user stations and a server, physically co-located with the AP, serves a mix of downloads and uploads according to the application described in section III-C. Each file has a random and equal probability of being a download or an upload, making it a 50/50 mix of downloads and uploads in each station. The file size is fixed to 300 kB and the total load is controlled by changing the average interval between consecutive file transfers. The total load is equally distributed among stations.

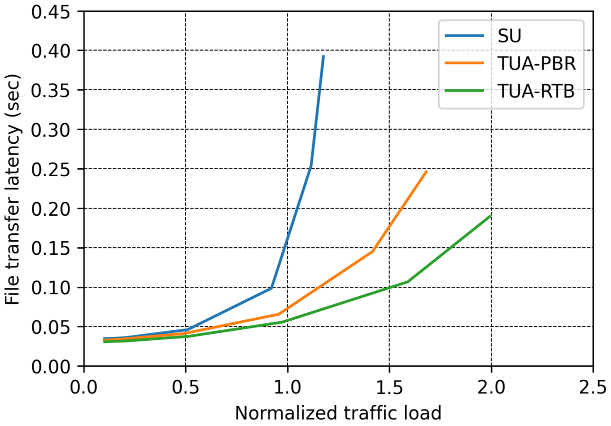


Fig. 4: Average latency to transmit each file versus the total traffic in the network.

**Results.** Figure 4 shows average upload and download file transfer latency as a function of traffic load for SU uplink network and the two TUA policies. Load is normalized by the PHY layer SISO rate of 86.7 Mbps to factor out the fixed per-stream MCS used and to provide the context of how much of the PHY data rate is being converted into transport layer throughput. Because we emulate perfect spatial multiplexing, an 8x8 MIMO channel has a maximum normalized PHY data rate of 8.

First, SU network (blue) not only has higher delay than MU for all loads, but also has a latency profile that increases with load significantly more sharply with load than MU. SU reaches an average file transfer latency of 0.39 sec for a normalized traffic load of 1.18. This point corresponds to the scenario where the time interval between file transmissions is zero. In other words, each station has an active file at all times, and the throughput is the maximum for the application, uplink strategy, and WLAN capabilities.

In contrast, the 11ax MU PBR strategy (orange), realizes a significantly reduced latency under increasing traffic load. For example, at the same traffic load 1.18, PBR's latency is 0.10 sec, a reduction of 74% compared to SU. Moreover, PBR is also able to support a significantly increased maximum traffic load. When the time interval between file transmissions is set to zero in each station, the maximum throughput with PBR is 1.68, an increase of 42% compared to SU.

Third, the RTB strategy (green) has the lowest average file transfer latency for all traffic loads indicating that real-time knowledge of stations' backlog at the AP has a significant performance benefit. For example, at the normalized traffic load of 1.18, RTB's latency is 0.07 sec, a reduction of 82% compared to SU network and 30% compared to PBR. At the application's limit with zero interval between file transfers, the maximum throughput with RTB increases to 1.99, an increase of 69% compared to SU network and 18% compared to PBR. Thus, at each load, perfect real-time backlog knowledge enables the AP to more efficiently trigger stations as compared to piggy-backed reports, thereby reducing the application's average file transfer latency. While both TUA mechanisms significantly outperform SU, compared to RTB, PBR fails to realize a large portion of the available throughput and latency improvements.

Moreover, in parallel with the potential latency reduction for a fixed application load of TUA, a dual result from this experiment is that for a fixed latency target, MU TUA can support a higher traffic load than SU network. For example, consider an application with an average file transfer latency constraint of 100 ms. Interpolating the results, the maximum load that could be supported by each of the three uplink strategies under this constraint is 0.922, 1.155, and 1.505, for SU network, PBR, and RTB, respectively. That is, when we compare the SU uplink network mode with the two MU modes, we find that the TUA strategies help the network carry substantially more load while keeping the average latency under the defined requirement. Likewise, having perfect real-time backlog knowledge available at the AP via PBR makes the maximum empirical load increase 63% when compared to the SU, which is more than 2.5 times of the increase with MU-reports alone. Thus, we can conclude that a WLAN with TUA can potentially serve more client stations using the same bandwidth and achieving the same average file transfer latency performance than that of a SU uplink WLAN.

Finally, observe that in the low-traffic load regime (loads less than 0.5 or 43.8 Mbps), MU-PBR and MU-RTB perform similarly to SU network. In this case, there is not enough traffic to have multiple stations simultaneously backlogged, and thus, even RTB can only rarely trigger simultaneous multi-user transmissions.Thus, TUA's benefits will be most pronounced in WLANs with medium to high station density or medium to heavy traffic.

**Findings.** For file transfers with average latency of 100 msec, MU with TUA can support significantly higher loads when compared to SU (63% and 98% for the standard PBR and the empirical bound, RTB perfect backlog knowledge). Likewise, under a fixed application load of 1.18 (102.3 Mbps), PBR and RTB reduce average file transfer latency by 74% and 82% respectively. Unfortunately, the standardized method of low-overhead backlog reporting leaves significant performance gains unrealized. For example, RTB attained a delay-constrained throughput up to 2.5 times greater than PBR in these experiments. In such cases, the AP is often not aware of the high backlog at multiple stations, and therefore fails to trigger TUA transmissions and forgoes uplink multiplexing gains. The slow down on the uplink subsequently causes TCP to throttle its congestion control, leading to even fewer opportunities for simultaneously backlogged stations to be triggered.

## V. FRAME AGGREGATION

Frame aggregation refers to sending two or more data frames in one channel access. It has the goal of increasing efficiency by reducing contention time per frame transmitted. In our previous experiment we used the maximum value for frame aggregation limit allowed by the standard. But because frame aggregation impacts both backlog behavior and TCP, in this section we evaluate its impact on TUA.

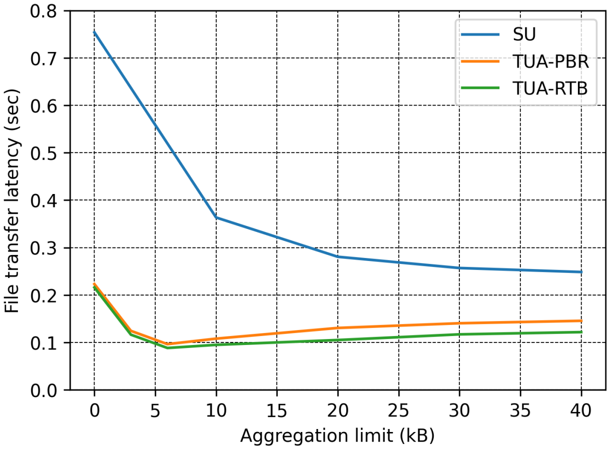


Fig. 5: Average latency to transmit files in all 3 uplink modes versus the aggregation limit in the network.

### A. Latency and throughput

**Research Question.** In PBR, the standardized method of piggy-backed backlog reporting, backlog information is fed back to the AP during uplink accesses and transmissions. With a greater frame aggregation limit, the reported backlog will be smaller as more bytes will be transmitted. In the limit, if frame aggregation is sufficiently high to empty the queue, the backlog will be correctly reported to be zero, and the AP does not subsequently trigger stations with reports of zero.

**Experimental setup.** To explore the impact of frame aggregation limit on TUA, we vary the maximum number of bytes (and hence frames) that can be aggregated, which we term $\mathrm{FA}_{\mathrm{max}}$. In particular, we begin with $\mathrm{FA}_{\mathrm{max}} = 1$ frame, which corresponds to no frame aggregation, i.e., each client can transmit only a single frame per access. In subsequent experiments, we increase $\mathrm{FA}_{\mathrm{max}}$ up to 40 kB, which corresponds to the maximum transmission time allowed by the standard at the data rate used in our experiments. Similar to the prior experiments, we repeat the setup of one AP with 8 antennas and 32 single-antenna stations with SU uplink, PBR, and RTB. Traffic is generated by the same TCP file transfer application as before, and we run the file transfer application at the maximum load, in which each file transfer starts immediately after the previous completes, without any idle interval.

**Results.** Figure 5 depicts file transfer latency as a function of the maximum frame aggregation parameter $\mathrm{FA}_{\mathrm{max}}$ for SU uplink network (blue) and MU-MIMO TUA with PBR (orange) and RTB uplink (green). First, observe that for the SU uplink network, increased frame aggregation monotonically reduces end-to-end latency, albeit with marginally increasing benefits. This result is in agreement with prior work that studied both open-loop [3] and TCP traffic [4] on a SU uplink and found that delay decreases monotonically with increasing

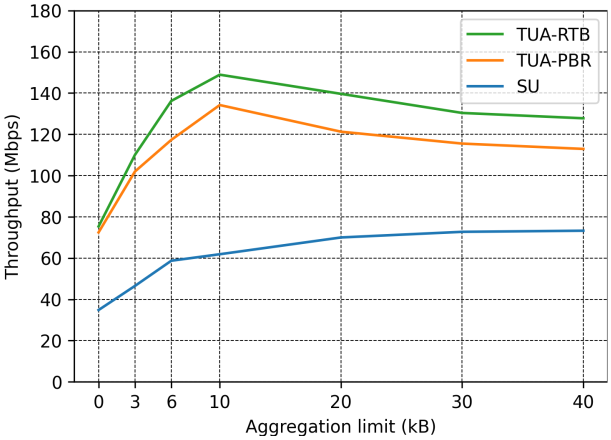


Fig. 6: Maximum aggregate throughput for the TCP end-to-end application with the 3 uplink strategies.

frame aggregation limit due to frame aggregation's increasing efficiency benefits.

In contrast, the same figure shows that when the TUA mechanism is active, latency does not decrease monotonically with aggregation limit. Strikingly, file transfer latency is not minimized with the maximum aggregation limit of 40 kB, nor is it minimized when frame aggregation is turned off. Instead a moderate aggregation limit of 6 kB minimizes latency. Thus, both non-use of frame aggregation and excessive frame aggregation reduce performance as these extremes hinder interacting control mechanisms (TCP congestion control and spatial multiplexing). In particular, while maximizing frame aggregation limits in principle maximizes layer 2 efficiency, the resulting reported backlog is often zero. In this case, the AP will not consider the station for a subsequent trigger until a non-zero report arrives (e.g., via a single-user uplink access). In the other extreme of no frame aggregation, there would nearly always be a non-zero backlog report ensuring trigger eligibility. Yet, this would be inefficient at layer 2 yielding more contention per frame. We further explore tradeoffs between these extremes in subsequent experiments.

Figure 6 shows maximum aggregate throughput as a function of $\mathrm{FA}_{\max}$. That is, the figure depicts application-layer throughput for the case that there is no pause between file transfers. Here, the SU uplink network strategy again exhibits monotonic behavior, with throughput increasing with aggregation limit, albeit with diminishing returns. In contrast to SU network, as was the case with latency, TUA yields non-monotonic behavior, with peak throughput at aggregation limit of 10 kB.

Despite the non-monotonicity, observe that when we compare either of the TUA MU-MIMO uplink strategies with SU uplink, for the same traffic level and same maximum aggregation limit value, TUA always delivers a significantly higher aggregate throughput (and lower latency), even with high sub-optimal frame aggregation limits. For instance, at zero aggregation the average file transfer latency is 0.22 and 0.75 for TUA and SU, respectively, and the aggregate throughput is at 75.33, 72.45, and 34.80 Mbps for RTB and PBR TUA and SU uplink network strategies, respectively. This difference can be explained because in the SU uplink network, the uplink queues are drained slowly across multiple channel accesses via SU uplink transmissions only. In the same scenarios, TUA is able to take advantage of MU uplink transmissions to drain the parallel backlog much faster. Therefore, even though a small aggregation limit value might not be optimal for TUA, it increases the probability of multiple stations having backlog at the same time and gives opportunity for the backlog information to arrive at the AP in time for triggering more TUA transmissions.

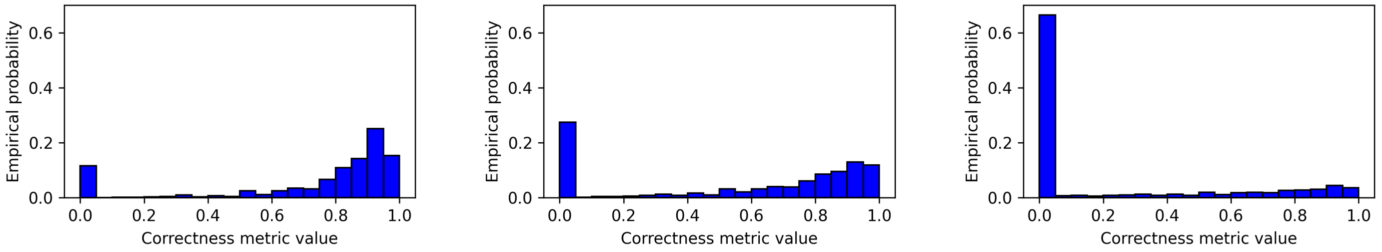
**Findings.** In contrast to a SU uplink network, end-to-end file transfer latency and throughput vary non-monotonically with the frame aggregation limit for multi-user uplinks using TUA. The best choice for the frame aggregation limit must balance efficiency gains from frame aggregation while still provide sufficient feedback of non-zero backlog information to help the AP trigger multi-user TUA transmissions. Nonetheless, even if the best frame aggregation limit is not chosen, the end-to-end TCP latency of a multi-user network with TUA is significantly smaller than for a SU uplink network because TUA either exploits uplink spatial multiplexing or falls back to SU uplink transmission if only a single station is inferred to be backlogged. The corollary holds for throughput.

### B. *Backlog report distribution*

**Research Question.** The previous experiments show that PBR leaves performance gains unrealized, and that the difference can be substantial. While the only difference between PBR and RTB is the backlog reporting mechanism, the subsequent differences between the timing and content of the reported information impact the AP's decisions for triggering uplink transmissions. Thus, here, we experimentally study the origins of the performance

**Experimental setup.** The backlog status, and hence the backlog status information, is a discrete variable over continuous time. However, for the purpose of triggering PBR transmissions, the value - and correctness - of this variable is most relevant at the time of an uplink transmissions, when the AP may or may not have the opportunity to use the more efficient multi-user access based on the backlog information it maintains. Thus, we define a correctness metric $\gamma_i(t)$ to quantify the efficiency of the backlog status reporting mechanism for PBR. We define $\gamma$ only for times of uplink channel access events, whether the transmission is single- or multi-user. Note that even in a multi-user uplink, single-user transmissions will still occur whenever a backlogged station's count-down timer expires before it receives a multi-user trigger.

Denote $B_i(t)$ as the true backlog of station $i$ at time $t$ and $\hat{B}_i(t)$ as the AP's estimate of station $i$'s backlog at time $t$. We define the correctness metric at $t$ for each of the station(s) $i$ that transmits at $t$ as $\gamma_i(t) = \hat{B}_i(t)/B_i(t)$. According to the definition, the AP's estimated backlog $\hat{B}_i(t)$ is never 0 at the time of a MU uplink transmission as the AP does not trigger stations that it infers are not backlogged. Thus,



(a) No frame aggregation (b) Aggregation limit of 6 kB (c) Aggregation limit of 40 kB

Fig. 7: Empirical distribution of the correctness metric with 3 different frame aggregation limits.

$\hat{B}_i(t) = 0$ and $\gamma_i(t) = 0$ represents a single-user transmission event. Moreover, in both single- and multi-user transmissions, $B_i(t) \neq 0$ as only (truly) backlogged stations can transmit.

Observe that for the PBR policy, $\hat{B}_i(t)$ is always less than or equal to $B_i(t)$ as the client's L2 backlog only grows over time, assuming no packets are discarded from the uplink queue. Thus, at a PBR transmission, $0 < \gamma_i(t) \leq 1$ and $\gamma_i(t)$ closer to 0 represents more severe under-reporting of backlog.

We do not report $\gamma$ for SU *networks* as $gamma$ is always zero in this case. Likewise, we do not report $\gamma$ for RTB as $\gamma$ is always by definition 1 in this case. Thus, all results are for PBR.

**Results.** Figure 7 shows histograms of $\gamma$ over all transmission epochs and all transmitting stations in three frame aggregation scenarios: no aggregation and frame aggregation limits of 6kB and 40kB. The depicted values of $\gamma$ are partitioned in 20 equal bins from 0 to 1, and the histogram values are normalized to a sum of 1 as a probability density.

We first observe from all three plots that a sizable portion of the distribution lies in the first bin for all aggregation limits. That happens because on a considerable amount of uplink transmissions, the backlog information at the AP is 0, which corresponds to the scenario where the previous uplink transmission from the station in question left the buffer empty. Moreover, as the aggregation limit increases, the occurance of zero estimates of client backlog also increases rapidly. This quantifies the manner in which frame aggregation can reduce the efficiency of the PBR reporting mechanism for TUA.

Second, the remaining distribution provides insights into the origins of PBR performance due to reporting. Excluding the first bin, between 0 and 0.05, most of the probability mass is concentrated on higher values, peaking at the second-to-last bin, between 0.9 and 0.95. This corresponds to the AP's backlog estimate being only slightly under-valued. However, this effect also varies with the aggregation limit. As the aggregation limit increases, this mass becomes much more evenly distributed among the entire range of correctness values. This indicates that when high frame aggregation limits are used, even when the AP knowledge of a station's backlog is non-zero, it has a smaller probability of being close to the correct buffer status. This hinders efficiency as the AP will only grant channel access time corresponding to its estimate of the backlog.

Finally, the average value of the correctness metric $\gamma$ can be used as an overall score for the reporting mechanism in TUA, since $\gamma = 1$ corresponds to perfect knowledge. In the three plots presented, with aggregation limit at 0, 6kB and 40kB, the average value for the correctness metric in our experiment was 0.81, 0.61, and 0.18, respectively. Thus, the 0 aggregation case points to the pitfall of having the most accurate backlog estimates, yet performing relatively poorly due to non-use of frame aggregation.

**Findings.** The PBR reporting mechanism can be very efficient and accurate in keeping the AP information updated if frame aggregation is not used, with an average correctness metric of 0.81, close to the real-time buffer status. However, as the frame aggregation limit increases, the efficiency of the reporting mechanism drops rapidly. For example, when the aggregation limit is 40 kB, the AP is unaware that stations are backlogged in about two-thirds of the uplink channel access events.

### C. Channel-time and Uplink/Downlink balance

**Research Question.** Because Wi-Fi half duplex, uplink and downlink transmissions share the same time-frequency resources. Consequently, efficiency improvements on the uplink due to TUA can provide additional resources for both the uplink and downlink. For closed-loop traffic, such as the TCP file-transfer application we use in our experiments, the impact can be complex, as congestion-controlled and reliable transport requires that data transmissions in one direction be met with transport layer acknowledgements in the opposite direction.

**Experimental setup.** To characterize the channel resource distribution, we measure how much time is used to transmit traffic in each direction, downlink and uplink. Additionally, we measure the time overhead from EDCA contention events and collided transmissions to present a complete view of average channel time usage. We use the same setup from previous experiments with an 8 antenna AP and 32 single-antenna station and we configure the AP to employ RTB TUA. As previously, we set the application parameters such that the TCP file transfer application produces an equal amount of files (offered load) for download and upload.

**Results.** Figure 8 shows the channel time usage vs. frame aggregation limit. Air time is partitioned into 5 categories, including transmission, contention and collision activity. Since

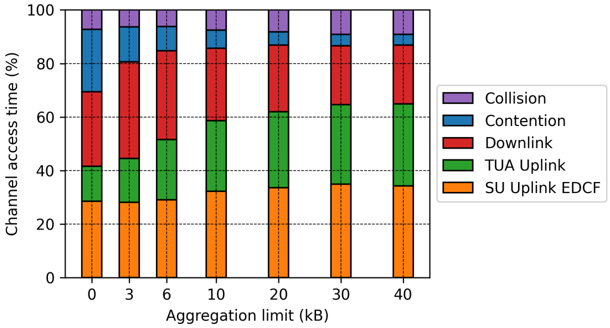


Fig. 8: Channel time usage for the RTB uplink strategy versus the frame aggregation limit in the WLAN.

the traffic in this experiment is set to produce the maximum throughput with the TCP application on each station (no time between file transfers), the medium is never idle without contention happening.

First, consider the partition of time from the left-most bar, without frame aggregation. Here, SU uplink transmission (the lower orange portion) utilizes 28% of air time. Recall that even with the TUA feature enabled, not all uplink transmissions can be grouped into multi-user transmissions. The reason is that all backlogged stations still contend for single-user channel access, which results in a SU uplink transmission via EDCF if the station is not triggered by TUA before its countdown timer reaches zero. Moreover, in the same partition, uplink TUA (the green portion) utilizes 13% of air time, amounting to a total of 42% air time for uplink transmissions.

As the aggregation limit increases, the channel-time dedicated for uplink access also increases. For example, the channel-time portion used for uplink increases to 65% from 41% when the aggregation limit increases to 40 kB from no aggregation (1 frame). Moreover, this increase arises mostly from the TUA portion of the uplink access, which increases from 13% to 31% in the same interval, while the SU transmission increases from 29% to 34%. This is because the increase in aggregation limit benefits the operation of TUA, which is able to trigger simultaneous stations for longer channel accesses. In particular, the increase in TUA channel time is steeper between the point of no aggregation and the aggregation limit of 10 kB, where it doubles from 13% to 26% of air time. The reduced gains for TUA channel time beyond aggregation limit of 10 kB arise from the balance between uplink and downlink transmissions, which share the same channel time resources to transmit near equal amounts of payload in this experiment.

Next, consider the partition of channel time used for downlink transmissions (the red partition), which are comprised of TCP data from file downloads and TCP ACKs from file uploads. The left-most bar shows a utilization of 28% of channel time, significantly smaller than the channel time used for uplink, since downlink can use MU-MIMO in all transmissions in which traffic is available. Thus, MU downlink transmission ensures that the same amount of traffic is transmitted using less air time. As the aggregation limit increases to 3kB, the downlink channel time increases to a peak of 36%, and then decreases to a minimum of 22% as the aggregation limit reaches the value of 40 kB. This non-monotonic behavior comes from the efficiency provided by frame aggregation and the contention between downlink and uplink for channel time and access. The initial gains from frame aggregation provide efficiency in channel access, with more traffic flowing in the network, and more channel time used for downlink transmissions. However, with further increased frame aggregation limit, the duration of transmissions also increases, reducing the frequency of contention events. Because the uplink is less efficient with the combined use of SU uplink transmission, the channel usage for uplink increases faster to serve the extra traffic and the downlink loses part of its channel-time for uplink. In particular, because TUA can only be triggered when the AP wins contention for channel access, with more frequent opportunities for uplink TUA transmissions a larger portion of the AP channel access goes for that mechanism, leaving less time for downlink transmissions.

Finally, the two upper partitions represent the overhead from contention (blue) and collision (purple) in the channel time. Without frame aggregation, on the left-most bar, contention utilizes 23% of channel time and collisions take over the remaining 7%. The two combined take 30% of the channel time resource. A higher aggregation limit reduces the channel time used for contention from 23% to only 4%, as the limit goes from 0 to 40 kB. This change comes from the efficiency brought by frame aggregation, where a single channel access can transmit more payload, reducing the number of contention events needed to transmit the total payload in the network. Moreover, this reduction is more pronounced for aggregation limits between 0 and 10 kB, and declines as the aggregation limit increases further. At the same time, the channel-time consumed by colliding transmissions is affected by the frame aggregation limit in the opposite direction. The channel time overhead from collisions increase from 7% to 9% when the aggregation limit goes from 0 to 40 kB. With the increase in frame aggregation, transmissions can be longer, which also makes the time lost in each collision grow proportionally. Even though the rate of channel access events in the network decreases, decreasing the number of collision events, the time of each collided transmission increases faster, leading to a net increase in the channel-time overhead from collisions.

**Findings.** By enabling multi-user uplink transmission, TUA improves channel efficiency, yielding more air time for other purposes and improving performance overall. In addition, when combined with frame aggregation, TUA yields a reduction of contention overhead from 23% to 4% of channel time, a decrease of more than $5\times$. However, with higher efficiency from aggregation and thus more traffic, the uploads increase the usage of channel time. The highly utilized uplink then leaves less space for downlink transmissions, which leads to a loss of performance for downloads when aggregation is large.

### D. *Download vs. Upload performance*

**Research Question.** By definition of the application, the

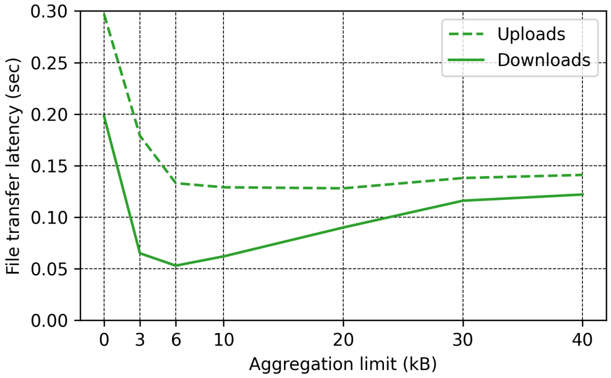


Fig. 9: Average end-to-end latency per file transmission across all stations in the network with the perfect real-time uplink backlog knowledge at the AP.

number of files downloaded and uploaded in the network is near the same. But the operation of downlink and uplink is distinct, which reflects in a difference on the end-to-end performance of downloads and uploads.

**Experimental setup.** In order to contrast the performance of files transmitted in each direction, we decompose latency into file downloads and uploads (which respectively use downlink for TCP data and uplink for TCP ACKs and vice versa), to investigate the separate performances as the aggregation limit changes. The TCP file transfer application is configured to produce an equal amount of files for download and upload, and we use the same setup from previous experiments with an 8 antenna AP and 32 single-antenna stations, with the AP configured to employ RTB TUA.

**Results.** Figure 9 shows the average end-to-end latency of all uploads and all downloads in the network. Note that, by design of the application and experiment, the throughput is equally distributed between download and upload files. The difference in latency comes from the asymmetry in downlink and uplink operations.

First, consider average latency for uploads (dashed line) with varying frame aggregation limit. The latency with no aggregation (left-most point) is 0.297 sec, and decreases sharply with the introduction of frame aggregation up to a threshold. The latency for uploads reaches a value of 0.133 sec for an aggregation limit of 6 kB, representing a reduction of more than 2x in average latency. Beyond this point, the curve remains relatively constant with minimum of 0.129 sec (at 20 kB) and maximum of 0.141 sec (at 40 kB).

Second, for downloads (solid line), latency presents a strong non-monotonic behavior with varying frame aggregation limit. Without frame aggregation, the average latency for downloads is 0.198 sec. It then reduces to a minimum of 0.053 sec with an aggregation limit of 6 kB, with an inflection point similar to the uploads. However, as the aggregation limit increases above 6 kB, the latency increase again, reaching a value of 0.141 sec for the maximum aggregation limit of 40 kB.

We know from our previous experiments that average latency over all files exhibits non-monotonic behavior, with latency minimized for a moderate aggregation limit of 6 kB. But here we find that this behavior is not completely explained by the latency of uploads, as one might have expected. Rather, the non-monotonic behavior arises primarily from downloads, with AP having a reduced probability to win contention for downlink transmissions because of longer transmission times and more opportunities for uplink channel access being created by the usage of TUA in the uplink, when the aggregation limit is large.

Finally, note that the average latency for uploads is still higher across all values of frame aggregation limit, even with the non-monotonic behavior of downloads.

**Findings.** Combined with frame aggregation, TUA yields a reduction of more than 2× in average latency for file uploads. However, for large values of aggregation limit, the loss in downlink channel time can offset the efficiency gains from the additional frame aggregation and make the overall latency of the network to rise (and throughput of the system to drop). This effect arises primarily from downloads.

## VI. RELATED WORK

**Multi-user Uplink for 802.11 WLANs.** Prior work proposed schemes for uplink MU-MIMO in the context of 802.11 networks to achieve full-rank uplink capacity [1]. However, most of this work studied PHY layer parameters, such as modulation rate adaptation [5], user selection based on CSI orthogonality [6], and sequential decoding of concurrent frames [7], and all were evaluated with the assumption of fully-backlogged traffic. The most recent 802.11ax amendment introduces uplink MU-MIMO to the standard in the form of TUA transmissions, among other enhancements. This amendment has been studied in several work since its announcements, including comprehensive tutorials and surveys [8, 9].

**Channel access strategies for 802.11 MU-MIMO.** Prior work extensively investigated, analysed and proposed alternate access mechanism for 802.11 WLANs, including antenna selection [10], heterogeneous MIMO [11], learning-based optimizations [12], among others [13–15]. More recent work also investigated 11ax specific medium access problems such as scheduling duration for uplink multi-user transmissions [16], backoff control [17], and OFDMA resource allocation for delay minimization [15].

**Traffic, MU-MIMO and Performance analysis.** Prior work investigated the performance of TCP and closed-loop traffic performance over Wi-Fi [18–21]. More recent work also investigated the performance of closed-loop traffic over downlink multi-user MIMO, with mathematical models and simulation results showing that the gains of downlink MU-MIMO in WLANs can be greatly reduced for long-lived TCP flows when they are served by a single-user (SU) uplink, due to the starvation of TCP ACKs [22, 23].

In contrast to all of the aforementioned work, this paper is the first to study TUA with closed-loop traffic.

## VII. CONCLUSIONS

We presented the first experimental evaluation of the IEEE 802.11 Triggered Uplink Access mechanism with closed-loop traffic. We defined two classes of backlog reporting mechanism, namely, Piggy-Backed Reports and Real-Time Backlog, and we perform extensive experimental analysis on the latency and throughput performance of both strategies in comparison with the legacy SU uplink network. Our results show that TUA can deliver significantly reduced latency compared to SU uplink, yet PBR leaves substantial gains unrealized. Moreover, we demonstrate a non-monotonic relationship between frame aggregation limits and performance of end-to-end latency of TCP file transmission, which is only seen when the TUA uplink is used.

## REFERENCES

[1] A. B. Flores, S. Quadri, and E. W. Knightly. “A Scalable Multi-User Uplink for Wi-Fi”. In: *13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16)*. 2016.

[2] V. Da Silva Goncalves. *”PERFORM: A Platform for Experimental Research in WLAN with Focus on Real Network Traffic and Multi-User Channel Access”*. Master’s Thesis, Rice University. 2020.

[3] T. Selvam and S. Srikanth. “A frame aggregation scheduler for IEEE 802.11n”. In: *2010 National Conference On Communications (NCC)*. 2010.

[4] M. Samih and E. Al-Hemiary. “Impact of TCP congestion control algorithms on IEEE802.11n MAC frame aggregation”. In: *Inter. Journal Computer Science and Engineering Technology (IJCSET)* 2.9 (2012).

[5] W.-L. Shen, K. C.-J. Lin, S. Gollakota, and M.-S. Chen. “Rate adaptation for 802.11 multiuser MIMO networks”. In: *IEEE Transactions on Mobile Computing* 13.1 (2013), pp. 35–47.

[6] A. Zhou, T. Wei, X. Zhang, M. Liu, and Z. Li. “Signpost: Scalable MU-MIMO signaling with zero CSI feedback”. In: *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing*. 2015.

[7] K. Tan, H. Liu, J. Fang, W. Wang, J. Zhang, M. Chen, and G. M. Voelker. “SAM: enabling practical spatial multiple access in wireless LAN”. In: *Proceedings of the 15th annual international conference on Mobile computing and networking*. 2009.

[8] B. Bellalta, L. Bononi, R. Bruno, and A. Kassler. “Next generation IEEE 802.11 Wireless Local Area Networks: Current status, future directions and open challenges”. In: *Computer Communications* 75 (2016).

[9] E. Khorov, A. Kiryanov, A. Lyakhov, and G. Bianchi. “A tutorial on IEEE 802.11 ax high efficiency WLANs”. In: *IEEE Communications Surveys & Tutorials* 21.1 (2018), pp. 197–216.

[10] Y. Kim, S. Cho, and D. K. Kim. “Low complexity antenna selection based MIMO scheduling algorithms for uplink multiuser MIMO/FDD system”. In: *2007 IEEE 65th Vehicular Technology Conference-VTC2007-Spring*. 2007.

[11] K. C.-J. Lin, S. Gollakota, and D. Katabi. “Random access heterogeneous MIMO networks”. In: *ACM SIGCOMM Computer Communication Review* 41.4 (2011).

[12] R. Ali, N. Shahin, Y. B. Zikria, B.-S. Kim, and S. W. Kim. “Deep reinforcement learning paradigm for performance optimization of channel observation–based MAC protocols in dense WLANs”. In: *IEEE Access* 7 (2018), pp. 3500–3511.

[13] N. Anand, J. Lee, S.-J. Lee, and E. W. Knightly. “Mode and user selection for multi-user MIMO WLANs without CSI”. In: *2015 IEEE Conference on Computer Communications (INFOCOM)*. 2015.

[14] W. Ahn, Y. Y. Kim, and R. Y. Kim. “An energy efficient multiuser uplink transmission scheme in the next generation WLAN for Internet of Things”. In: *International Journal of Distributed Sensor Networks* 12.7 (2016).

[15] D. Bankov, A. Didenko, E. Khorov, V. Loginov, and A. Lyakhov. “IEEE 802.11 ax uplink scheduler to minimize, delay: A classic problem with new constraints”. In: *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications*. 2017.

[16] M. Karaca, S. Bastani, B. E. Priyanto, M. Safavi, and B. Landfeldt. “Resource management for OFDMA based next generation 802.11 WLANs”. In: *2016 9th IFIP Wireless and Mobile Networking Conference (WMNC)*. 2016.

[17] Y. Kim, L. Kwon, and E.-C. Park. “OFDMA Backoff Control Scheme for Improving Channel Efficiency in the Dynamic Network Environment of IEEE 802.11ax WLANs”. In: *Sensors* 21.15 (2021).

[18] H. Wu, Y. Peng, K. Long, S. Cheng, and J. Ma. “Performance of reliable transport protocol over IEEE 802.11 wireless LAN: analysis and enhancement”. In: *Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies*. Vol. 2. 2002.

[19] D. Miorandi, A. A. Kherani, and E. Altman. “A queueing model for HTTP traffic over IEEE 802.11 WLANs”. In: *Computer networks* 50.1 (2006), pp. 63–79.

[20] J. Yu and S. Choi. “Modeling and analysis of TCP dynamics over IEEE 802.11 WLAN”. In: *2007 Fourth Annual Conference on Wireless on Demand Network Systems and Services*. 2007.

[21] S. R. Pokhrel, M. Panda, H. L. Vu, and M. Mandjes. “TCP performance over Wi-Fi: Joint impact of buffer and channel losses”. In: *IEEE Transactions on Mobile Computing* 15.5 (2015), pp. 1279–1291.

[22] P. Nayak, M. Garetto, and E. W. Knightly. “Multi-user downlink with single-user uplink can starve TCP”. In: *2017 IEEE Conference on Computer Communications (INFOCOM)*. 2017.

[23] P. Nayak, M. Garetto, and E. W. Knightly. “Modeling Multi-User WLANs Under Closed-Loop Traffic”. In: *IEEE/ACM Transactions on Networking* 27.2 (2019), pp. 763–776.