Experimental description of Q-L-AOLLA

I. Introduction

This article gives a description of the complete process of deploying the four algorithms and experimental validation mentioned in the article, Reinforcement Learning for a Deployment-friendly Adaptive Outer Loop Link Adaptation, and instructions for using the source code. A complete log of the experimental data from the paper and the training model for reinforcement learning is provided in the attached file.

II. Operation of IEEE 802.11ax

The wireless transmission standard used in our experiments is IEEE 802.11ax with HESU format packets. The frame encapsulation process, packet format and receiving process are summarized from Fig. 1 to Fig. 3. In order to improve the transmission efficiency, we enhance the payload ratio by means of frame aggregation, specifically by concatenating multiple MPDUs in Figure 1 together as a PSDU into a PPDU packet to be sent out at once. However, the high frame aggregation level leads to an increase in frame duration. On the one hand, the standard specifies that the frame duration should be less than 5.484ms, and on the other hand, once the frame duration is longer than the channel correlation time, it will lead to severe signal fading, which will cause packet loss at the receiver and affect the transmission efficiency. After our experimental verification, the frame aggregation level of 12 is a reasonable choice, and finally, the specific parameters of the HESU packet are set as shown in Table I. The specific information of the MCS is shown in Table II. It is important to note that all four link adaptive algorithms to be given below are to be based on the feedback packets from the receiver to the transmitter, and this can be achieved through the 802.11 transmission mechanism, as shown in Fig. 4, where we feed the information needed by the algorithm in the feedback ACK packets.

III. Implementation of AARF

In this section we give the principle of the AARF implementation. This algorithm requires information of the CRC result of the last transmitted packet, and the next MCS is selected based on the CRC result. This algorithm

TABLE I
Configuration of IEEE 802.11ax

IEEE 802.11ax	Channel bandwidth	40 MHz
	Antenna	SISO
	APEP length	variable
	MCS	variable
	Cyclic prefix duration	3.2 us
	Number of HE-LTF	4

TABLE II

MODULATION ORDER, CODE RATE AND PEAK DATE RATE OF 40 MHZ HESU

MCS	Modulation order	Code rate	Peak data rate (Mbps)
0	BPSK	1/2	14.625
1	QPSK	1/2	29.25
2	QPSK	3/4	43.875
3	16-QAM	1/2	58.5
4	16-QAM	3/4	87.75
5	64-QAM	2/3	117
6	64-QAM	3/4	131.625
7	64-QAM	5/6	146.25
8	256-QAM	3/4	175.5
9	256-QAM	5/6	195
10	1024-QAM	3/4	219.375
11	1024-QAM	5/6	243.75

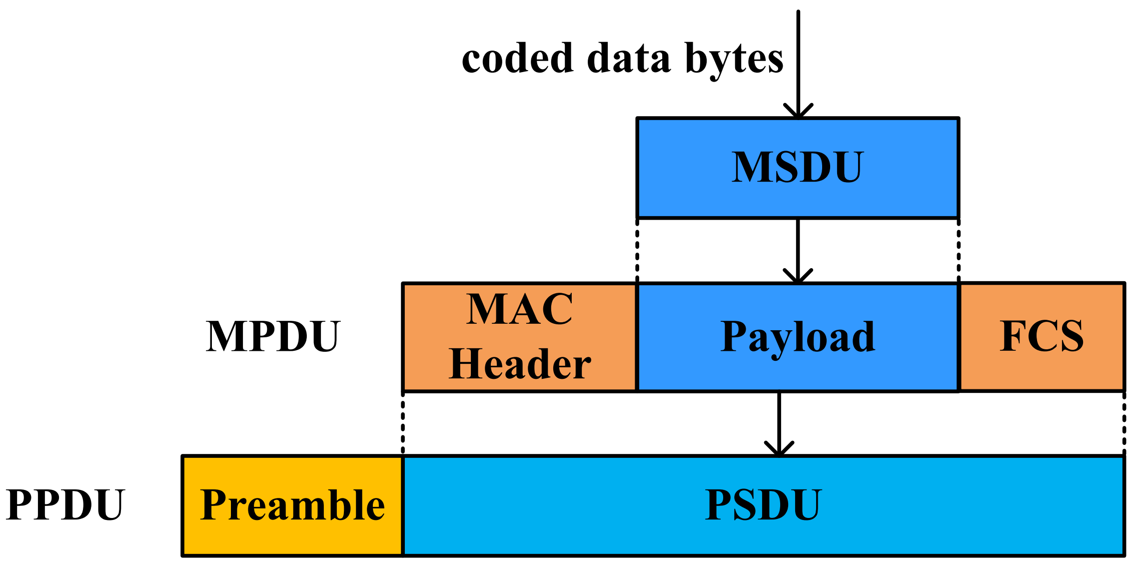

Fig. 1. IEEE802.11ax frame encapsulation process.

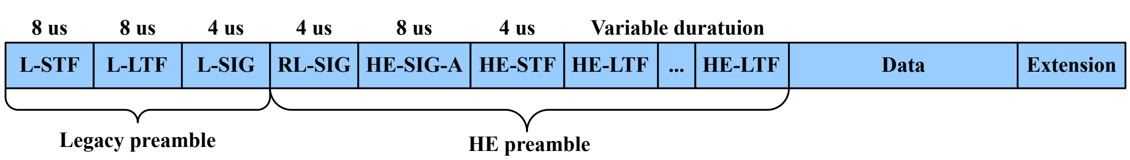

Fig. 2. IEEE802.11ax HESU packet format.

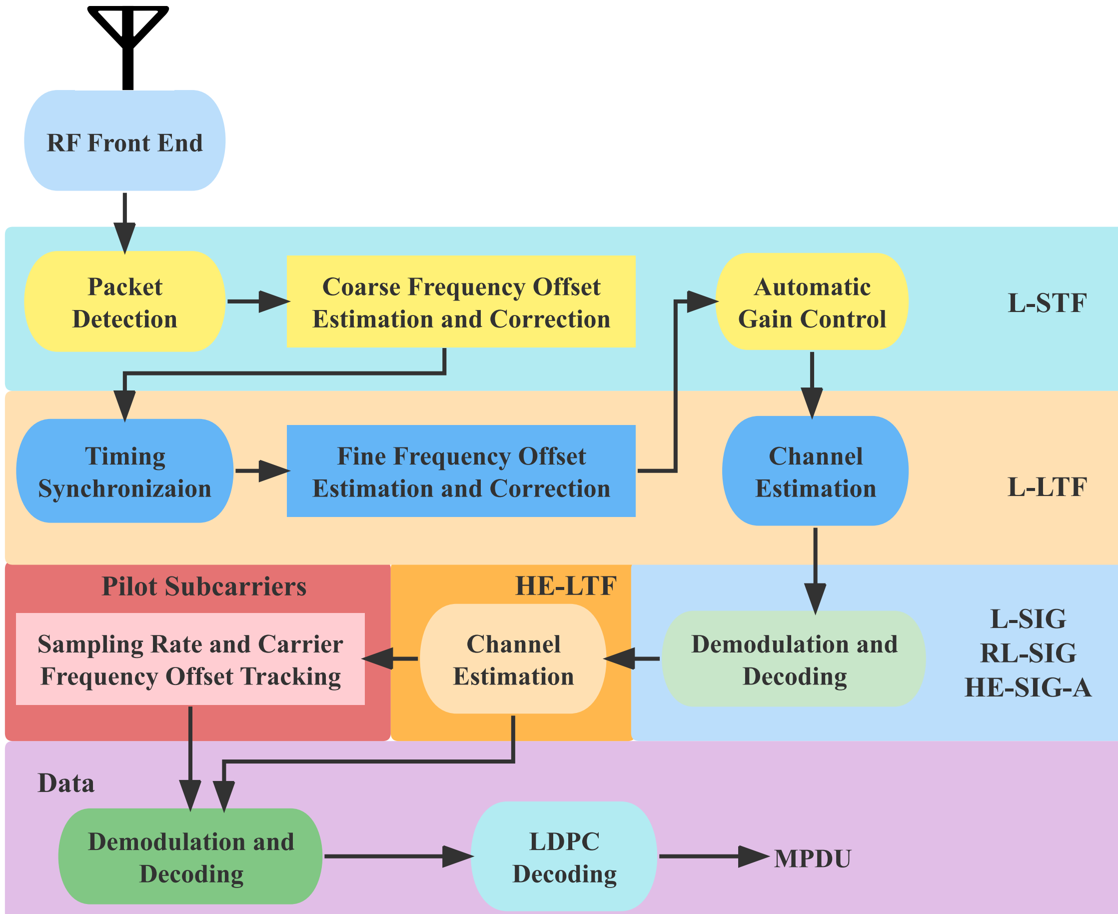

Fig. 3. HESU packet receiving process and the functions of each field.

can be specifically represented by a decision tree diagram shown in Fig. 5, where the initial MCS is set to 4, the window length n is set to $n_{min} = 10$ and $n_{max} = 50$, and the initial window length is set to 10.

IV. Implementation of OLLA

In this section we give the principle of the OLLA implementation. Compared to AARF, OLLA requires an offline link simulation with a pre-generated offline link model (OLM) and SNR estimation.

Firstly, to obtain an offline link simulation we simulated the IEEE 802.11ax link under TGax Model-B channel model, which is modeled by IEEE 802.11ax work group on common indoor wireless multipath fading channel model. Then, by adding white noise with different variance to filtered signal, we can assign the SNR of signal and the packet error rate under specific SNR was calculated after demodulation, decoding and CRC. We simulated 1000 HESU packets under each SNR ranging from 7 dB to 35 dB with a step of 4 dB, and if 50 packets were already failed to pass CRC, the simulation went on the next SNR directly. Eventually we obtained the SNR to packet error rate curves covering different MCSs shown in Fig. 6. Secondly, we obtained a SNR threshold vector through a horizontal line with a packet error rate equal to 10% across the SNR-PER curves of different MCS. The

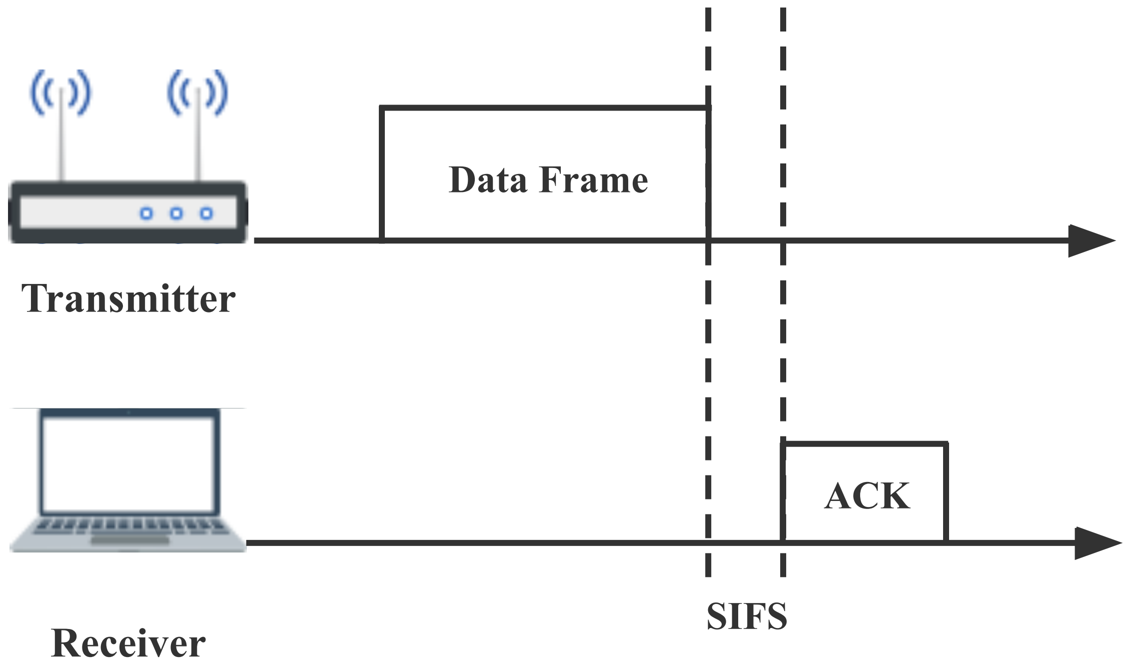

Fig. 4. The transmission mechanism in IEEE 802.11ax.

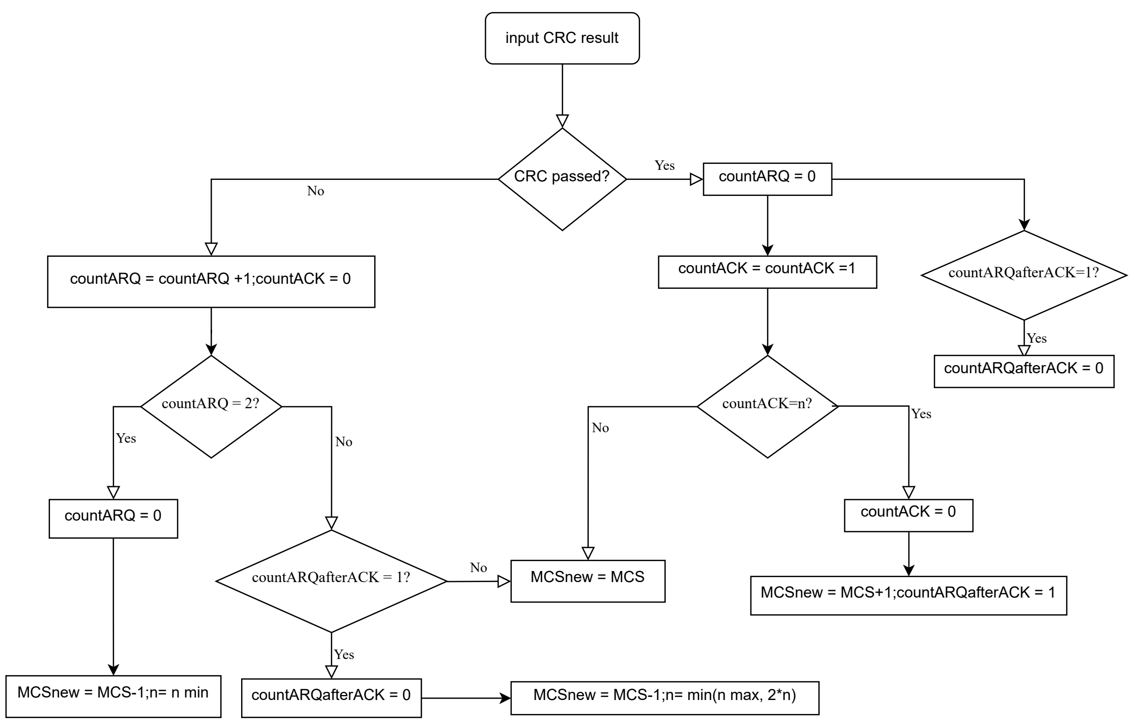

Fig. 5. Implementation of AARF.

SNR threshold vector is listed in Table III. Thirdly, we estimated the SNR of each packeted at the receiver. The power of the received signal of a HESU packet is expressed as:

$$P_{rx} = \frac{\mathbf{rxHELTF}^H \mathbf{rxHELTF}}{N_{rxHELTF}}, \tag{1}$$

where $\mathbf{rxHELTF}$ is a $N_{rxHELTF} \times 1$ complex-vaule vector which stands for the sampling data segment of HELTF field in the received HESU packet. The power of noise is estimated in the process of Data field channel equalization, which is the variance of the noise used in the MMSE equlizer. To estimate the variance of noise in a received HESU packet, we need to obtain the error matrix at the start, which is expressed as:

$$\mathbf{\Gamma} = \mathbf{H}_{pilotData} - \mathbf{H}_{pilotHELTF}, \tag{2}$$

where $\mathbf{H}_{pilotData}$ and $\mathbf{H}_{pilotHELTF}$ represent the QAM symbols of pilot subcarriers, which are in the process of OFDM demodulation after cyclic prefix removing and FFT but without equalization. $\mathbf{H}_{pilotHELTF}$ stands for the HELTF field symbols. $\mathbf{H}_{pilotData}$ stands for the Data field symbols and need to flip the sign at corresponding subcarriers to keep the same sign as the $\mathbf{H}_{pilotHELTF}$ symbols. The information of pilot symbols is specified in IEEE 802.11ax standard. Afterwards, the noise variance is estimated as:

$$\hat{\sigma^2} = \frac{\|\mathbf{\Gamma}\|_F^2}{N_{pilot} N_{sym}} = \frac{\mathbf{Tr}(\mathbf{\Gamma}^H \mathbf{\Gamma})}{N_{pilot} N_{sym}}, \tag{3}$$

where N_{pilot} represents the number of pilot subcarriers and N_{sym} represents the number of OFDM symbols in Data field. $N_{pilot} \times N_{sym}$ is the dimension of matrix $\mathbf{\Gamma}$. As a result, SNR is estimated as:

$$S\hat{N}R = \frac{P_{rx}}{\hat{\sigma^2}}. \tag{4}$$

OLLA algorithm adds an offset to the $S\hat{N}R$ to correct the estimated SNR, and then compares the modified estimated SNR to the threshold value:

$$S\hat{N}R_{OLLA} = S\hat{N}R - \Delta_{OLLA}. \tag{5}$$

The value of Δ_{OLLA} is based on the CRC result of the last packet. If CRC is passed, Δ_{OLLA} will decreased by a fixed step Δ_{down}, and if CRC is failed, Δ_{OLLA} will increased by a fixed step Δ_{up}. The ratio between the predefined step values determines the PER_{target} as:

$$PER_{target} = \frac{1}{1 + \frac{\Delta_{up}}{\Delta_{down}}}. \tag{6}$$

Since we set our PER_{target} to 10%, we set Δ_{up} to 1dB and Δ_{down} to 0.1dB in our experiment.

V. Experimental Validation on an SDR-based Hardware Platform

In this section we introduce how to validate the algorithms on an SDR-based Hardware Platform, which was used in the paper. The four link adaptation algorithms were validated under three different environments, In particular, the Q-learning-based AOLLA algorithm requires a training phase in one environments.

The equipment required to reproduce the experiments in the paper are described as follows.

- ADALM-PLUTO SDRs: Two ADALM-PLUTO SDRs are required and controled by two computers respectively as transmitter and receiver. Table IV lists the main features of this SDR hardware platform. From Table

TABLE III
SNR THRESHOLD VECTOR

MCS	SNR threshold (dB)
1	10
2	13
3	15
4	20
5	23
6	24.5
7	27.5
8	29.5
9	32
10	34
11	37

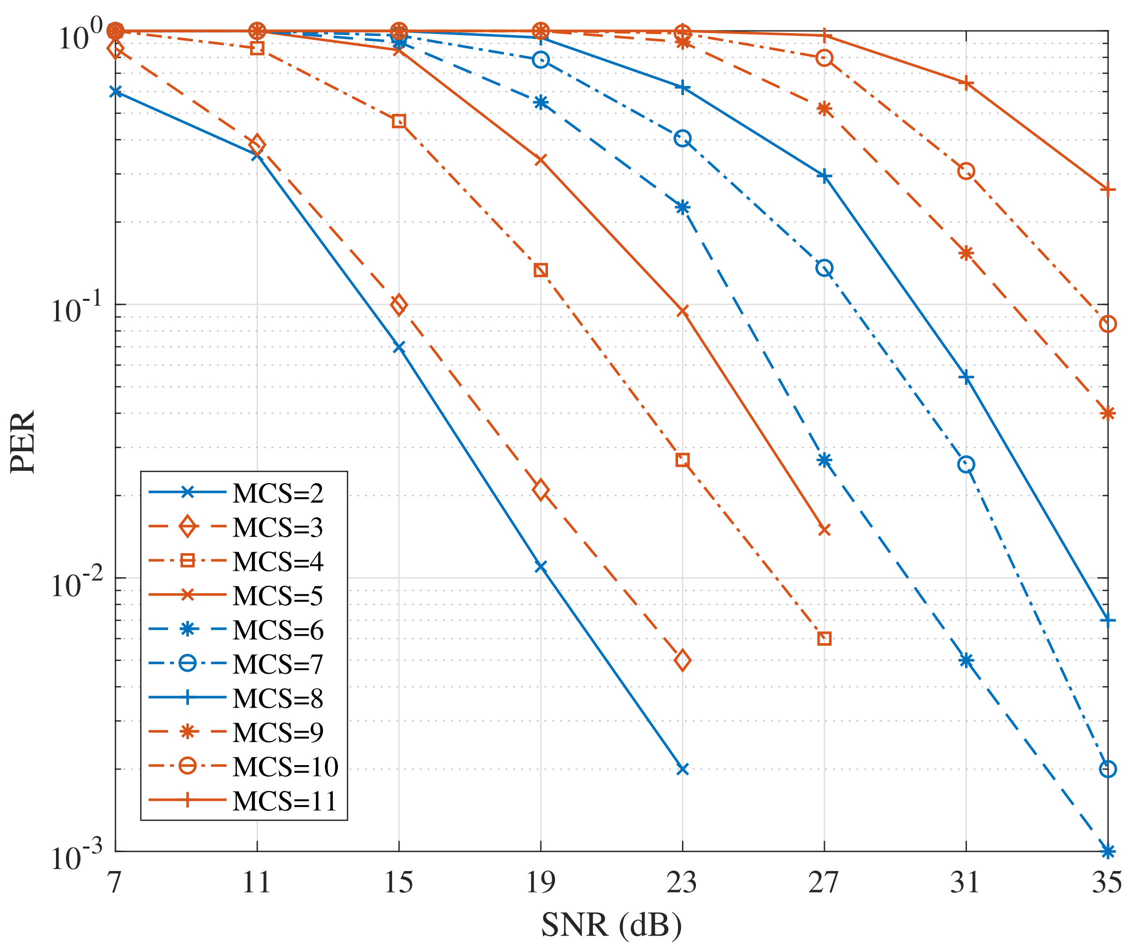

Fig. 6. SNR to packet error rate curves covering different MCSs simulated under TGax Model-B channel model.

TABLE IV
MAIN FEATURES OF ADALM-PLUTO SDR

Parameter	Value
Size	117 mm × 79mm × 24mm
Weight	114g
Power supply	5 V, 500 mA (USB 2.0)
Interface with PC	USB 2.0
ADC/DAC resolution	12 bits
ADC/DAC sampling rate	65.2 kSPS to 61.44 MSPS
Bandwidth	Up to 56MHz
Antenna	SISO
RF frequency range	70MHz to 6GHz
Crystal oscillator	25 ppm

IV, USB 2.0 power supply indicates a low-power-consumption with a portable size and weight. IEEE 802.11ax HESU data frame transmission can be realized on ADALM-PLUTO SDR based on its noise performance and baseband capability. The utilization of phase offset tracking algorithm in received data processing significantly reduces the damage of sampling clock offset caused by a non-high-cost, originally mounted, crystal oscillator with 25 ppm to decode a 40 MHz bandwidth, 1024-QAM HESU packet successfully. Both sides of SDR realized a half-duplex communication to fulfil the transmission task.

- Computers: Two computers are required so that each ADALM-PLUTO SDR is connected to a computer. Matlab with the versions 2020b is required to be installed in the computers. The computers we used have Intel Core i5-1035G4 (1.10 GHz) processor, 8 GB of RAM and Microsoft Windows 10 operating system.

To reproduce the 4 link adaptive algorithms based on the IEEE 802.11ax standard on the ADALM-PLUTO SDR platform, we first need two computers with Matlab 2020b to install the accessory function, *Communications Toolbox Support Package for Analog Devices ADALM-Pluto Radio*. Secondly, to run all the code in the attached file, make sure that Matlab 2020b on both computers can run the example, *Recovery Procedure for an 802.11ax Packet*,in Matlab WLAN Toolbox. It is recommended to run the example with the Add to path option as a convenient way to do so. Because the decision of the next MCS for the four link adaptive algorithms we designed is done at the receiver, the transmitter only needs to set the MCS for the next transmission according to the information in the received feedback packet, so for all four algorithms, the transmitter only needs to run the same code, *tx\SDRWifi6tx.m*. To validate the AARF algorithm you need to run the code, *rx\AARF\SDRwifi6rxAARF.m*, at the receiver. To validate the OLLA algorithm you need to run the code, *rx\OLLA\SDRwifi6rxOLLA.m*, at the receiver. To validate the AOLLA algorithm you need to run the code, *rx\AOLLA\SDRwifi6RxAOLLA.m*, at the receiver. As for the Q-learning-based AOLLA algorithm, If you want to validate the model we have trained, you need to run the code, *rx\Q-Learning-AOLLA\SDRwifi6rx1104QLearning.m*. This program will directly load the Q table, *Qtable1114.mat*, which is the action-value function tables of the β_1 and β_2, trained according to the method in our paper. If you want to train your own model from the beginning by following the reinforcement learning training method in our paper, you need

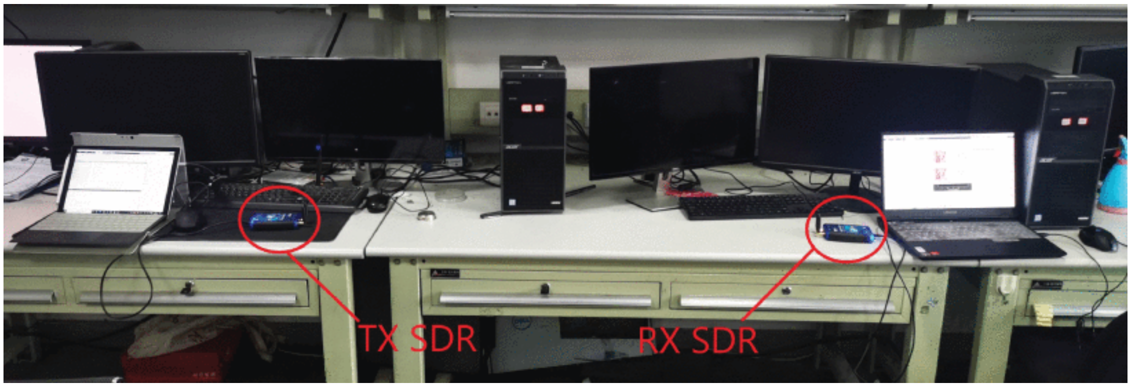

Fig. 7. Performing experiments in the office environment.

to run the code, *rx\Q-Learning-AOLLA\SDRwifi6rxQLearningTrain.m*. Each of the above five programs generates the corresponding data sets, including CRC results during transmission, MCS selection results, estimated SNR and OLLA offset values for algorithms except for the AARF algorithm. It helps us to calculate the data rate, PER, and analyze the algorithm in detail ,after a simple cleaning of invalid data. The first row of matrix *feedbackState* represents the CRC results with the transmission packet index. The second row of matrix *feedbackState* represents the MCS selection results with the transmission packet index. The 5th row of matrix *feedbackState* represents the estimated SNR with the transmission packet index. The vector *deltaThresholdList* represents OLLA offset values for algorithms except for the AARF algorithm. The experimental data in our paper are computed from these datasets, and we provide the logs of these datasets in the attached file, *log.xlsx*. Next, we describe if we validate the four link adaptive algorithms we designed under three different environments, office, hallway, and classroom, and if you want to get your own dataset logs, you can implement them as we do next.

First we validate our algorithm in an office environment, and the experimental setup is shown in Fig. 7. It is worth noting that the placement of the two SDRs can be more arbitrary or even NLOS, and the composition of the platform is only shown in Fig. 7 for clarity. The SNR, data rate and PER comparisons for the four algorithms obtained from the dataset logs are shown in Fig. 8 and 9. Our trained reinforcement learning model was trained in this environment, and if you want to retrain your own reinforcement learning model you can also run the corresponding program in this environment.

Secondly, we validate our algorithm in a hallway environment, and the experimental setup is shown in Fig. 10. It is worth noting that the placement of the two SDRs can be more arbitrary or even NLOS, and the composition of the platform is only shown in Fig. 10 for clarity. The SNR, data rate and PER comparisons for the four algorithms obtained from the dataset logs are shown in Fig. 11 and 12.

Finally, we validate our algorithm in a classroom environment, and the experimental setup is shown in Fig. 13. It is worth noting that the placement of the two SDRs can be more arbitrary or even NLOS, and the composition of the platform is only shown in Fig. 13 for clarity. The SNR, data rate and PER comparisons for the four algorithms obtained from the dataset logs are shown in Fig. 14 and 15.

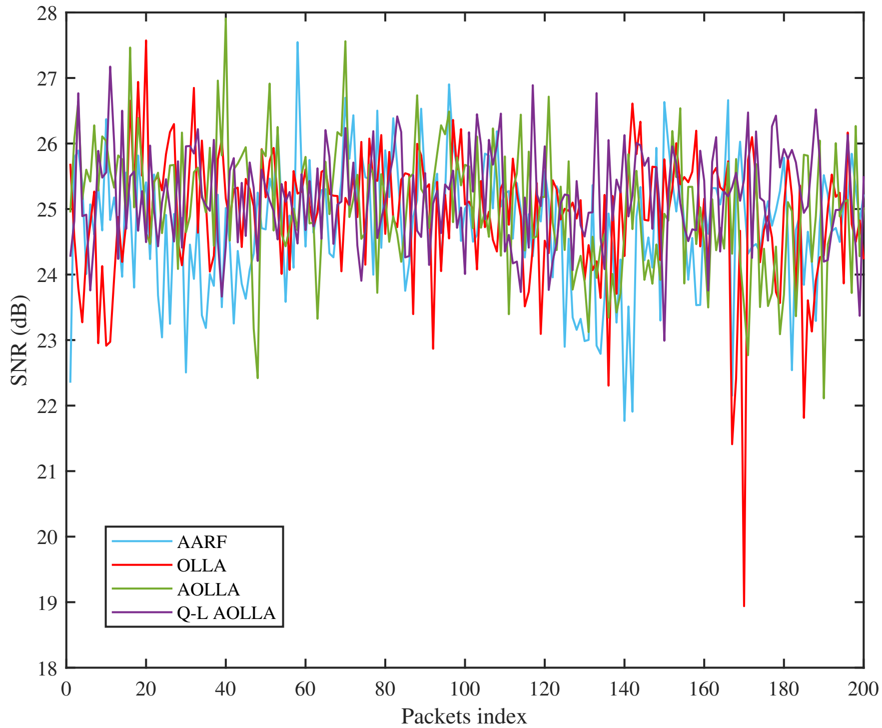

Fig. 8. SNR of 4 link adaptive algorithms, evaluated in office environment (Q-learning training environment).

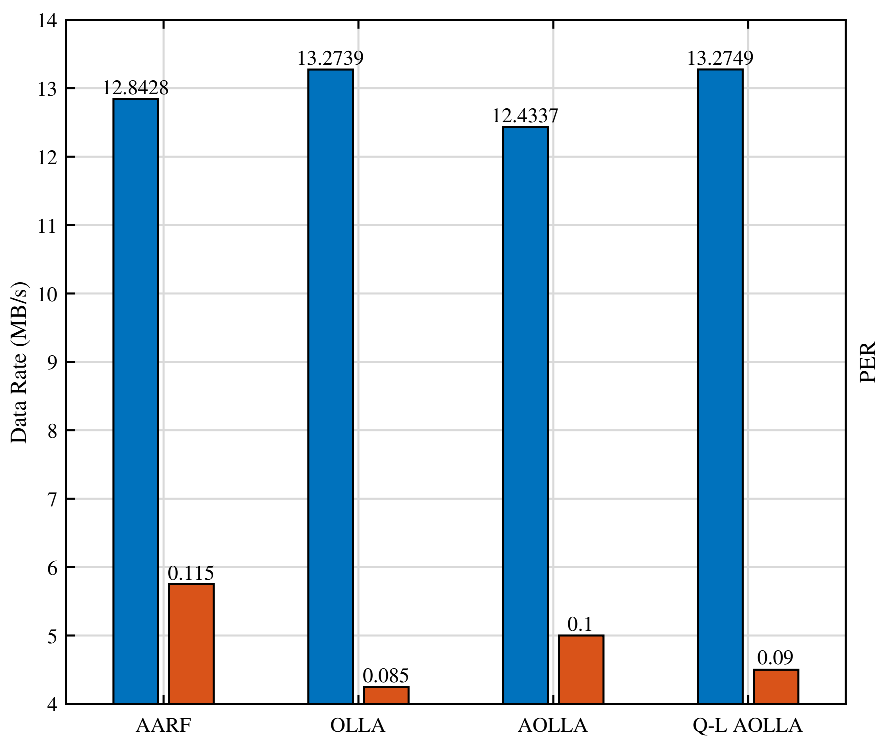

Fig. 9. Comparison of 4 link adaptive algorithms, evaluated in office environment (Q-learning training environment).

Fig. 10. Performing experiments in the hallway environment.

VI. Contact us

The authors are with the University of Electronic Science and Technology of China, Chengdu 611731, China. If you have any questions please contact us (e-mail: 2439021231@qq.com), we are looking forward to discussing with you!

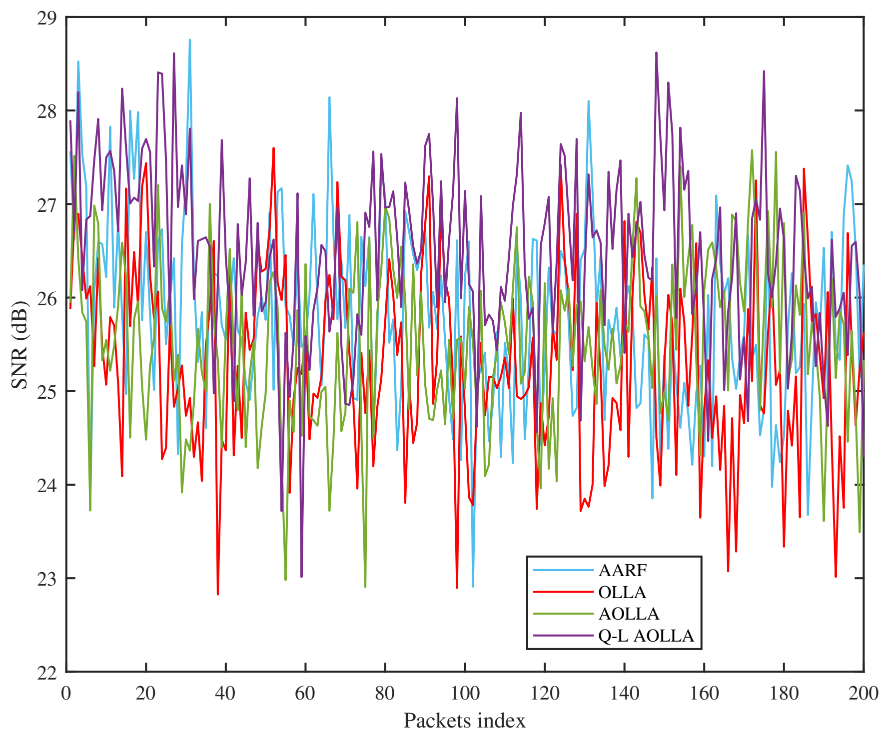

Fig. 11. SNR of 4 link adaptive algorithms, evaluated in hallway environment.

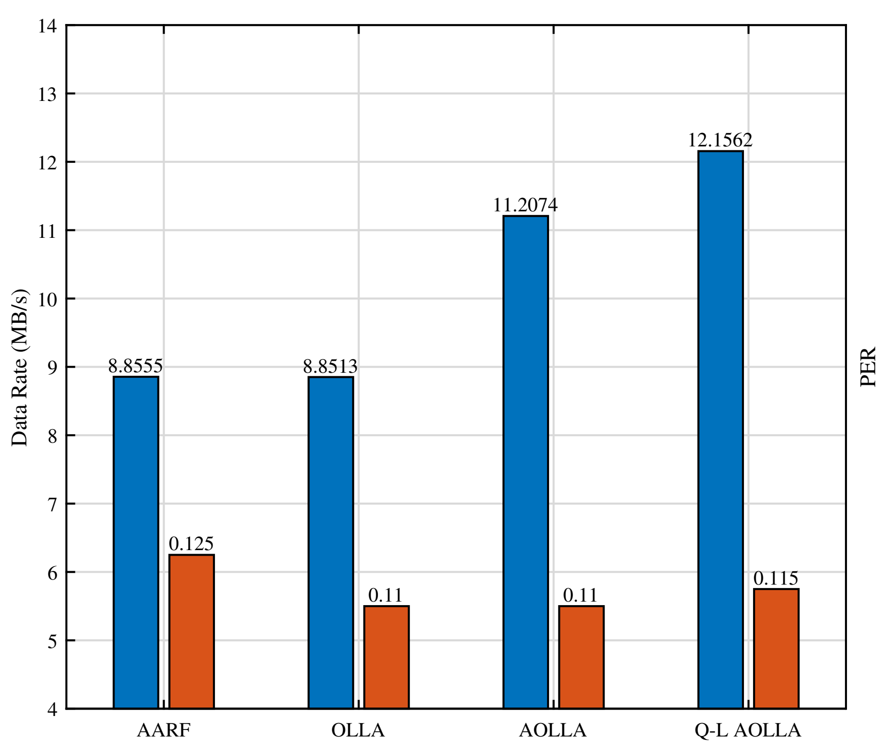

Fig. 12. Comparison of 4 link adaptive algorithms, evaluated in hallway environment.

Fig. 13. Performing experiments in the classroom environment.

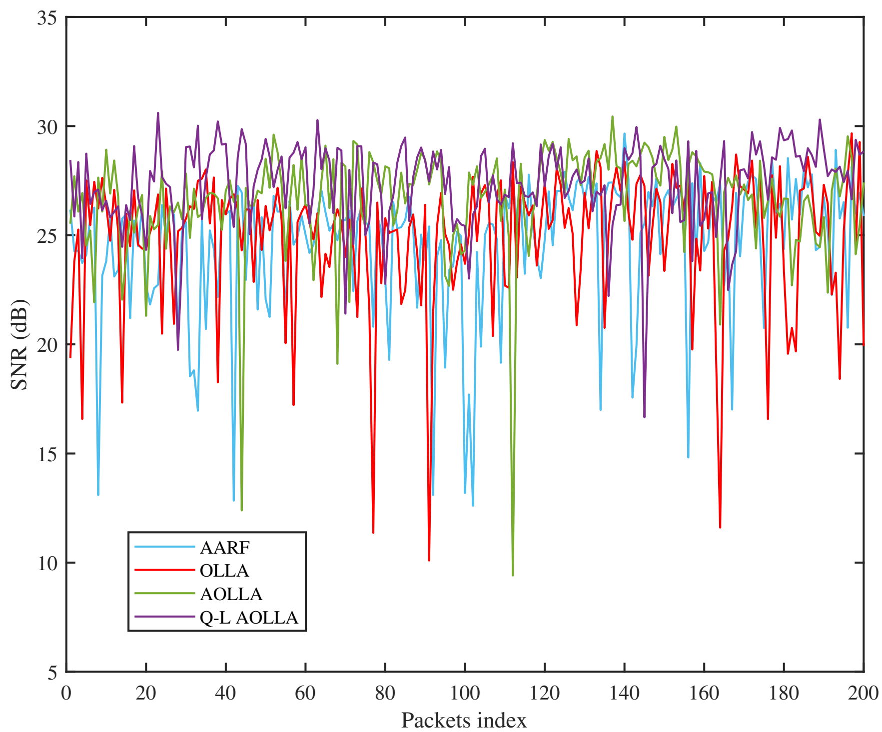

Fig. 14. SNR of 4 link adaptive algorithms, evaluated in classroom environment.

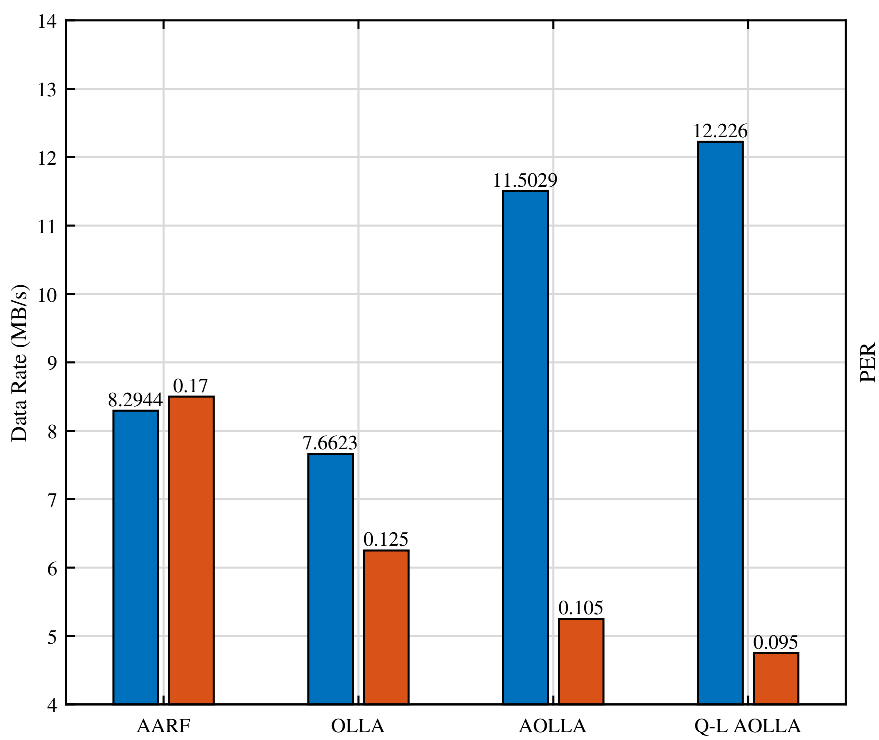

Fig. 15. Comparison of 4 link adaptive algorithms, evaluated in classroom environment.