Evaluating UORA-Based Polling Mechanism for Latency-Sensitive Uplink Traffic in Wi-Fi Networks

Douglas Dziedzorm Agbeve, Andrey Belogaev, Chris Blondia, Jeroen Famaey
IDLab, University of Antwerp – imec, Belgium
{douglas.agbeve, andrei.belogaev, chris.blondia, jeroen.famaey}@uantwerpen.be

***Abstract*—IEEE 802.11ax (Wi-Fi 6) introduced Orthogonal Frequency Division Multiple Access (OFDMA), which enables simultaneous transmissions through centralized resource allocation. However, effective uplink scheduling requires the Access Point (AP) to identify which stations (STAs) have data to transmit. This typically necessitates polling for buffer status reports, a process that becomes increasingly inefficient and unscalable with growing device density. In this paper, we study how the Uplink OFDMA-based Random Access (UORA) feature improves the scalability and delay experienced by latency-sensitive data streams. We show that UORA enables efficient uplink scheduling while opportunistically identifying buffered traffic from unscheduled STAs, striking a balance between coordination and scalability. Performance evaluation of different polling strategies is done by means of simulation in ns-3. The results indicate that UORA-based polling outperforms alternative schemes in densely deployed network environments with heterogeneous uplink traffic patterns. Furthermore, under highly sparse and sporadic traffic conditions, UORA-based polling yields over 40 % delay reduction compared to Scheduled Access (SA) OFDMA.**

***Index Terms*—Wi-Fi, OFDMA, UORA, polling, scalability, simulation**

I. Introduction

Enhanced Distributed Channel Access (EDCA) has traditionally served as the primary channel access mechanism in Wi-Fi networks. This decentralized, contention-based approach allows multiple stations (STAs) to independently compete for the wireless medium. A STA that wins the contention gains access to the entire channel bandwidth for its transmission. However, when multiple STAs attempt to transmit simultaneously, collisions occur, resulting in packet loss and necessitating retransmissions through the same contention process. EDCA offers flexibility and does not have overhead related to transmission of coordination and scheduling information from Access Point (AP) to STAs. However, it lacks scalability and struggles to meet the Quality of Service (QoS) requirements of latency-sensitive and high-reliability applications, such as Extended Reality (XR) and Industrial Automation, especially in densely deployed network environments [1]. To address these challenges, the IEEE 802.11ax amendment (Wi-Fi 6) introduced, among other features, Orthogonal Frequency Division Multiple Access (OFDMA) as an alternative medium access method to better manage the shared wireless spectrum [2].

OFDMA partitions the available bandwidth into multiple sub-bands called Resource Units (RUs), which the AP can use to simultaneously schedule transmissions from multiple STAs by assigning each a different RU. This allows multiple STAs to transmit/receive data packets simultaneously thereby reducing contention and collisions substantially. This structured resource allocation opens up new possibilities for centralized resource management. In uplink (UL) OFDMA, for instance, EDCA can be completely disabled on STAs, making the AP the sole controller of all UL transmissions. However, to allocate RUs efficiently, the AP must have knowledge of each STA's buffer status. This necessitates the collection of Buffer Status Reports (BSRs) through UL polling mechanisms. As the number of STAs increases, polling becomes progressively more challenging and inefficient. Poorly timed and misdirected polling—such as assigning RUs to STAs with empty buffers while those with data to send remain unserved—can lead to significant resource underutilization and degraded network performance [3]. To address this, Wi-Fi 6 introduces a hybrid access mechanism known as Uplink OFDMA-based Random Access (UORA), which combines the benefits of both scheduled and random access for buffer status polling and data transmission.

During UORA transmissions, the AP divides the available RUs into Scheduled Access (SA) and Random Access (RA) categories, and announces their allocation via a Trigger Frame (TF). While scheduled STAs transmit without contention using their assigned SA RUs, all other STAs may contend for RA RUs using an UORA random access procedure (cf., Section III). This hybrid approach enables efficient scheduling of known buffered data while simultaneously supporting scalable, opportunistic polling for the unknown buffer status of STAs, thereby balancing centralized coordination with adaptive responsiveness. In this paper, we study specifically the performance of UORA as polling mechanism.

The main contributions of this paper are as follows:

- We evaluate the efficacy of UORA as a scalable alternative to buffer status polling mechanisms such as A2P [3] and SA UL OFDMA, hereafter also referred to as SA OFDMA.
- We investigate the impact of the minimum OFDMA contention window size on the performance of UORA under varying traffic loads.
- We demonstrate that, beyond a certain threshold of traffic sparsity, the performance advantage of UORA over SA OFDMA plateaus.

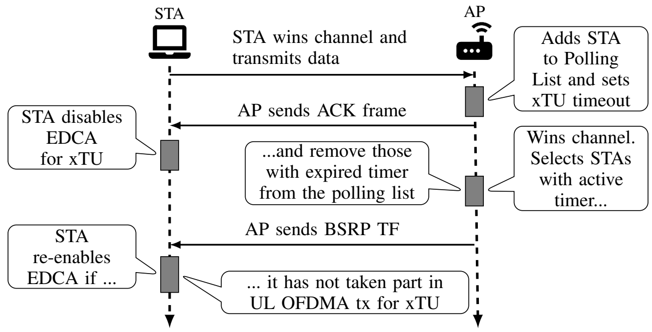


Fig. 1. Diagramatic representation of the A2P algorithm [3]

The remainder of the paper is organized as follows. Section II reviews recent research that addresses the polling problem and considers UORA in OFDMA-based Wi-Fi networks. Section III presents an overview of key OFDMA parameters and the UORA procedure. We discuss the relevant implementation details of the various schemes in Section IV. In Section V, we conduct a comparative performance evaluation of UORA against alternative approaches. Finally, Section VI concludes the paper.

II. Related Work

A considerable number of studies in the literature have explored strategies to enhance OFDMA performance in Wi-Fi networks, with a strong emphasis on resource allocation mechanisms [4–9]. However, the role of BSR collection—an essential prerequisite for effective uplink scheduling—is often underexplored. For instance, the resource allocation task has been framed as an optimization problem and addressed using a sub-optimal divide-and-conquer recursive algorithm [4]. Classical scheduling techniques—MaxRate, Proportional Fair, and Shortest Remaining Processing Time—have been adapted for Wi-Fi [5]. A scheduling algorithm for deadline-aware traffic has also been proposed, utilizing queuing theory on buffer state information reported by STAs to estimate the Head-of-Line (HOL) delay [6]. A related heuristic prioritizes transmissions based on buffer sizes across different access categories [7]. More recent efforts apply deep reinforcement learning to improve scheduling efficiency [8, 9]. While buffer status information is central to many of these approaches, the overhead and challenges associated with BSR collection are generally overlooked. To address these limitations, we proposed the A2P algorithm in our earlier paper [3]. A2P improves resource utilization efficiency by combining EDCA and OFDMA. The AP tracks a polling list of STAs expected to transmit, allowing them to use only UL OFDMA and bypass EDCA contention. A STA joins the list by initiating a transmission via EDCA. After successful reception, the AP disables EDCA for that STA for a preconfigured time period using the MU EDCA Parameter Set. STAs that have not reported any data after being polled during the pre-specified time period are removed from the polling list. Figure 1 illustrates the mechanism.

As it operates on a random access mechanism, UORA is susceptible to collisions and the subsequent need for retransmissions of lost packets. For this reason, it is most suited for BSRs, which are significantly smaller than full data packets, thereby limiting the impact of collision overhead. Several proposals in the literature seek to improve the performance of UORA. For example, researchers have explored more efficient ways of selecting the OBO counter [10–12], optimizing the back-off countdown procedure [13–15], improving resource allocation [16] and addressing collision resolution [17–19]. Other works have integrated UORA with recently introduced Wi-Fi features, such as multi-link operation [20], and target wake time [21]. Furthermore, some studies have proposed scheduling algorithms that combine both random and scheduled access procedures [16, 22]. The performance of UORA has also been evaluated using analytical modeling and simulation [23]. However, their publicly available UORA implementation in ns-3 exhibits several limitations, which we addressed in our work [24]. Our implementation is available as open source [25].

Several alternative solutions to UORA have been proposed in the literature. A fully deterministic channel access method has been introduced, in which the AP centrally schedules all uplink transmissions by disabling random access altogether [26]. Although this method is effective for predictable traffic patterns, it tends to perform poorly under bursty or unpredictable traffic conditions. Another approach involves using multiple rounds of BSRP Trigger Frames to collect BSRs from all STAs prior to scheduling, which enhances fairness but increases the overhead associated with BSR collection [27]. Additionally, a mechanism that enables client-side switching between EDCA and OFDMA based on buffer status has been proposed [28]. However, this conflicts with the standard, where the AP governs access states, and bypassing the AP risks inconsistent behavior and inefficient resource use. To the best of our knowledge, no prior work has evaluated the effectiveness of using UORA as a polling mechanism in comparison with other approaches such as SA UL OFDMA.

In this paper, we assess the performance of UORA-based polling for buffer status reporting and compare it against alternative mechanisms, including SA UL OFDMA and A2P. EDCA serves as the baseline approach.

III. Background: SA OFDMA and UORA in Wi-Fi

In this section, we detail the general principles of SA OFDMA, followed by a discussion of the MU EDCA Parameter Set, and conclude with the operational specifics of UORA.

A. *SA UL OFDMA*

OFDMA, debuted in the Wi-Fi 6 standard, enables simultaneous frame transmissions to and from STAs by dividing the available bandwidth into multiple RUs, which can be allocated to different STAs. The standard supports both downlink (DL) and UL OFDMA transmissions. However, as this work

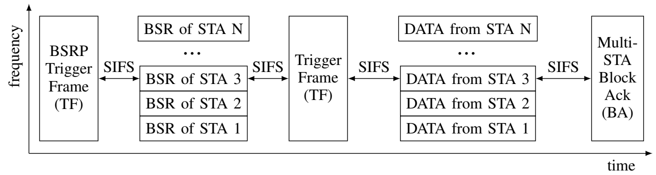


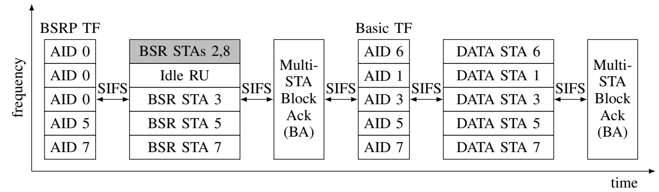
Fig. 2. UL OFDMA frame exchange sequence

focuses on UORA, which is an UL transmission mechanism, DL procedures are beyond the scope of this discussion.

Once the AP successfully gains channel access through contention, it initiates UL OFDMA transmission by broadcasting a Trigger Frame (TF), specifically a Basic TF. This frame includes, among other parameters, the mappings of selected STAs to their assigned RUs. Following a Short Inter-Frame Space (SIFS), the scheduled STAs transmit concurrently using their allocated RUs. Upon completion of the UL transmission, the AP responds, following another SIFS, with a Multi-STA Block Acknowledgement (BA) to confirm successful packet reception. To allocate RUs effectively for UL transmissions, the AP must first determine which STAs have buffered data. This is achieved by sending a special type of TF, known as a Buffer Status Report Poll (BSRP). Similar to a Basic TF, the BSRP frame includes RU assignments that indicate which STAs should respond. STAs transmit their buffer status reports (BSRs) either explicitly by sending a QoS Null frame–triggered when the user index in the BSRP TF matches their Association ID (AID)–or implicitly by embedding the buffer size in the QoS control field of any outgoing frame. Based on these reports, the AP identifies STAs with pending data and considers them for scheduling in subsequent UL transmissions. To enhance airtime fairness and minimize contention in dense deployments, the standard introduced the Multi-User (MU) EDCA Parameter Set. STAs participating in UL OFDMA transmissions apply this set of parameters to contend for the channel less aggressively or defer access entirely for a specified duration. Figure 2 illustrates the complete SA UL OFDMA frame exchange sequence.

B. MU EDCA Parameter Set

The MU EDCA Parameter Set includes EDCA parameters such as contention window and Arbitration Inter-Frame Space Number (AIFSN) that can be used by STAs after participating in UL OFDMA transmission. By using it, the AP can exert greater control over UL transmissions, while the STAs themselves compete less aggressively for the channel or do not compete at all. The AP announces the MU-EDCA Parameter Set through management frames. When the parameter set includes an AIFSN value of zero, it signals the STAs to completely disable EDCA-based contention. In such cases, the AP fully orchestrates UL transmissions, and STAs do not contend for medium access. In addition to EDCA parameters, the MU EDCA Parameter Set includes an MU EDCA timer that dictates how long a STA should apply the received parameters. The timer is reset each time the STA successfully



Before BSRP TF

STA	OBO
1	15
2	1
3	2
4	5
6	7
8	3

After BSRP TF

STA	OBO
1	12
2	0
3	0
4	2
6	4
8	0

Fig. 3. UL OFDMA frame exchange sequence with UORA. Tables in the bottom show the values of OFDMA Back-Off (OBO) for all STAs before and after BSRP TF is received. STAs whose OBO reaches 0 are framed with red.

transmits data via OFDMA and receives a corresponding Block Ack from the AP. If the timer expires without a successful transmission, the STA reverts to its default EDCA settings.

C. The UORA operation

As with all UL transmissions orchestrated by the AP, the AP initiates the transmission by broadcasting a TF to signal its start. In UORA, the AP can designate a subset of RUs in the TF for either RA, allowing all STAs to use them, or allocate an RU for SA, restricting its use to a single designated STA. During the association stage, the AP shares information about the OFDMA Contention Window (OCW) range defined by $EOCW_{min}$ and $EOCW_{max}$. These parameters are transmitted in the management frames, and their values can be adjusted on demand. If a STA receives a TF that does not explicitly assign it an RU, but the frame indicates that RA is permitted, the STA may attempt UL transmission via UORA—for instance, to transmit a new buffer status report. In this scenario, the STA initializes its OCW to $OCW_{MIN} = 2^{EOCW_{min}} - 1$ and randomly selects an initial OFDMA Back-Off (OBO) value within the range $[0, OCW]$. Upon receiving subsequent TFs, the STA decreases its OBO value by the number of RA RUs specified in the TF. If the updated OBO counter is less than or equal to the number of RA RUs, the STA randomly chooses one of advertised RA RUs in the TF and uses it to transmit. Following a successful transmission, the STA resets its OCW to OCW_{MIN}. However, if transmission fails (e.g., due to collision), the STA doubles its OCW up to an upper bound of $OCW_{MAX} = 2^{EOCW_{max}} - 1$.

Figure 3 depicts the UORA frame exchange sequence. The AP initiates an UL OFDMA transmission by sending a BSRP TF, which includes three RA RUs (denoted by AID 0) and two SA RUs assigned to STAs 5 and 7. This BSRP TF prompts STAs to report their buffer status, enabling the AP to identify which STAs require resources. After one SIFS, STAs 2, 3, 4, 7 and 8 transmit BSRs using either a randomly selected RA RU or an SA RU assigned to them. STAs 2, 3, and 8 are eligible to transmit in a RA RU, because their OBO values are less than or equal to 3, the number of RA RUs advertised in the BSRP TF. Specifically, STA 3 selects the third RA RU and successfully transmits, while STAs 2 and 8 transmit using the same RA RU, resulting in a collision (shown as the shaded area). Following this, the AP acknowledges the transmission by sending a Multi-STA Block ACK after a SIFS. STAs with unsuccessful transmissions double their OCW values and choose new OBO values. After another SIFS, the AP allocates RUs to STAs that have reported having data to transmit (STAs 3, 5, and 7). The AP can also allocate resources to STAs that have previously reported non-zero buffer statuses, or to STAs that it thinks might have data for transmission but failed to deliver their buffer statuses. For example, in the figure, the AP also allocates resources to STAs 1 and 6. It is also allowed to assign some of the RUs for random access, but the overhead due to collisions of the data packets is usually significantly higher than that for BSRs. This is because the transmission time of data packets is generally longer than that of BSRs. Note that RUs can be of different sizes depending on the needs of the STAs, but for simplicity, they are considered the same in the figure. This allocation is communicated through the Basic TF. Following one more SIFS, the STAs transmit on their respective RUs. To ensure synchronized transmission, smaller payloads are padded to match the size of the largest payload. Finally, the AP sends a Multi-STA Block ACK after a SIFS, thereby concluding the UL OFDMA transmission.

IV. Implementation

In this section, we discus the relevant implementation details of UORA, SA OFDMA, A2P, and EDCA. The operational differences among the various schemes are:

1) *UORA:* A subset of RUs is reserved for RA in the TF, allowing unscheduled STAs to transmit buffer status reports opportunistically.
2) *SA OFDMA:* All RUs are allocated for SA without reserving any RUs for RA.
3) *A2P:* STAs indicate their need for resources by transmitting their initial packet via the contention-based EDCA mechanism, thereby prompting the AP to schedule them in subsequent transmissions.
4) *EDCA:* All STAs contend for the channel using traditional EDCA, transmitting over the entire bandwidth upon winning access.

For the OFDMA-based mechanisms (UORA, SA OFDMA, and A2P), multi-user transmissions are intentionally delayed on both the AP and STAs for a brief period following system initialization. This delay ensures that initial setup procedures—such as establishing acknowledgments—are completed via EDCA, which remains uninterrupted by the absence of multi-user transmissions during this initialization phase. Additionally, the AP is configured to request channel access even when it does not have data queued for transmission. This configuration is necessary because our experiments do not include downlink traffic, and thus the AP would otherwise have no opportunity to contend for channel access. The time between consecutive access requests is referred to as the Access Request Interval (ARI).

In SA OFDMA and A2P, the AP allocates all available RUs to selected STAs for both buffer status reporting and data transmission in a round-robin manner. However, under UORA, only the RUs reserved for SA are assigned in round-robin fashion for BSRs, while all available RUs are scheduled in the Basic TF for data transmission. This implies that, RUs left unused during the BSRP/BSR exchange—either due to collision or lack of selection—are subsequently scheduled for data transmission. In A2P, the AP selects STAs from a polling list, whereas in SA OFDMA and UORA, all associated STAs are considered.

V. Performance Evaluation

We evaluate the performance of different schemes using the ns-3 network simulator [29]. In particular, we used our previously developed open-source UORA implementation [24, 25] for assessment and adapt the A2P algorithm [3] to suit the targeted traffic pattern. We compare the performance of the following four schemes; UORA, SA OFDMA, A2P and EDCA.

The comparative evaluation focuses on the *Uplink Delay* of the latency-sensitive STAs and the *Total Throughput* of all the associated STAs. *Uplink Delay* is defined as the time interval between the generation of a packet by a STA and its successful reception by the AP. *Total Throughput* refers to the aggregate rate, measured in packets per second, of packets successfully received by the AP from all STAs during the simulation.

A. Simulation Setup

In the experiments, we model a single Basic Service Set (BSS) consisting of multiple STAs and a single AP, both compliant with the Wi-Fi 6 (or newer) standard. The STAs are categorized into two groups, *deterministic* and *stochastic*, based on their traffic generation behavior. Deterministic STAs generate Constant Bit Rate (CBR) UDP traffic at approximately 6.54 Mbps (i.e., (1700×8) bits $\div\ 0.00208$ s), while stochastic STAs generate latency-sensitive UDP traffic with a fixed packet size of 1700 B following an exponentially distributed packet arrival rate. All traffic is assigned to the Voice (VO) Access Category (AC). This configuration emulates a network of wirelessly connected devices sharing the same AC, where some devices generate data at regular intervals, while others generate event-based latency-sensitive traffic unpredictably.

We set packet size to 1700 bytes to ensure that each transmission fits entirely within the allocated Transmit Opportunity (TXOP) for data transmission, particularly in the OFDMA based (i.e., SA OFDMA, UORA and A2P) scenarios where only 26-tone RUs are employed. In the A2P configuration, frame aggregation is disabled to avoid skewing the expected performance outcome, as it could allow multiple packets to be bundled into the initial EDCA transmission, thereby masking the intended behavior. Conversely, aggregation is enabled in the EDCA simulation to bolster performance.

We select the smallest RU type (i.e., 26-tones) to promote equitable distribution of resources among associated STAs and maximize the number of simultaneous transmissions. The number of deterministic STAs is equal to the number of 26-tone RUs in the chosen bandwidth. Additionally, downlink traffic generation is deliberately disabled, as this study focuses solely on the polling mechanisms used for UL transmissions.

In the UORA setup, EDCA is fully disabled by setting the MU EDCA timer to the full simulation duration and the AIFSN value to zero. This configuration delegates full control of UL transmission scheduling to the AP, with STAs refraining from any independent contention. Accordingly, the ARI is set to one Short Inter-Frame Space (SIFS) to enable frequent polling and UL data transmissions. In contrast, the A2P configuration allows the AP to request access less aggressively, enabling fairer contention opportunities for STAs that are not on the polling list. To avoid saturating the polling list with non-transmitting stochastic STAs, the MU EDCA timer is configured to 8 Time Units (TUs), which is the minimum duration allowed by the standard. Furthermore, the $\mathrm{OCW_{MIN}}$ value is varied across simulations to evaluate its effect on performance, while $\mathrm{OCW_{MAX}}$ is held constant at its maximum value.

To ensure that packet loss due to channel errors is negligible, we configure transmissions to be at sufficiently high power levels such that all STAs remain within the communication range of the AP. Consequently, packet losses only occur when multiple STAs transmit at the same time using the same resource (RU or bandwidth). The simulation parameters used across all scenarios are summarized in Table I.

TABLE I
LIST OF SIMULATION PARAMETERS

Parameter	Value
Carrier frequency	5 GHz
Bandwidth	20 MHz
Guard Interval	0.8 µs
MCS Index	8
Resource Unit Type	26-tone only
Transmit Opportunity	2.08 ms
AP Access Request Interval (ARI)	16 µs, UORA 128 µs, A2P
EDCA Access Category	VO
$\mathrm{OCW_{min}}$	{0, 1, 3, 7, 15, 31, 63}
$\mathrm{OCW_{max}}$	127
MU EDCA Timer	180 s, UORA 8 TUs, A2P
Payload Size	1700 B
Deterministic Inter-Packet Interval	2.08 ms
Stochastic Inter-Packet Interval	Exp. Distribution, (means: {0.03, 0.05, 0.1, 0.3, 0.5, 1.0} s)
Number of Deterministic STAs	9
Duration of Simulation	180 s

B. Discussion of Results

In this section, we compare the performance of UORA with the other schemes in terms of delay and throughput. We also investigate how the intensity of the sporadic traffic generated by the stochastic STAs influence the delay reduction associated with the usage of UORA.

Simulations corresponding to the results presented in Figures 4, 5, 6 and 7 are conducted using a packet generation model with an exponential inter-arrival time having a mean of 100 ms. Figures 4 and 5 show the uplink delay for a varying number of stochastic STAs generating latency-sensitive traffic. The results are represented in a box plot, where the median, lower, and upper quartiles, and extreme values of delay are clearly visible—particularly under higher numbers of stochastic STAs. We begin our analysis by examining the impact of the chosen $\mathrm{OCW_{min}}$ parameter on the performance of UORA. As illustrated in Figure 4, an inappropriate selection of OCWmin can significantly degrade UORA's performance relative to SA OFDMA (i.e., the configuration with 0 RA RUs). Conversely, an appropriate choice of $\mathrm{OCW_{min}}$ can yield notable improvements in delay performance. Specifically, setting $\mathrm{OCWmin} = 63$ results in performance inferior to SA OFDMA, whereas selecting $\mathrm{OCWmin} = 7$ leads to reduced average delay compared to $\mathrm{OCWmin} = 0$. Consequently, we select the optimum OCWmin value to show the best achievable performance of UORA in the subsequent analysis.

Average Delay: Figure 5 includes the EDCA result for only one stochastic STA, as EDCA's performance is already significantly degraded under this load. For A2P, results are shown for up to 60 STAs, since the delay increases and throughput decreases substantially beyond this point. Furthermore, each UORA box plot (i.e., 1-9 RA RUs) represents the distribution of individual packet delays obtained from the experiment configuration that yields the lowest average delay across the different $\mathrm{OCW_{min}}$ values for a given number of stochastic STAs N and number of RA RUs R. The optimal $\mathrm{OCW_{min}}$ value is determined independently for each (N, R) pair, meaning that the best-performing $\mathrm{OCW_{min}}$ setting may vary across different combinations of (N, R) pairs, as shown in Figure 6. To accomplish this, we conduct T independent experiment runs for every combination of number of stochastic STAs ($N \in \{1, 10, 20, \ldots, 90\}$), number of RA RUs ($R \in \{1, 3, 5, 7, 9\}$) and minimum OCW parameter $\mathrm{OCW_{min}} \in \{1, 3, 7, 15, 31, 63\}$. For each configuration $(N, R, \mathrm{OCW_{min}})$, we compute the average delay across the T runs as:

$$\bar{D}(N, R, \mathrm{OCW_{min}}) = \frac{1}{T} \sum_{t=1}^{T} D_t(N, R, \mathrm{OCW_{min}}),$$

where D_t is the average delay observed in the t^{th} run.
We then identify, for each (N, R) pair, the $\mathrm{OCW_{min}}$ that

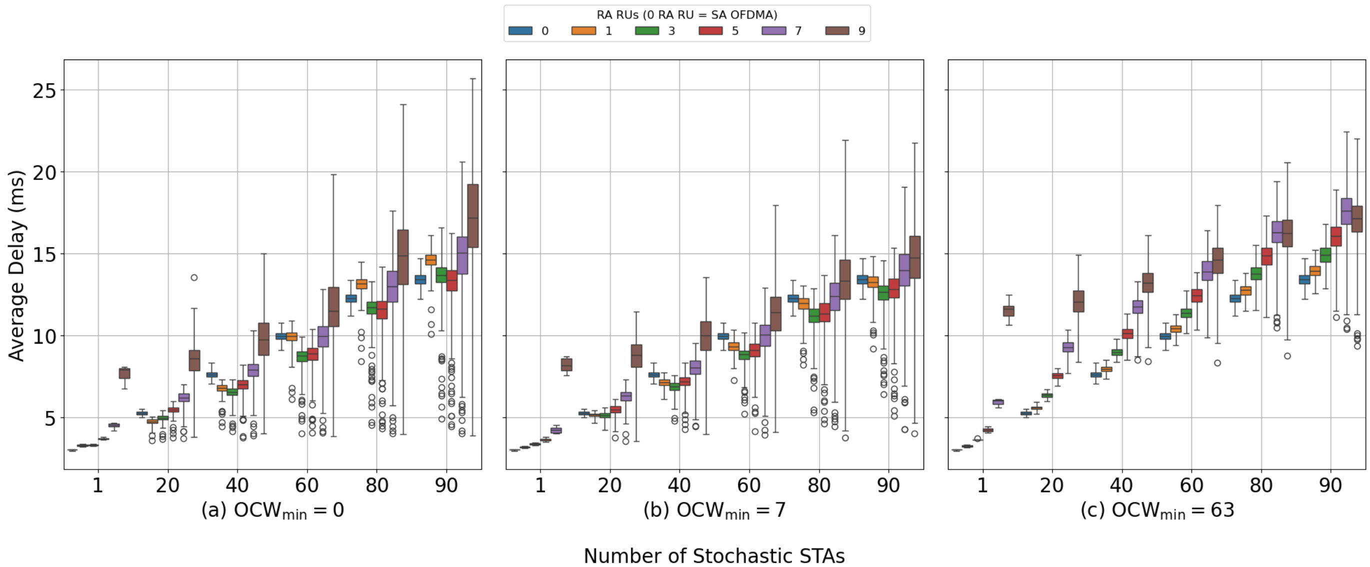


Fig. 4. Average delay of stochastic STAs as a function of the number of these STAs across different minimum contention window sizes

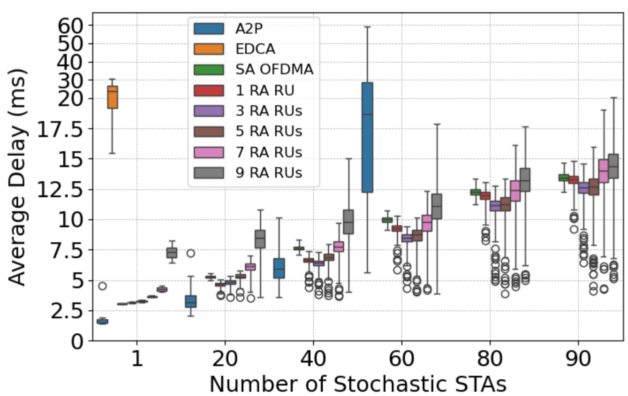


Fig. 5. Average delay of stochastic STAs as a function of the number of these STAs

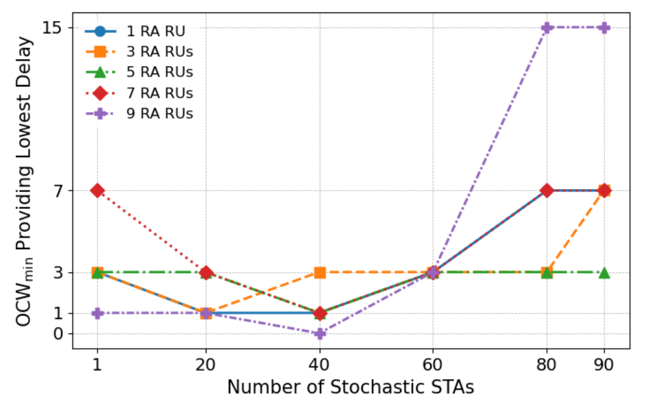


Fig. 6. OCWmin providing the lowest delays as a function of the number of stochastic STAs

results in the lowest average delay:

$$\mathrm{OCW}^*_{\min}(N,R) = \arg\min_{\mathrm{OCW}_{\min}} \bar{D}(N,R,\mathrm{OCW}_{\min})$$

Figure 6 depicts the results of the optimum $\mathrm{OCW}_{\min}$ value for each unique (N,R) pair. The final box plot for each (N,R) combination in Figure 5, visualizes the distribution of individual packet delays aggregated from the T runs conducted using that combination's identified optimal $\mathrm{OCW}^*_{\min}(N,R)$.

For instance, for 40 stochastic STAs and 3 RA RUs, all six $\mathrm{OCW}_{\min}$ settings are evaluated over T independent runs each, and the setting yielding the lowest average delay is selected. The corresponding box plot then aggregates the individual packet delays from all T runs conducted using that optimal $\mathrm{OCW}_{\min}$ value.

While in EDCA all associated devices (i.e., both deterministic and stochastic STAs) compete for the channel and are prone to packet losses due to collisions, the other schemes manage to orchestrate transmissions with little or no contention. In particular, A2P consistently disables EDCA on deterministic STAs and, for stochastic STAs, for 8 TUs, thereby reducing contention between STAs with new packets and the AP. As a result, A2P achieves the lowest delay for up to 40 stochastic STAs. The sharp increase in delay observed beyond this number of stochastic STAs is due to heightened contention among devices which prevents stochastic STAs from getting on the polling list to be subsequently allocated resources. With the gradual increase of stochastic STAs in the polling list, the scheduler's ability to efficiently allocate RUs to stochastic STAs, for buffer status reporting and subsequent data transmission, diminishes. This explains the gradual increase in delay with a growing number of stochastic STAs observed in the contention-free SA OFDMA scenario. Moreover, contention in UORA is less aggressive than in EDCA, as it utilizes multiple RUs. The aggressiveness of UORA contention depends on the values of $\mathrm{OCW}_{\min}$ and

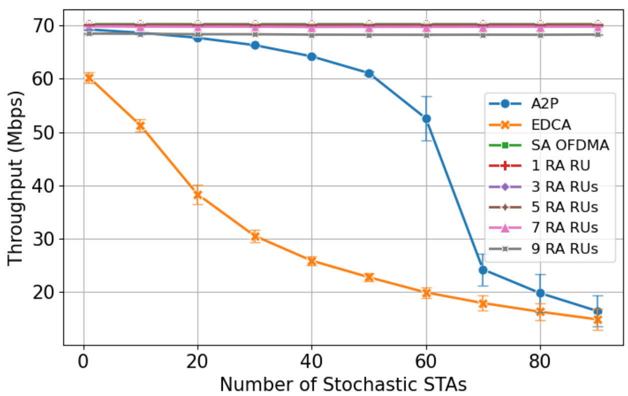


Fig. 7. Total throughput as a function of the number of stochastic STAs

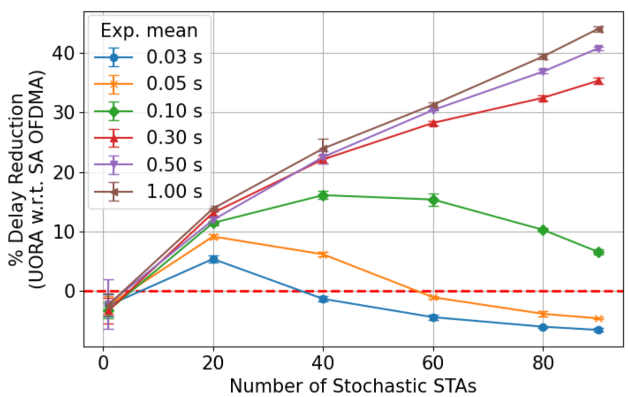


Fig. 8. Delay reduction achieved by UORA with respect to UL OFDMA as a function of the number of stochastic STAs for varying intensity of traffic generated on these STAs. Data points above the dashed red line indicate UORA outperforming SA OFDMA

the number of RA-RUs, both of which can be dynamically adjusted based on network load. As visualized in Figure 6, different traffic loads—characterized by the number of stochastic STAs—and varying allocations of RA RUs require distinct contention window sizes in order to achieve optimal UORA performance. Both SA OFDMA and UORA emerge as the most scalable solutions, maintaining delay below 15 ms while supporting up to 90 stochastic STAs.

The results further show that increasing the number of RUs reserved for random access reduces delay as the population of latency-sensitive devices grows—up to the point where RUs are equally divided between scheduled access (SA) and RA (i.e., 5 RA RUs). Beyond this midpoint of equal RA and SA allocation, the delay begins to increase again. This behavior can be attributed to the trade-off between contention-based (UORA) and contention-free (SA OFMDA) access mechanisms. As more RUs are allocated for RA, stochastic STAs benefit from increased transmission opportunities by not having to wait to be scheduled, which helps reduce delay, particularly when the number of such devices is high. However, once the RA allocation surpasses the point of balance with SA, more stochastic STAs begin to rely on UORA, which exacerbates contention and collision rates, ultimately increasing the delay experienced by packets.

Throughput: As discussed in Section IV, with SA OFDMA and UORA, resource allocation for BSRs from STAs is decoupled from RUs assignment for data transmission. As such, RUs in the BSRP TF that experience collisions during BSR transmission can be rescheduled in Basic TF for uplink data transmission. In our simulations, the AP reserves RUs for RA only in the BSRP TF, while all RUs in the Basic TF are assigned for SA uplink data transmission. Consequently, the throughput remains fairly constant throughout the experiment, as shown in Figure 7. The relatively lower throughput observed when all RUs are reserved for RA is attributed to the contention-based mechanism used to select an RU for BSR transmission. This mechanism can lead to resource underutilization, as the AP may fail to receive BSRs from multiple STAs, including those with deterministic traffic patterns.

Delay Reduction: We further investigate the variations in delay between SA OFDMA transmissions (i.e., RA RU = 0) and UORA-based (i.e., RA RUs $>$ 0) transmissions under varying traffic intensities generated by critical STAs. The objective is to study how the delay reduction that can be obtained by using UORA varies with the intensity of latency-sensitive traffic required for UORA to provide measurable performance benefits. Figure 8 depicts the percentage of delay reduction achieved with UORA relative to the baseline case of SA OFDMA across different values of the exponential mean used in the traffic generation model. Values above 0 indicate that UORA outperforms SA OFDMA, whereas values below 0 indicate the opposite. We compute the delay reduction as follows:

$$D_{gain} = \frac{D_{base} - D_{min}}{D_{base}} \times 100,$$

where D_{min} denotes the minimum delay observed across the different values of OCW_{min} and number of RA RUs for each configuration of exponential mean and number of critical STAs when using RA RUs. The delay when no RA RUs are used is denoted by D_{base}. Additionally, the error bars represent the standard deviation across the T experimental runs.

The results show that reserving RUs for RA becomes less advantageous when a large number of critical STAs are actively and frequently generating data. In such scenarios, the contention-based nature of UORA leads to increased collisions, thereby diminishing its performance benefits. Conversely, UORA is more effective under sparse traffic conditions. For example, under relatively higher traffic loads—characterized by exponential means of 0.30 s and 0.50 s—the performance gains of UORA begin to decline beyond 40 and 60 critical STAs, respectively. Furthermore, the advantages of using UORA tend to plateau when the traffic becomes sufficiently sparse, as observed with an exponential mean of 0.30 s and higher.

VI. Conclusion

In this work, we evaluate the efficacy of UORA as a scalable alternative to buffer status polling mechanisms such as A2P and SA UL OFDMA. EDCA serves as a baseline scheme. We demonstrate that UORA effectively mitigates the limitations of both the contention-based channel access (EDCA) and centralized polling (SA UL OFDMA), while also offering advantages over hybrid approaches like A2P, particularly in dense environments with sparsely sporadic uplink traffic. The results show that UORA achieves lower delay and higher throughput compared to the alternative approaches in scenarios with a large number of stochastic STAs generating latency-sensitive traffic, underscoring its potential to enhance UL performance in Wi-Fi networks.

In our future work, we plan to investigate ways for the optimal selection of UORA parameters, e.g., using mathematical optimization.

Acknowledgements

The research was supported by the FWO WaveVR project (G034322N).

References

[1] T. Adame *et al.*, "Time-sensitive networking in ieee 802.11be: On the way to low-latency wifi 7," *Sensors*, vol. 21, no. 15, 2021, ISSN: 1424-8220. DOI: 10.3390/s21154954. [Online]. Available: https://www.mdpi.com/1424-8220/21/15/4954.

[2] IEEE Std 802.11ax-2021, "Enhancements for High Efficiency WLAN," IEEE, Standard, 2021.

[3] D. D. Agbeve *et al.*, "A2P: A Scalable OFDMA Polling Algorithm for Time-Sensitive Wi-Fi Networks," in *2025 IEEE Wireless Communications and Networking Conference (WCNC)*, 2025, pp. 1–6. DOI: 10.1109/WCNC61545.2025.10978627.

[4] K. Wang *et al.*, "Scheduling and Resource Allocation in 802.11ax," in *Proc. of IEEE INFOCOM*, 2018, pp. 279–287. DOI: 10.1109/INFOCOM.2018.8486204.

[5] D. Bankov *et al.*, "OFDMA Uplink Scheduling in IEEE 802.11ax Networks," in *Proc. of IEEE ICC*, 2018, pp. 1–6. DOI: 10.1109/ICC.2018.8422767.

[6] M. Inamullah *et al.*, "Will My Packet Reach On Time? Deadline-Based Uplink OFDMA Scheduling in 802.11ax WLANs," in *Proc. of the 23rd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, Association for Computing Machinery, 2020, pp. 181–189. DOI: 10.1145/3416010.3423232.

[7] Y. A. Qadri *et al.*, "Preparing Wi-Fi 7 for Healthcare Internet-of-Things," *Sensors*, vol. 22, no. 16, 2022. DOI: 10.3390/s22166209.

[8] H. Noh *et al.*, "Joint Optimization on Uplink OFDMA and MU-MIMO for IEEE 802.11 ax: Deep Hierarchical Reinforcement Learning Approach," *IEEE Communications Letters*, 2024.

[9] Q. Tan *et al.*, "Deep Reinforcement Learning based OFDMA Scheduling for WiFi Networks with Coexisting Latency-Sensitive and High-Throughput Services," in *2024 5th ICTC*, IEEE, 2024, pp. 146–150.

[10] A. Rehman *et al.*, "Collision-Based Up-Link OFDMA Random Access Mechanism for Wi-Fi 6," *IEEE Access*, vol. 11, pp. 117 094–117 109, 2023.

[11] Y. Kim *et al.*, "OFDMA Backoff Control Scheme for Improving Channel Efficiency in the Dynamic Network Environment of IEEE 802.11ax WLANs," *Sensors*, vol. 21, no. 15, p. 5111, 2021.

[12] P. Liu *et al.*, "Performance optimization of ieee 802.11ax ul ofdma random access," *Journal of Communications and Networks*, vol. 26, no. 6, pp. 580–592, 2024. DOI: 10.23919/JCN.2024.000069.

[13] K. Kosek-Szott *et al.*, "An Efficient Backoff Procedure for IEEE 802.11ax Uplink OFDMA-based Random Access," *IEEE Access*, vol. 10, pp. 8855–8863, 2022.

[14] K. Kosek-Szott *et al.*, "Improving IEEE 802.11ax UORA Performance: Comparison of Reinforcement Learning and Heuristic Approaches," *IEEE Access*, vol. 10, pp. 120 285–120 295, 2022.

[15] A. Rehman *et al.*, "Enhancing wi-fi 6 spectrum access control with a heuristic ofdma back-off mechanism," *Results in Engineering*, vol. 26, p. 105 086, 2025, ISSN: 2590-1230. DOI: https://doi.org/10.1016/j.rineng.2025.105086. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2590123025011612.

[16] S. Bhattarai *et al.*, "Uplink Resource Allocation in IEEE 802.11ax," in *Proc. of IEEE ICC*, 2019, pp. 1–6. DOI: 10.1109/ICC.2019.8761594.

[17] E. Avdotin *et al.*, "OFDMA Resource Allocation for Real-Time Applications in IEEE 802.11ax Networks," in *Proc. of IEEE BlackSeaCom*, 2019, pp. 1–3. DOI: 10.1109/BlackSeaCom.2019.8812774.

[18] E. Avdotin *et al.*, "Enabling Massive Real-Time Applications in IEEE 802.11be Networks," in *Proc. of IEEE PIMRC*, pp. 1–6. DOI: 10.1109/PIMRC.2019.8904271.

[19] D. Xie *et al.*, "Multi-Dimensional Busy-Tone Arbitration for OFDMA Random Access in IEEE 802.11ax," *IEEE Transactions on Wireless Communications*, vol. 19, no. 6, pp. 4080–4094, 2020.

[20] M. Jin *et al.*, "Enhancing the Performance of Wi-Fi 7 Network via Investigation of Multi-link and UORA," *IEEE Communications Letters*, 2024.

[21] Q. Chen *et al.*, "Scheduling Channel Access Based on Target Wake Time Mechanism in 802.11ax WLANs," *IEEE Transactions on Wireless Communications*, vol. 20, no. 3, pp. 1529–1543, 2020.

[22] E. Avdotin *et al.*, "Resource Allocation Strategies for Real-Time Applications in Wi-Fi 7," in *Proc. of IEEE BlackSeaCom*, 2020, pp. 1–6. DOI: 10.1109/BlackSeaCom48709.2020.9234994.

[23] G. Naik *et al.*, "Performance Analysis of Uplink Multi-User OFDMA in IEEE 802.11ax," in *Proc. of IEEE ICC*, 2018, pp. 1–6. DOI: 10.1109/ICC.2018.8422692.

[24] D. D. Agbeve *et al.*, "Design and Evaluation of IEEE 802.11ax Uplink Orthogonal Frequency Division Multiple Random Access in ns-3," in *Proc. of ICNS3*, 2025.

[25] D. D. Agbeve, *WiFi UORA in NS-3*, 2025. [Online]. Available: https://github.com/imec-idlab/WiFi_UORA_ns3.

[26] B. Schneider *et al.*, "Deterministic Channel Access Using MU EDCA in OFDMA-based Wi-Fi Networks," in *Proc. of IEEE WFCS*, 2023, pp. 1–4. DOI: 10.1109/WFCS57264.2023.10144244.

[27] S. Shao *et al.*, "Access optimization in 802.11 ax wlan for load balancing and competition avoidance of iptv traffic," *IEEE Transactions on Broadcasting*, 2024.

[28] V. D. S. Goncalves *et al.*, "Access priority adaptation for triggered uplink channel access in 802.11 ax wlans," in *Proc. of IEEE ICC*, 2023, pp. 1061–1067.

[29] R. T. Henderson *et al.*, "Network Simulations with the ns-3 Simulator," *SIGCOMM demonstration 14*, p. 527, 2008.