Intelligent MU-MIMO User Selection With Dynamic Link Adaptation in IEEE 802.11ax

Raja Karmakar, *Student Member, IEEE*, Samiran Chattopadhyay, *Member, IEEE*, and Sandip Chakraborty, *Member, IEEE*

***Abstract*— IEEE 802.11ax high-throughput wireless access networks support multi-user multiple-input multiple-output (MU-MIMO)-based communication, where a set of spatially apart wireless stations forms a user group and uses different spatial streams for simultaneous transmission and reception. In this architecture, dynamic user group selection is an important aspect for maintaining high-throughput fair channel access. In addition, the physical and media access control parameters, like channel bonding levels, modulation, and coding schemes need to be tuned based on the selected user group to utilize the maximum available capacity. In this paper, we design an online learning-based approach over a centralized logical control architecture, called intelligent MU-MIMO user selection with link adaptation (IMMULA), where a central controller collects the performance statistics under various configuration space and applies a reinforcement learning strategy to select the best-suited configurations dynamically at periodic intervals. The performance of IMMULA is analyzed over a testbed consisting of 6 IEEE 802.11ac access points and 20 wireless stations. The results show that IMMULA improves network performances significantly compared to other baseline mechanisms.**

***Index Terms*— IEEE 802.11ax, MU-MIMO, block acknowledgement, link adaptation.**

I. Introduction

THE growing demands of throughput and capacity have dragged IEEE 802.11 standards towards high throughput extensions such as IEEE 802.11ac, 802.11ad, 802.11aq etc., together known as IEEE 802.11ax [1] or *High Throughput Wireless Local Area Networks (HT-WLANs)*. In IEEE 802.11ax, physical (PHY) and media access control (MAC) are enhanced with advanced technologies for link configurations, such as number of spatial streams and user modes for *multi-user multiple input multiple output (MU-MIMO) communication* [2], *channel bonding* levels and upgraded *modulation & coding schemes (MCS)* etc. MU-MIMO achieves substantial gains in the network capacity by applying precoding scheme for supporting multiple concurrent data streams to a group of wireless stations (STAs) [3]. However, the number of concurrent spatial streams is limited by the number of antennas available at the access point (AP). Based on the available number of spatial streams, a AP can select a set of STAs at periodic instances of time, so that maximum throughput benefit can be realized. The precoding scheme is used for this purpose, which consists of computing gains and phases of the transmitterâĽ™s antenna through the analysis of *channel state information (CSI)* [4]. For realizing capacity gains in MU-MIMO, the AP must choose (a) *mode:* the number of transmitting and the collective number of receiving antennas and (b) *users:* the set of antenna(s) at the STAs for beamforming. However, MU-MIMO mode and user selection may lead to the problem of unfairness, since AP-STA association is controlled by signal strength. Consider a wireless STA at the overlapping region of two APs – AP_a and AP_b. An instance may occur, where a STA in the overlapping region associates with AP_a, but gets starved to be a part of the active MU-MIMO group formed by AP_a because of insufficient number of spatial streams; whereas it has a chance to be with the active MU-MIMO user group of AP_b, in case it gets associated with that AP.

MU-MIMO user selection leads to another issue. Even with the multiple spatial streaming technology, the end-user throughput depends on the selection of various link parameters, like channel bonding and MCS levels. Based on the spatial streaming, the link parameters depend on the signal quality feedback and the interference from neighboring STAs. The optimal set of link parameters may change if the beamforming direction gets changed due to the result of MU-MIMO mode and user selection. In this paper, we address these three interrelated problems jointly – (1) dynamic AP-STA association based on MU-MIMO streaming opportunities, (2) MU-MIMO mode and user selection, and (3) dynamic link parameter adaptation based on MU-MIMO beamforming.

However, solving the aforesaid three problems simultaneously requires complex estimation of the communication environment parameters and the adaptation of the methodology based on continuous feedback from the environment. Existing methodologies [5]–[9] have studied mechanisms for MU-MIMO user selection; however, they do not consider the interdependencies and joint optimization among MU-MIMO

Manuscript received October 24, 2017; revised April 29, 2018 and September 7, 2018; accepted December 14, 2018. Date of publication January 7, 2019; date of current version February 11, 2019. The associate editor coordinating the review of this paper and approving it for publication was M. Payaró. *(Corresponding author: Raja Karmakar.)*

R. Karmakar and S. Chattopadhyay are with the Department of Information Technology, Jadavpur University, Kolkata 700098, India (e-mail: rkarmakar.tict@gmail.com; samiranc@it.jusl.ac.in).

S. Chakraborty is with the Department of Computer Science and Engineering, IIT Kharagpur, Kharagpur 721302, India (e-mail: sandipc@cse.iitkgp.ernet.in).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TWC.2018.2890219

user selection and link parameter adaptation. This joint optimization requires network wide information for optimal balancing of MU-MIMO user modes and their corresponding optimal link configuration parameters. Further, this estimation also helps in solving the AP-STA association problem to reduce unfairness due to the MU-MIMO user selection based on throughput optimization. Accordingly, every AP needs to learn the environment and the effect of channel parameters (like noise and interference) over the communication environment. As this learning requires a network wide inter-dependent information and analysis of various link parameters, we utilize the services provided by *Software Defined Networking* (SDN) architecture [10] for wireless environments, as it already gives programming interfaces for processing device data and controlling devices through a logical centralized architecture. In this architecture, the logically centralized controller develops the learning module based on the collected statistics. This learning module can be utilized by the APs (i) to select MU-MIMO user groups and the corresponding link parameters and (ii) to control the AP-STA association for improved fairness. In this paper, we design an online learning-based ***I**ntelligent **MU**-**MIMO** **U**ser selection with **L**ink **A**daptation (IMMULA)* mechanism for HT-WLANs. We design SDN-based software-controlled architecture to implement fair STA association and the learning-based module, which help APs to select MU-MIMO user groups and to perform dynamic link adaptation. It can be noted that link adaptation requires a fast decision, and therefore we also consider the scalability aspects of the controller design based on a hierarchical architecture. The performance of IMMULA is analyzed over a testbed consisting of 6 IEEE 802.11ac APs and 20 wireless STAs. The analysis shows that IMMULA improves the network performance significantly compared to other baseline mechanisms.

II. Related Work

The existing works on MU-MIMO user selection [3], [5]–[8] mostly utilize *CSI* for understanding the network environment. However, CSI estimation requires overhead intensive channel sounding process [9]. Authors in [3] designed Orthogonality Probing based User Selection (OPUS) for selecting users in MU-MIMO networks, where an AP with M antennas require M rounds of CSI feedback. At every round, a novel probing mechanism is used to facilitate each user to find out its orthogonality with other users, and finally a distributed contention based method is utilized to select the best users. In [9], a method, called Pre-sounding User and Mode selection Algorithm (PUMA), was designed for MU-MIMO mode and user selection without using CSI. PUMA estimates per-stream transmission rate and aggregate throughput in each mode for a potential set of users by utilizing the theoretical properties of MU-MIMO. In [5], authors addressed the problem of AP selection in MU-MIMO WLANs by using NULL frame probing for CSI estimation. A different approach, called Selectivity-Aware MU-MIMO (SAMU) [7], optimizes MU-MIMO selection and assigns priorities to the users from application layer. SIEVE [8] focuses on scalable user selection for implementing large-scale MU-MIMO systems. In SIEVE, a branch-and-bound method is proposed, which may create an excessive search space in a congested network scenario. Sur *et al.* [6] have proposed MU-MIMO User Selection (MUSE) along with a MU-MIMO-based rate adaptation scheme for IEEE 802.11ac networks. MUSE utilizes a novel SINR measurement for user group selection and rate adaptation jointly. However, the above works focus on MU-MIMO user group selection only without looking into its effect on link parameter adaptation. In the literature, a few works consider link parameter adaptation by considering MU-MIMO supported communication. TurboRate [11] selects only data rate for uplink MU-MIMO transmission. However, it does not consider the interplay between MU-MIMO user/mode selection and link parameter adaptation. In [12], a framework for dynamic link adaptation is proposed for MU-MIMO networks by using limited feedback information. At the receivers, the consideration of perfect CSI is a limitation, and the scheme also does not consider the interference signal. It also assumes prior knowledge of the training set but in practical scenario, the training samples should be acquired online. Our initial works like [13] and [14] falls within this group, where we have focused on dynamic link parameter adaptation based on learning based methodologies. However, during the implementation of the above works over a MU-MIMO supported network, we understood the requirements of a joint optimization for user group selection and link parameter adaptation. Both the problems are interdependent and need to be handled dynamically based on the change in the communication environment.

III. IMMULA: System Model

The proposed adaptive learning based design requires periodic monitoring and management of AP states; therefore, a distributed state management incurs a significant overhead as well as makes the execution of the learning algorithm (we call this as the learning controller) difficult. Therefore, we utilize a SDN based approach where a centralized controller periodically collects the link and channel state information and executes the adaptive learning to select MU-MIMO user groups and link states for communication. Here a link state is defined as the set of link parameters along with the corresponding parameter values. In our design approach, first we have considered the AP-STA association for implementing the fairness in the selection of MU-MIMO user group. Then, we select the best possible MU-MIMO user group and the corresponding link configuration parameter set. The two relevant modules are as follows.

(1) Fair STA Distribution (FSD): This module controls the AP-STA association based on the MU-MIMO beamforming possibility, so that unfairness due to MU-MIMO user selection can be reduced.

(2) Joint MU-MIMO Group and Link State Selection (JMMLS): Since MU-MIMO group selection is a part of link adaptation, we consider a single module for selecting MU-MIMO group and the values of different link parameters. For both of these selections (MU-MIMO group and link state), an adaptive learning mechanism is incorporated into JMMLS.

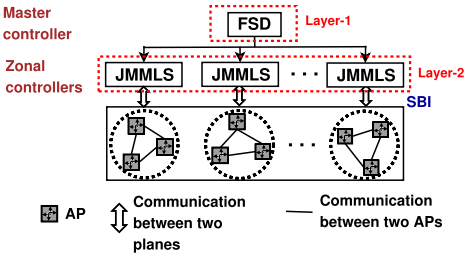

Fig. 1. SDN-based network components of IMMULA.

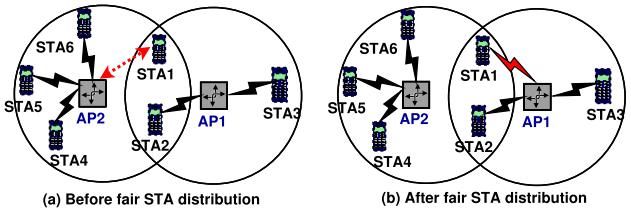

Fig. 2. (a) Unfair STA distribution and (b) Fair STA distribution.

A. SDN-Based Learning Controller

As mentioned earlier, we utilize SDN framework for designing the learning controller. However, managing the entire access network using a single controller may result in performance bottleneck; therefore, we use a hierarchical control architecture. The proposed SDN framework has two *logical* layers – layer-1 and layer-2. A *master SDN controller* is placed in layer-1, which runs the application to implement FSD. However, several *zonal SDN controllers* are implemented in layer-2. Controllers in layer-2 work after the execution of layer-1 controllers. Each zonal controller executes the JMMLS module. Since the master controller controls the overall STA association related to the network topology, it is installed in a single centralized system. Whereas, the zonal controllers have more than one instances. Hence, the zonal controllers are implemented in several systems, and they execute a part of JMMLS. Fig. 1 shows the proposed SDN architecture.

B. Implementation of Fair STA Distribution

While a STA discovers the set of APs within its communication range, it sends *association requests* to the respective target AP with whom the STA wants to connect (based on the signal strength, the standard procedure). Along with it, the STA also sends a list of APs that it discovered during the scanning. This information is shared periodically with the zonal controller by the APs; the zonal controllers in turn forward this information to the master controller. Let $\mathbb{S}_i$ and $\mathbb{S}_j$ be the set of STAs associated with AP_i and AP_j, respectively, where AP_i and AP_j have overlapping communication range. Further assume that $\mathbb{S}_{ij}$ be the set of STAs in the overlapping region of AP_i and AP_j (these STAs have sensed both the APs during the scanning). The master controller runs the following steps if $\mathbb{S}_{ij} \neq \emptyset$.

1) If $|\mathbb{S}_j| - |\mathbb{S}_i| \geq 2$, where $|\mathbb{S}|$ is the cardinality of set $\mathbb{S}$, STAs from $\mathbb{S}_{ij} \cap \mathbb{S}_j$ (the STAs which are in the overlapping region but is associated with AP_j) are re-associated with AP_i until $|\mathbb{S}_j| - |\mathbb{S}_i| \leq 1$. The master controller selects the STAs from the set $\mathbb{S}_{ij} \cap \mathbb{S}_j$ for re-association, which have sensed the maximum signal strength for AP_i compared to others.
2) If $|\mathbb{S}_i| - |\mathbb{S}_j| \geq 2$, STAs from $\mathbb{S}_{ij} \cap \mathbb{S}_i$ (the STAs which are in the overlapping region but is associated with AP_i) are re-associated with AP_j until $|\mathbb{S}_i| - |\mathbb{S}_j| \leq 1$. The master controller selects the STAs from the set $\mathbb{S}_{ij} \cap \mathbb{S}_i$ for re-association, which have sensed maximum signal strength for AP_j compared to others.

This information is forwarded to the APs by the zonal controllers. The APs send de-association request to the corresponding STAs with a suggestion to the alternate AP to connect. For example, in Fig. 2, if $STA1$ is connected with $AP1$ instead of $AP2$, equal number of STAs would be served by $AP1$ and $AP2$. It can be noted that we restrict the execution of FSD too frequently to avoid the ping-pong problem. Next, we discuss the methodology for MU-MIMO user group selection and link parameter adaptation based on the STA association with the MU-MIMO supported APs.

IV. Detailed Design of IMMULA

Let $\mathbb{M}$ be the set that represents the total number of transmit antennas and $\mathbb{K}$ be the set of collective receive antennas. Assume that each STA is equipped with a single antenna. At any time t, let us consider that $\mathbb{K}_q$ is the set of users (STAs) who want to initiate data transmission and $\mathbb{K}_q \subseteq \mathbb{K}$. Let $\mathbb{A}$ denote a MU-MIMO group and $\mathbb{A} = [\mathbb{M}, \mathbb{K}_q]$. Here, $\mathbb{A}$ denotes a set which has two parameters – the set of transmit antennas and the set of receive antennas, where the transmit antennas are transmitting data to the receive antennas by using MU-MIMO technology. Hence, $\mathbb{A}$ specifies a MU-MIMO user group. The selection of a MU-MIMO user group is based on the maximization of user throughput. In the rest of the paper, we denote the cardinalities of $\mathbb{M}$ and $\mathbb{K}$ by M and K respectively. Next, we discuss the methodology to estimate per-user throughput for MU-MIMO based transmissions.

A. Estimation of Per-User Throughput

Let R_i be the estimated throughput of user i. For any wireless transmission, R_i is generally represented as $R_i = \frac{L_{D_i}}{T_{D_i}+T_{OH_i}}$. In this case, L_{D_i} and T_{D_i} represent the total amount of data transmitted and the total data transmission time for user i respectively. Assume that L_i is the maximum packet length of user i. If a_i is the size of frame aggregation of user i, we have $L_{D_i} = a_i \times L_i$. T_{D_i} is estimated by dividing the amount of data to be transmitted by the physical bit rate. T_{OH_i} is the protocol overhead time that negatively impacts on the performance of MU-MIMO technology. T_{OH_i} is defined as $T_{OH_i} = T_{S_i} + T_{CF_i} + T_{ACK_i}$. Here, T_{S_i} and T_{CF_i} denote channel sounding time and channel feedback time of the receiver i, respectively. T_{ACK_i} is the acknowledgment time of the receiver i.

We apply channel sounding [15] and CSI feedback mechanism [4] for the estimation of user throughput based on MU-MIMO transmission model. Channel sounding [15] is a channel measurement technique that mimics the operation of

a wireless communication system to evaluate the radio environment for communication in wireless network, especially for MIMO systems. The transmitted signal is affected by channel conditions when it travels through the air. While this signal reaches the receiver, a measurement system can extract the characteristics of the wireless channel by applying signal processing technique.

For collecting the CSI feedback, an AP first transmits a frame, called *Null Data Packet Announcement* (NDPA), to notify each intended user for beamforming. After a *Short Interframe Space (SIFS)* period, a *Null Data Packet* (NDP) is sent. NDP holds a training preamble. After receiving the NDP, each user estimates its CSI and sends it to the AP. Channel feedback time is the time interval between sending the NDP or polling and receiving the CSI from the user [4].

B. MIMO Transmission Model

We consider *additive white Gaussian noise (AWGN)* channel in the use of multi-antenna system. On the basis of *singular value decomposition (SVD)*, MIMO transmission is an effective mathematical formulation to obtain the capacity of MIMO channel [16]. For M number of transmit antennas and K number of receive antennas, the multi-antenna system can be modeled as given in the following [17].

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n} \tag{1}$$

where $\mathbf{x} \in \mathbf{C}^{M\times 1}$ is the transmitted vector, $\mathbf{H} \in \mathbf{C}^{K\times M}$ is the channel matrix and $\mathbf{y} \in \mathbf{C}^{K\times 1}$ is the received vector and $\mathbf{n} \in \mathbf{C}^{K\times 1}$. $\mathbf{n}$ is a zero mean circularly symmetric complex Gaussian noise vector such that $E[\mathbf{n}\mathbf{n}^H] = \sigma^2\mathbf{I}_K$, where $\mathbf{n}^H$ denotes the *conjugate transpose (or Hermitian transpose)* of matrix $\mathbf{n}$ and $E[\mathbf{n}\mathbf{n}^H]$ is the covariance of $\mathbf{n}$ and $\mathbf{n}^H$. Here, σ^2 is AWGN noise power at each receive antenna and $\mathbf{I}_K$ is $K \times K$ identity matrix. The channel matrix $\mathbf{H}$ consists of the path gains $\mathbf{H}_{ij}$ between each pair of transmit and receive antenna. For supporting MU-MIMO beamforming, IEEE 802.11ac AP follows a sounding protocol [18]. With the help of this, the AP probes users and collects *Very High Throughput (VHT) Compressed Beamforming Feedback (CBF)* from the users [4], [19]. $\mathbf{H}$ is known at both the transmitter (AP) and the receiver (STA) ends of the communication system and it is measured from *sounding packets* [4], [19]. By applying the SVD theorem, $\mathbf{H}$ can be written as follows [17].

$$\mathbf{H} = \mathbf{U}\mathbf{D}\mathbf{V}^H \tag{2}$$

where $\mathbf{D}$ is a rectangular diagonal matrix whose diagonal elements are non-negative real numbers. $\mathbf{U}$ and $\mathbf{V}$ are $K \times K$ and $M \times M$ unitary matrices respectively. That is, $\mathbf{U}\mathbf{U}^H = \mathbf{I}_K$ and $\mathbf{V}\mathbf{V}^H = \mathbf{I}_M$, where $\mathbf{I}_M$ is $M \times M$ identity matrix. The diagonal elements of $\mathbf{D}$ are non-negative square roots of the eigenvalues of $\mathbf{H}\mathbf{H}^H$ and these elements are referred to as *singular values* of $\mathbf{H}$. The singular values denote signal gain in the channel. In MU-MIMO system with K receivers, the rank of $\mathbf{H}$ is at most K since the maximum of K receivers can be served simultaneously [17]. Let us denote the singular values of $\mathbf{H}$ by λ_i, $i = 1, 2, \ldots, K$. From Eq. (1) and Eq. (2), we can represent $\mathbf{y}$ as follows.

$$\mathbf{y} = \mathbf{U}\mathbf{D}\mathbf{V}^H\mathbf{x} + \mathbf{n} \tag{3}$$

Let us consider the following transformations. $\widetilde{\mathbf{y}} = \mathbf{U}^H\mathbf{y}$, and $\widetilde{\mathbf{n}} = \mathbf{U}^H\mathbf{n}$. At the receiver side, $\mathbf{y}$ is multiplied by $\mathbf{U}^H$ and thus, from Eq. (3), we get

$$\mathbf{U}^H\mathbf{y} = \widetilde{\mathbf{y}} = \mathbf{U}^H\mathbf{U}\mathbf{D}\mathbf{V}^H\mathbf{x} + \mathbf{U}^H\mathbf{n} = \mathbf{D}\mathbf{V}^H\mathbf{x} + \widetilde{\mathbf{n}} \tag{4}$$

The transmitter precodes its transmitted signal $\mathbf{u}$ as $\mathbf{x} = \mathbf{V}\mathbf{u}$ by following the Eigen-subspace beamforming [20]. Therefore, from Eq. (4), we can represent the received vector as $\widetilde{\mathbf{y}} = \mathbf{D}\mathbf{V}^H\mathbf{V}\mathbf{u} + \widetilde{\mathbf{n}} = \mathbf{D}\mathbf{u} + \widetilde{\mathbf{n}}$. Now, the covariance of $\widetilde{\mathbf{n}}$ and $\widetilde{\mathbf{n}}^H$ is as follows. $E[\widetilde{\mathbf{n}}\widetilde{\mathbf{n}}^H] = E[\mathbf{U}^H\mathbf{n}\mathbf{n}^H\mathbf{U}] = \mathbf{U}^H\sigma^2\mathbf{I}_K\mathbf{U} = \sigma^2$.

1) Interference Model: Let us consider that y_i is the received signal at the receiver i. From Eq. (1), when the receiver i applies the matrix $\mathbf{U}$ to the received signal, we have

$$\begin{aligned}
\mathbf{U_i}^H\mathbf{y_i} &= \mathbf{U_i}^H\mathbf{H_i}\mathbf{V}\mathbf{u} + \mathbf{U_i}^H\mathbf{n_i} \\
&= \mathbf{U_i}^H(\mathbf{U_i}\mathbf{D_i}\mathbf{V_i}^H)\mathbf{V_i}\mathbf{x_i} \\
&\quad + \mathbf{U_i}^H(\mathbf{U_i}\mathbf{D_i}\mathbf{V_i}^H)\sum_{j\neq i}\mathbf{V_j}\mathbf{x_j} + \widetilde{\mathbf{n_i}} \\
\implies \mathbf{y_i} &= \mathbf{D_i}\mathbf{x_i} + \mathbf{D_i}\sum_{j\neq i}\mathbf{V_i}^H\mathbf{V_j}\mathbf{x_j} + \widetilde{\mathbf{n_i}}
\end{aligned} \tag{5}$$

where $\mathbf{D_i}\mathbf{x_i}$ = precoded transmitted signal, $\mathbf{D_i}\sum_{j\neq i}\mathbf{V_i}^H\mathbf{V_j}\mathbf{x_j}$ = inter-user interference and $\widetilde{\mathbf{n_i}}$ = noise.

Let the transmitter's transmission power be split equally among K receive antennas and thus, $E[\mathbf{x}\mathbf{x}^H] = 1/K$. Hence, based on Eq. (5), we can find signal strength as: $E[\{\mathbf{D_i}\mathbf{x_i}\}\{\mathbf{D_i}\mathbf{x_i}\}^H] = \frac{||\mathbf{D_i}||^2}{K}$. Furthermore, inter-user interference I_i at the receiver i is given as follows.

$$\begin{aligned}
I_i &= E\big[\{\mathbf{D_i}\sum_{j\neq i}\mathbf{V_i}^H\mathbf{V_j}\mathbf{x_j}\}\{\mathbf{D_i}\sum_{j\neq i}\mathbf{V_i}^H\mathbf{V_j}\mathbf{x_j}\}^H\big] \\
&= \frac{||\mathbf{D_i}||^2}{K}\sum_{j\neq i}||\mathbf{V_i}^H\mathbf{V_j}||^2
\end{aligned} \tag{6}$$

2) Estimation of Per-User SINR: After receiving the precoded data, the individual receiver extracts its own data by applying the matrix U as reported in [6]. The receiver treats other receivers' data as interference. Let S_i be the SINR of the receiver i and N represent the noise floor, where $N = \sigma^2$. For K number of receive antennas, K parallel transmission would create K number of parallel channels, where the transmission signal power of i^{th} parallel channel can be represented as $W_i = \lambda_i^2\ P_i$ [16]. Here, P_i and λ_i are the transmitted power and signal gain in i^{th} channel, respectively. Let the total transmitted power be P. Since P is split equally among K receive antennas, $W_i = \lambda_i^2\ P/K$. The transmitter estimates the S_i of user i as follows.

$$S_i = \frac{W_i}{I_i + N} = \frac{\lambda_i^2\ P/K}{\frac{||\mathbf{D_i}||^2}{K}\sum_{j\neq i}||\mathbf{V_i}^H\mathbf{V_j}||^2 + \sigma^2} \tag{7}$$

We require matrices V and D along with noise floor N. Here, V is fed back by the user. Thus, CSI is used to compute V by applying SVD. D and N are computed as given in [6]. In our proposed approach, this CSI feedback is forwarded to

the zonal controllers which uses this information to estimate per-user throughput and execute the learning mechanism as discussed later.

3) Maximization Function for the Optimal MU-MIMO User Group: Let the channel bandwidth for i^{th} transmission be c_i. Now, as per Shannon capacity, the maximum capacity C_i of i^{th} transmission is as given in the following.

$$C_i(K) = c_i \log\Big(1 + \frac{\lambda_i^2\ P/K}{\frac{||\mathbf{D_i}||^2}{K}\sum_{j\neq i}||\mathbf{V_i}^H\mathbf{V_j}||^2 + \sigma^2}\Big) \quad (8)$$

Our objective is to find the optimal value of K, that maximizes $C_i(K)$. Now, we formally define the MU-MIMO group selection problem (selection of $\mathbb{K}$) as a maximization problem over $C_i(K)$ for $i \in [0, K-1]$ as follows.

$$\max_K\ C_i(K) \quad (9)$$

$$\text{subjected to } K \leq M \quad (10)$$

$$P_i = P/K \quad (11)$$

The above optimization indicates that the system should find out the set of users K that maximizes the capacity of each channel i. The constraints indicate that the value of K should be less than the total number of available receivers (M) and the total transmit power (P) is equally split among all the channels. To find the value of K that maximizes $C_i(K)$, we apply Lagrangian method [21] to solve the constrained optimization problem given in Eq. (9), Eq. (10) and Eq. (11). For $C_i(K)$, we define the Lagrangian, denoted by $L(C_i(K), \Lambda_1, \Lambda_2)$, for $\Lambda_1 > 0$ and $\Lambda_2 > 0$, as follows.

$$L(C_i(K), \Lambda_1, \Lambda_2) = C_i(K) + \Lambda_1(M - K) + \Lambda_2(P_i - P/K) \quad (12)$$

From Eq. (12), $L(C_i(K), \Lambda_1, \Lambda_2)$ would be maximized when

$$\frac{d\Big(L\Big(C_i(K), \Lambda_1, \Lambda_2\Big)\Big)}{dK} = 0 \quad (13)$$

From Eq. (8), we have $C_i(K) = c_i \log\Big(\frac{a+b+eK}{b+eK}\Big)$, where $a = \lambda_i^2\ P$, $b = ||\mathbf{D_i}||^2\sum_{j\neq i}||\mathbf{V_i}^H\mathbf{V_j}||^2$ and $e = \sigma^2$. Therefore, from Eq. (13), we have

$$c_i\Big[\frac{ae(b+eK)}{(a+b+eK)^3} + \frac{\Lambda_2 P}{K^2} - \Lambda_1\Big] = 0 \quad (14)$$

By solving Eq. (14), we can find out $K = K^*$ that can maximize the value of $C_i(K)$. Now, to check whether Eq. (14) provides the maximum value, we can compute the second order derivative of $L(C_i(K), \Lambda_1, \Lambda_2)$, that is $\frac{d^2\left(L\left(C_i(K),\Lambda_1,\Lambda_2\right)\right)}{dK^2}$ but it is negative. Therefore, we can conclude that there exists $K = K^*$ that maximizes the $C_i(K)$ and thus, the optimal MU-MIMO group is $\mathbb{A}^* = [\mathbb{M}, \mathbb{K}^*]$.

However, it can be noted that Eq. (14) is a non-linear and non-convex equation, and so finding a solution is non-trivial. For a given value of K^*, there may exist multiple user sets. Hence, the dynamic selection of the best suited user group from a set of available user groups is not straightforward and thus, it should be performed in an intelligent way. Therefore, we have applied an online learning mechanism (as discussed later in this paper) to adaptively find out the best possible set of MU-MIMO user group from a set of available user groups. After finding the MU-MIMO group $\mathbb{A}^*$, we need to select the best possible set of link parameters for the users in $\mathbb{A}^*$. Next, we discuss the selection of such parameters for $\mathbb{A}^*$.

C. Link State

Let us consider that c is channel bandwidth, m is value of MCS, f is size of frame aggregation and g represents guard interval. Let the link parameters (c, m, a, g) collectively create a set $P < c, m, f, g >$ and thus, P defines a *link state*. Assume that $\mathcal{C}$ defines the set of all possible combinations of such link parameters. Therefore, $P \in \mathcal{C}$. When a set of users is selected for a MU-MIMO group, a state $p \in P$ is selected separately for each user in that group because the SINR at the user side varies according to the position of the user. Let us consider that c varies from $minBand$ to $maxBand$ and m ranges from $minMcs$ to $maxMcs$. Levels of frame aggregation lie between $minLevel$ and $maxLevel$. We include n_a number of levels of a. While $n_a = 1$, we utilize the maximum number of aggregated frames ($maxLevel$) which is reduced by $step$ in successive higher values of the parameter n_a. Therefore, we have $minLevel = maxLevel - (n_a - 1) \times step$. The value of g is either $maxGi$ (800 ns) or $minGi$ (400 ns).

Let $T_i(S_i, P)$ be the calculated throughput of user i, which is observed at the link layer under SINR S_i and link parameter set P. For the transmission through channel i, we define throughput as the ratio of the amount of data transmitted successfully and the total amount of data transmitted over the channel i. As acknowledgment-based data transmission is used in IEEE 802.11ac, we can compute the number of frames transmitted successfully over a link by simply counting the number of acknowledgment frames received over that link and then from that we can calculate the value of $T_i(S_i, P)$. Furthermore, we consider that $R(P)$ is the ratio between the prescribed physical data rate for a IEEE 802.11ac transmitter as defined by the standard [18], corresponding to the link parameters P, and the actual transmitted data rate. We define the link layer performance (F_i) for the link i as given in the following.

$$F_i(S_i, P) = \alpha T_i(S_i, P) + \frac{(1-\alpha)}{R(P) - T_i(S_i, P)} \quad (15)$$

Here, the objective is to find out the link parameter set P that maximizes the value of F_i. The intuition behind the Eq. (15) is that, we want to maximize the throughput $T_i(S_i, P)$ while to minimize the difference between the maximum possible prescribed data rate $R(P)$ and the achieved throughput $T_i(S_i, P)$ with the link parameter set P, under the signal strength S_i. Here, $0 \leq \alpha \leq 1$ is a weighting factor which provides priority to the individual terms in Eq. (15).

Maximization Function for Selecting the Best Link State: From Eq. (15), for a given S_i, our task is to find out $P \in C$, which maximizes the value of $F_i(P)$. Thus, we represent the maximization function as given in the following.

$$\max_{P\in\mathcal{C}} F_i(P) \quad (16)$$

To find the value of P that maximizes $F_i(P)$, we apply the second order sufficient condition (Karush-Kuhn-Tucker conditions [22]) for a maximization problem as follows.

$$\frac{\delta F_i(P)}{\delta P} = 0 \tag{17}$$

Let $P^* \in \mathcal{C}$ be the best possible set of link parameters, which maximizes F. Therefore, from Eq. (17), we have

$$\alpha - \frac{(1-\alpha)}{(R(P^*) - T_i(P^*))^2} = 0 \tag{18}$$

The function $T_i(P)$ is a linear function of P (throughput values are approximated with a linearly separable function of channel conditions and link parameters). By solving Eq. (18), we can compute P^* that maximizes the value of $F_i(P)$. Now, to check whether Eq. (17) provides the maximum value, the second order partial derivative of $F_i(P)$ is calculated as given in the following.

$$\frac{\delta^2 F_i(P)}{\delta P^2} = -\frac{2(1-\alpha)}{(R(P) - T_i(P))^3} \tag{19}$$

Eq. (19) always provides a negative value. Hence, we can conclude that there exists $P = P^*$ that maximizes $F_i(P)$.

Now, we formally define the selection of the best possible link parameter set as a maximization problem over $F_i(P)$.

$$\max_{P} \; F_i(P) \tag{20}$$

$$\text{subjected to } P \in \mathcal{C} \tag{21}$$

Next, we discuss the stability of the solution for the above optimization.

D. Stability Analysis

We apply Lyapunov functions [23] to analyze the stability of Eq. (12) and Eq. (15) since these two equations lead to the best possible MU-MIMO group and link parameter set respectively.

1) Stability in Computing the Optimal MU-MIMO Group: Let the Lyapunov function be $V_1(x) = (1 + x^2)$, where $x = C_i(K) + \Lambda_1(M - K) + \Lambda_2(P_i - P/K)$ (from Eq. (12)). Therefore,

$$V_1'(x) = 2x\frac{dx}{dK} = 2xc_i\Big[\frac{ae(b+eK)}{(a+b+eK)^3} - \frac{(\Lambda_1\ K^2 - \Lambda_2\ P)}{K^2}\Big]$$

Now, from Lyapunov stability condition, the system is asymptotically stable if $V_1'(x) < 0$. Hence, we have the following condition, $\frac{ae(b+eK)}{(a+b+eK)^3} < \frac{(\Lambda_1\ K^2 - \Lambda_2\ P)}{K^2}$.

2) Stability in Computing the Best Suited Link Parameter Set: Let the Lyapunov function $V_2(y) = (1 + y^2)$, where $y = F_i(P)$ for a given S_i (from Eq. (15)). Thus,

$$V_2'(y) = 2y\frac{\delta y}{\delta P} = 2y\Big[\alpha - \frac{(1-\alpha)}{(R(P) - T_i(P))^2}\Big]$$

By following Lyapunov stability condition, the system will be asymptotically stable while $V_2'(y) < 0$. Therefore, for stability of the system, we have, $\frac{(1-\alpha)}{(R(P)-T_i(P))^2} > \alpha$.

E. Approximation Algorithm

The proposed maximization problem can be mapped to a *bin-packing* problem where the objective function finds the minimum number of bins. Let $\langle \mathbb{K}, P \rangle$ represent a *bin*. We do not want to reduce the number of bins as the reduction of the number of bins reduces the number of options of applying $\mathbb{K}$ and P. Higher number of bins provides more combinations of link state, the number of transmit and receive antennas. However, the large cardinalities of $\mathbb{K}$ and P increase the solver execution time of the optimization. Hence, in this context, we try to optimize the number of *bins* to obtain the maximum network performance. As bin-packing is a combinatorial *NP-Hard* problem, we use an approximate algorithm for the estimation of $\mathbb{K}$ and P. By exploiting this feature of the proposed maximization problems, we apply ϵ-greedy policy [24] to solve the optimization.

F. ϵ-Greedy Policy

ϵ-greedy [24] is a well-known policy in online reinforcement learning. A parameter ϵ, known as exploration probability, is used to control the learning rate. At a time instant t, ϵ_t is calculated as follows.

$$\epsilon_t = min(1, dE/t^2) \tag{22}$$

Here, d is a control parameter such that $d > 0$. We define E as $E = M \times K \times |P|$. Here, $|P|$ denotes the total number of link states. The policy enforcement employs two phases as follows.

Exploration: A configuration is selected randomly, and the probability of this selection is ϵ.

Exploitation: The configuration which has produced the best performance (in our case, it is the average throughput) in the past is chosen in this case. The probability of exploitation is $(1-\epsilon)$. These two approaches can be combined as a *Strategy*. Hence, a Strategy is defined as $Strategy = \epsilon \times Explore + (1-\epsilon) \times exploit$.

1) Statistic Table: We design two statistic tables – $\mathcal{H}_{MMGS}$ and $\mathcal{H}_{LSS}$, for MU-MIMO Group Selection (MMGS) and Link State Selection (LSS) respectively. Both $\mathcal{H}_{MMGS}$ and $\mathcal{H}_{LSS}$ are stored in JMMLS. After the data transmission phase of a MU-MIMO group, $\mathcal{H}_{MMGS}$ is updated. However, $\mathcal{H}_{LSS}$ is updated after the completion of the data transmission phase for the individual users in the MIMO group.

- $\mathcal{H}_{MMGS}$ stores the past information about MU-MIMO group and $\mathcal{H}_{MMGS} =< S_{avg}, \mathbb{A}, Q_{avg} >$. In this table, S_{avg} represents average SINR of all values of per-user SINR and $\mathbb{A}$ denotes MU-MIMO group. Q_{avg} specifies the average link layer throughput which has been achieved for MU-MIMO group $\mathbb{A}$ in the past.
- $\mathcal{H}_{LSS}$ contains the history about link adaptation and $\mathcal{H}_{LSS} =< S_{per-user}, P, Q_{per-user} >$. Here, $S_{per-user}$ denotes the estimated per-user SINR by using Eq. (7) and $Q_{per-user}$ is per-user average link layer throughput achieved under link state P.

G. Algorithmic Description of IMMULA

Fig. 3 illustrates the executions of several steps of the algorithm of IMMULA in several devices. Let n number of users

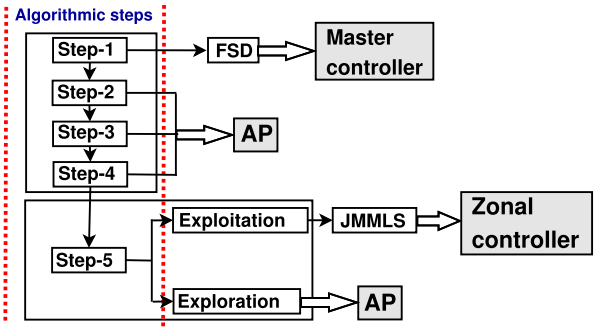

Fig. 3. Executions of different steps of IMMULA in different devices.

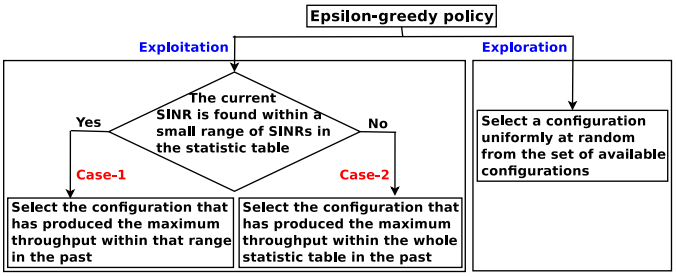

Fig. 4. Basic mechanism of online learning in IMMULA.

are waiting for initiating their data transmissions. The mechanism for FSD is performed by the master controller in Step-1. Each AP runs Step-2, Step-3, Step-4 and the exploration phase of the ϵ-greedy policy. Whereas, the exploitation phases (Step-5) of the selection of MU-MIMO group and link state are executed by the zonal controllers and then, their results are transferred to APs. The exploitation phase is executed in the zonal controllers since this phase requires some storage capacity to store the past experience. In the selected MU-MIMO group, data transmissions are carried out concurrently for the users in that group. Let a MU-MIMO group be assigned a data transmission interval of t_{dur}. Now, if there are n users in that MU-MIMO group, each user is assigned a transmission time of t_{dur}/n. We compute t_{dur} experimentally based on quick convergence of the algorithm.

Initialization: The exploration is applied with $\epsilon_t = 1$ for a random t_{init} time to gain knowledge about the network. After *initialization*, we execute following steps.

Step-1: Execute the FSD module.

Step-2: Calculate per-user SINR by using Eq. (7). Let S_i be the SINR at user i. Compute the average SINR of all values of per-user SINR. Let S_{avg} denote this average SINR, i.e., $S_{avg} = \frac{1}{n}\sum_{i=1}^{n} S_i$.

Step-3: Calculate per-user estimated throughput. For user i, it is $R_i = \frac{L_{D_i}}{T_{D_i}+T_{OH_i}}$.

Step-4: Sort users according to R_i in descending order, $1 \leq i \leq n$. Let U_n be the set of sorted users according to R_i i.e., $U_n = \{u_1, u_2, u_3, \ldots, u_n\}$. Here, the i^{th} user is specified by u_i, for $1 \leq i \leq n$.

Step-5: The basic mechanism of online learning in IMMULA is illustrated in Fig. 4 where the *configuration* refers to either MU-MIMO group or link state. Let ϵ-greedy policy be applied at time t by using Eq. (22) to select a MU-MIMO group and a link state as follows.

Exploitation: In this case, we consider two cases – (i) *case-1:* the current S_{avg} is found in the statistic tables and (ii) *case-2:* the current S_{avg} is not present in the statistic tables. The first case helps to exploit the past best knowledge that was experienced for a near value of the current S_{avg}. Whereas, the second case applies the past best experience regardless of the present SINR since the objective of the exploitation is to utilize the best configuration. Now, perform the following selections with probability $(1 - \epsilon_t)$.

Case-1: *Consideration of subsets of statistic tables.*

i) **MU-MIMO group selection:** If $S_{avg} \in [(S_{avg} - \delta), (S_{avg} + \delta)]$ (for a small value of $\delta > 0$) in $\mathcal{H}^S_{MMGS} \subset \mathcal{H}_{MMGS}$, choose MU-MIMO group $\mathbb{A}_t = [\mathbb{M}, \mathbb{K}_q]$ ($\mathbb{K}_q \subseteq \mathbb{K}_n$, $1 \leq q \leq n$) from $\mathcal{H}^S_{MMGS}$ such that $\mathbb{A}_t$ provides the highest Q_{avg} in $\mathcal{H}^S_{MMGS}$.

ii) **Link state selection:** For user i in $\mathbb{A}_t$, if $S_i \in [(S_{per-user} - \delta), (S_{per-user} + \delta)]$ in $\mathcal{H}^S_{LSS} \subset \mathcal{H}_{LSS}$, the link state $p_t \in P$ is chosen from $\mathcal{H}^S_{LSS}$. Here, p_t provides the highest $Q_{per-user}$ in $\mathcal{H}^S_{LSS}$.

Case-2: *Consideration of the whole statistic tables.*

i) **MU-MIMO group selection:** If $S_{avg} \notin [(S_{avg} - \delta), (S_{avg} + \delta)]$ in $\mathcal{H}_{MMGS}$, select the MU-MIMO group $\mathbb{A}_t = [\mathbb{M}, \mathbb{K}_q]$ ($\mathbb{K}_q \subseteq \mathbb{K}_n$, $1 \leq q \leq n$), that produces the highest Q_{avg} in $\mathcal{H}_{MMGS}$.

ii) **Link state selection:** For user i in $\mathbb{A}_t$, if $S_i \notin [(S_{per-user} - \delta), (S_{per-user} + \delta)]$ in $\mathcal{H}_{LSS}$, choose the link state $p_t \in P$ from $\mathcal{H}_{LSS}$, that provides the highest $Q_{per-user}$ in $\mathcal{H}_{LSS}$.

Exploration: Perform this step with probability ϵ_t. Select a MU-MIMO group $\mathbb{A}_t = [\mathbb{M}, \mathbb{K}_r]$ uniformly at random such that $\mathbb{K}_r \subseteq \mathbb{K}_n$ and $1 \leq r \leq n$. Also, a link state p_t is selected at random in P.

After selecting the MU-MIMO group $\mathbb{A}_t$, the first q users (in case of exploitation) or the first r users (in case of exploration) are chosen from U_n. Let us consider that U_q (or U_r) denotes the set of users for MU-MIMO group $\mathbb{A}_t$. Before transmission begins for an user, link state p_t is computed.

Theorem 1: IMMULA has a linear time bound of $O(n \log(|\mathcal{H}_{LSS}|))$, where $|\mathcal{H}_{LSS}|$ is the number of entries in $\mathcal{H}_{LSS}$, at any time t. Here, n denotes the number of users in the selected MU-MIMO group.

Proof: At time t, let $|\mathcal{H}_{MMGS}|$ and $|\mathcal{H}_{LSS}|$ represent the number of entries in $\mathcal{H}_{MMGS}$ and $\mathcal{H}_{LSS}$ respectively. Assume that the statistic tables are updated in sorted way on the basis of SINR and we apply binary search in statistic tables. The average time complexity of the searching in $\mathcal{H}_{MMGS}$ is $O(\log(|\mathcal{H}_{MMGS}|))$. For a MU-MIMO group, JMMLS performs link adaptation for the individual users. Hence, at t, if A_t has n number of users, the average time complexity of the searching in $\mathcal{H}_{LSS}$ is $O(n \log(|\mathcal{H}_{LSS}|))$. Thus, the total time bound $= O(\log(|\mathcal{H}_{MMGS}|) + n \log(|\mathcal{H}_{LSS}|))$. Now, $|\mathcal{H}_{LSS}| > |\mathcal{H}_{MMGS}|$ since $|\mathcal{H}_{LSS}|$ is updated after

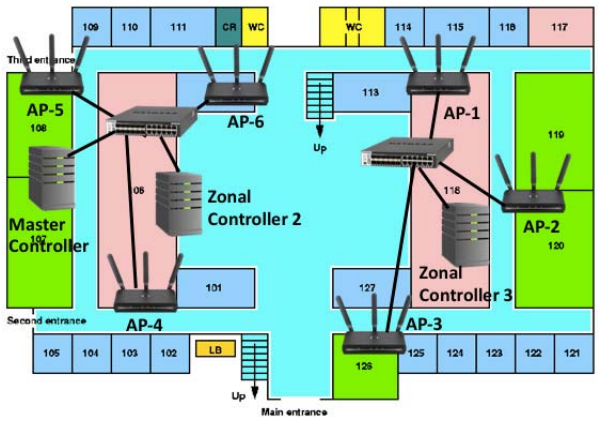

Fig. 5. Testbed connectivity layout (AP-1, AP-2 and AP-3 belongs to one zone, and remaining three APs belong to the second zone).

the data transmission of each user. Therefore, we have Time bound $= O(n \log(|\mathcal{H}_{LSS}|))$. ■

V. Performance Analysis From Testbed

We have implemented IMMULA over a IEEE 802.11ac wireless networking testbed with 6 APs and 20 wireless stations. The details of the testbed setup and performance analysis are discussed next.

A. Testbed Setup

In the testbed, we have used Marvell Avastar 88W8964 chipset, a commodity AP board that uses 4×4 MIMO with 3 spatial streams for MU-MIMO. The board uses 5 GHz radio with a maximum of 160 MHz channel bandwidth and 256-QAM modulation level. However, we have used up to 80 MHz channel bandwidth based on the compatibility with the client boards. The routers have been flushed with Unix based open source OpenWRT/LEDE kernel[1] along with open source chipset driver `mwlwifi`.[2] This chipset supports `softMAC` architecture, and therefore, the link adaptation and MU-MIMO group selection algorithms have been implemented as a part of the `mac80211` module of the device kernel. The configuration parameters for 802.11ac AP, such as channel bandwidth and modulation level, have been exported to the transport layer, so that the parameters can be configured using SDN `OpenFlow` protocol. At the client side, we use ASUS USB-AC56 IEEE 802.11ac dongles that use Realtek RTL8812AU chipset for supporting IEEE 802.11ac in STA mode. Here, we have configured the chipset with open source `rtl8812au`[3] driver.

The testbed connectivity layout and the AP coverage over the testbed are shown in Fig. 5. The signal strength from the APs has been measured at different time of the day, and the average signal strength is considered for roughly estimating the coverage circle. We have considered that a STA can get associated with the AP if the signal strength is less than -70 dBm. With this, the largest coverage radius (for AP-2) is $55m$, and the smallest coverage area (for AP-3) is $46m$ (the coverage area varies because of the indoor environment and nearby devices that create signal attenuation). In our testbed, we have implemented the SDN controller modules (FSD, JMMLS) using `Ryu` controller that uses `OpenFlow` to forward the configurations to the APs. Based on the configurations received, we have written a few kernel scripts in the device firmware. Free-moving users at the department worked as volunteers who have generated data from various web based applications (mainly video streaming). Consequently, majority of the traffic in the network is TCP traffic, although a small amount of traffic (around 10%) originates from UDP applications (like background syncing, interactive applications like Google hangout). In our testbed, every experiment is executed continuously for 3 hours, and repeated on 10 different days. The average performance metrics are then used to analyze the performance of the proposed architecture. We evaluate the efficiency of IMMULA with respect to MUSE [6], SAMU [7] and SIEVE [8]. These baselines have been implemented as a part of the testbed based on the respective paper guidelines. However, these approaches primarily consider MU-MIMO user group selection, and so, the link parameter adaptation mechanisms have been executed independently based on Minstrel-HT (the standard link parameter adaptation for Unix based systems) unless the method has an embedded link adaptation mechanism.

[1]https://openwrt.org/ (last accessed: March, 2018)
[2]https://github.com/kaloz/mwlwifi (last accessed: March, 2018)
[3]https://github.com/gnab/rtl8812au (last accessed: March, 2018)

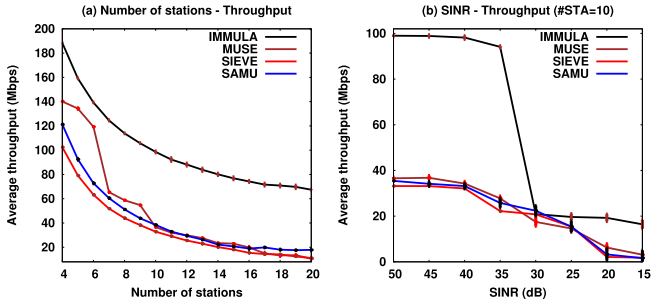

Fig. 6. Performance in terms of throughput.

B. Analysis of Throughput

Fig. 6 shows the performance gain of IMMULA compared to baselines, in terms of average link layer throughput. From Fig. 6(a), we observe that IMMULA has a throughput gains of 539.06%, 277.94% and 506.00% compared to MUSE, SAMU and SIEVE, respectively (with 20 STAs). MUSE, SAMU and SIEVE primarily focus on MU-MIMO user group selection. SAMU and SIEVE do not consider link parameter adaptation explicitly based on the selected MU-MIMO user groups. Fig. 6(b) shows that the throughput gain is better than the baselines even when the SINR is less than 30 dB. The throughput gain of IMMULA under dynamic network structures can be justified as follows. As the number of STAs increases, the noise level due to interference also increases. When exploration is applied in a congested network scenario, the system gathers information regarding link layer throughput of several link configuration parameter sets under

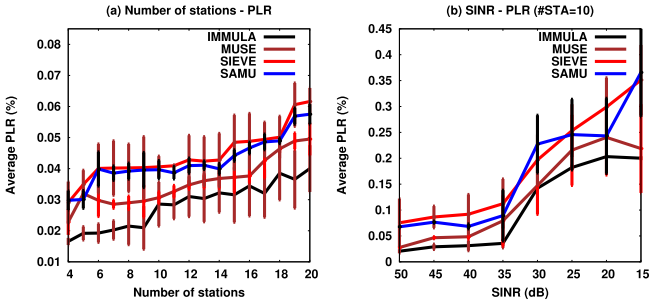

Fig. 7. Performance in terms of PLR.

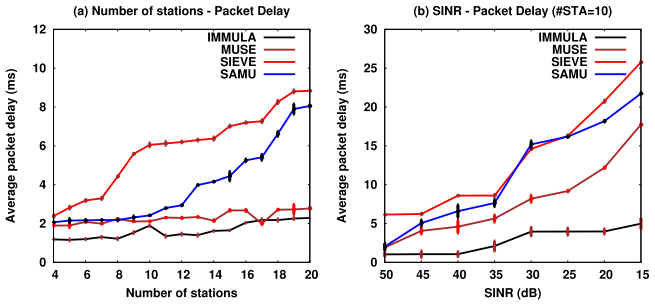

Fig. 8. Performance in terms of packet delay.

different number of wireless stations. This information serve as an experience for the network in the future, where the previous information is used in the exploitation phase to apply the past experience along with the exploration of new configurations. With a complete new setup where the past experience (exploitation) is not useful, the learning explores new configurations uniformly randomly towards the direction of the performance maximization curvature (given by Eq. 15) based of the received feedback. In this way, the system takes decision adaptively according to the node density of the network. The impact of learning convergence under a dynamic network scenario is discussed later in this section.

C. Analysis of PLR and Packet Delay

The optimal selection of MU-MIMO user groups and corresponding link configurations in IMMULA leads to lower packet loss and packet delay than the other schemes as shown in Fig. 7 and Fig. 8 respectively. For example, IMMULA has 19.18% less average PLR than that of MUSE, whereas this reduction is 30.42% less in SAMU (for 20 STAs in Fig. 7(a)). In case of average packet delay, IMMULA incurs 17.65%, 86.69% and 74.12% reductions in average packet delay compared to MUSE, SAMU and SIEVE respectively (for 20 STAs in Fig. 8(a)). The application of both exploitation and exploration helps to impose a balance between the unexplored and the best options for MU-MIMO user group and link state. As time goes by, the exploitation increases. The VHT CBF also helps to measure signal quality at different users. With the help of VHT CBF, the system is able to select the user group and link state as per the density of the network and signal level. An optimal selection of MU-MIMO user group and the link configuration parameters helps in reducing inter-node interference in the network. Consequently, we observe lower packet loss and packet delay in IMMULA

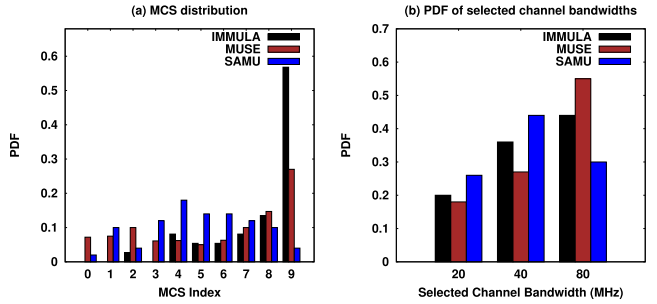

Fig. 9. Analysis of IMMULA in selecting parameters: (a) PHY rate distribution; (b) channel bandwidths.

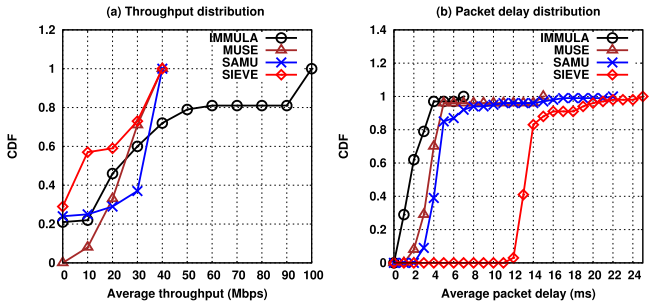

Fig. 10. Analysis of IMMULA: (a) average throughput distribution; (b) average packet delay distribution.

compared to other baselines as shown in Fig. 7 and Fig. 8 respectively.

D. Selection of Parameters

According to the channel condition and the network scenario (the number of users), the selection of the appropriate MU-MIMO user group and the link state are the key factors to achieve high throughput. We compute probability density functions (PDFs) of different link configuration parameters as shown in Fig. 9 (for 10 STAs). We impose dynamic channel condition. Since SIEVE does not deal with channel bandwidth and different MCS levels, we exclude it in the analysis of these two parameters. In the exploitation, the maximum values of link parameters are set according to the channel condition. Consequently, as the number of run increases, the probability of selecting higher values of link parameters enhances. Thus, in IMMULA, the probabilities of using MCS 9 and channel bandwidth of 80 MHz are 57% and 44% respectively. IMMULA selects these link configuration parameters dynamically based on the selected MU-MIMO group and also maximizes the overall system performance.

E. Analysis of Throughput and Packet Delay Distribution

Fig. 10 shows the cumulative density functions (CDFs) of average throughput and packet delay distributions (for 10 STAs and dynamic channel condition). As the average throughput is significantly high in IMMULA compared to other schemes (Fig. 6), CDFs of MUSE, SAMU and SIEVE converge within 40 Mbps and these distributions are high in the range of 10 Mbps – 30 Mbps. On the other hand, the density of such distribution in IMMULA spans through 40 Mbps – 100 Mbps.

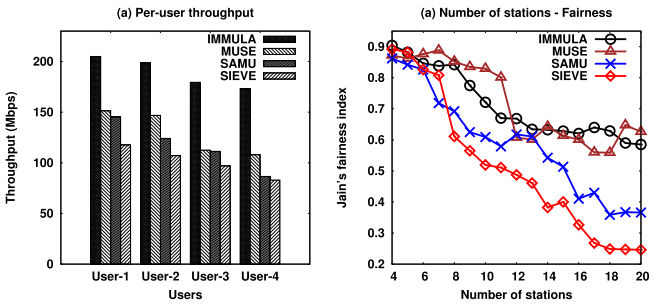

Fig. 11. Analysis of IMMULA: (a) per-user throughput performance; (b) Fairness comparison.

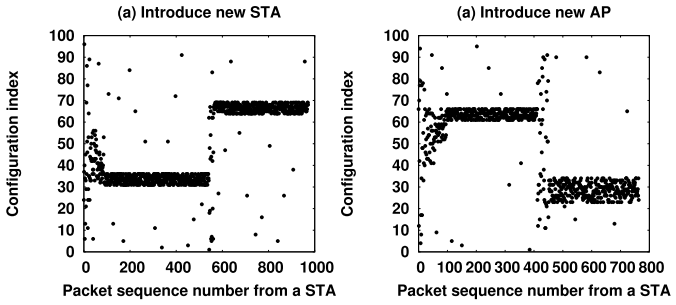

Fig. 12. Packet wise selected Configuration Index.

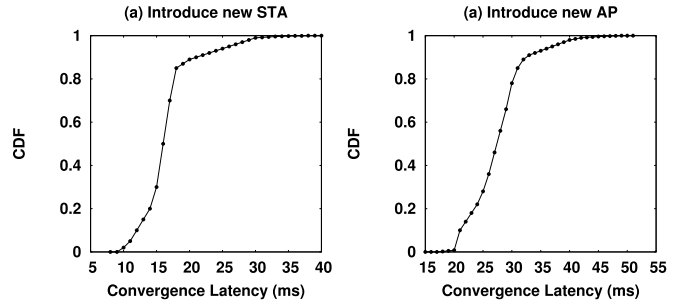

Fig. 13. Distribution of Convergence Latency.

In IMMULA, average packet delay distribution is also very low due to the overall low average packet delay (Fig. 8).

F. Per-User Throughput and Fairness

We evaluate per-user throughput considering 4 users and dynamic channel condition (Fig. 11(a)). The distance between the AP and the users (STAs) increases randomly as the indices of the users increase. The online learning helps to create the group with the maximum number of possible users. For example, User-1 is the closest STA of the AP. The online learning helps to create the group with the maximum number of possible users. By applying CSI, signal quality at each user is identified, and the user group is formed accordingly. The online learning helps to create the group with the maximum number of possible users such that the transmitter antenna can provide the maximum throughput gain per user. This approach leads to the best possible throughput for each user in IMMULA. The analysis of Jain's fairness index [25] of throughput is shown in Fig. 11(b). In a congested environment, after some runs, IMMULA finds the link parameter sets which have produced the best throughput so far. Additionally, the searching for the maximum size of the MU-MIMO group can consider as much number of users as possible. Based on MU-MIMO beamforming, the control of AP-STA association in FSD module also helps to impose a fairness in the network.

G. Convergence Analysis

We have done a set of controlled experiments to check the convergence of the proposed learning mechanism. The AP in the testbed supports a total of 96 configurations (combinations of 3 spatial streams, 4 channel bonding levels and 8 MCS levels). Each of these configurations has been termed as a configuration index and has been numbered from 1 to 96. We have collected the log for per packet link configuration index from one STA and have shown that in Fig. 12 for two different cases. The objectives of Fig. 12 are to show the convergence behavior and to discuss the methodology for the computation of convergence latency. In Fig. 12(a), we have first connected three STAs with the AP, and then after some time, connected three more STAs to the same AP. The figure shows the change in configuration index due to the joining of new STAs in the system. Similarly, in Fig. 12(b), we have connected six STAs with a single AP; then, after some time, we have introduced another AP with the same Basic Service Set Identifier (BSSID) under the same coverage area, so that the STAs get distributed among the two APs. We have following observations from Fig. 12. (a) Initially, the configuration indexes fluctuate and then converge among a small set of configurations (this small set corresponds to the minor change or fluctuation in channel quality). (b) Whenever we introduce new STAs or a new AP in the system, the configuration indexes fluctuate again (the learning explores new configurations) and the system converges to a new set of configurations. The time it takes for stabilizing the configurations is termed as the configuration latency. We also observe some outliers in the configuration indexes even when the system is stable – this corresponds to the *exploration* phase of the proposed learning mechanism that searches for possible better configurations outside the already learned values (exploited values).

We next repeat this experiment by dynamically introducing (as well as removing) STAs and APs in (from) the testbed and compute the convergence latency for each cases. We plot the distribution of convergence latency for introducing (removing) STAs (Fig. 13(a)) and introducing (removing) APs (Fig. 13(b)). We observe that the convergence latency is very low for almost all the cases, which indicates the robustness of the proposed learning algorithm used in IMMULA. The convergence is fast even in the dynamic scenario, because a change in the network structure (addition or removal of network devices) ultimately impacts the channel condition and MU-MIMO user grouping. The learning exploits the past experience, and if it has observed a similar scenario in the past, it converges very quickly. Otherwise, the learning explores new configurations and converges as it observes an optimal value. We observe that when a complete new and very different network structure is introduced in the system, the convergence time is little high (more than 30ms), but for other cases (more than 80% of the cases) the convergence time is very low.

VI. Conclusion

This paper proposes an adaptive learning based approach for joint MU-MIMO user group and dynamic link configuration parameter selection for IEEE 802.11ax based networks. The proposed system, IMMULA, exploits a SDN based approach to build up a centralized and low-overhead learning model by observing the best configuration parameters depending on the compressed CSI feedback. Further, IMMULA reduces channel access unfairness by distributing STAs among the APs considering link configuration parameter values along with MU-MIMO beamforming capabilities. The performance analysis of IMMULA over a practical IEEE 802.11ac testbed shows that the proposed learning model is robust to significantly improve the system throughput and fairness compared to the baselines, while providing a quick convergence under dynamic network environment. As a next phase of this work, we plan to extend the proposed model under mobile network environment when rapid fluctuations of channel quality play a key role in deciding the configurations.

References

[1] B. Bellalta, "IEEE 802.11ax: High-efficiency WLANs," *IEEE Wireless Commun. Mag.*, vol. 23, no. 1, pp. 38–46, Feb. 2016.

[2] M. X. Gong, B. Hart, and S. Mao, "Advanced wireless LAN technologies: IEEE 802.11ac and beyond," *ACM GetMobile, Mobile Comput. Commun.*, vol. 18, no. 4, pp. 48–52, Oct. 2014.

[3] X. Xie and X. Zhang, "Scalable user selection for MU-MIMO networks," in *Proc. IEEE Int. Conf. Comput. Commun.*, Apr./May 2014, pp. 808–816.

[4] O. Bejarano, E. W. Knightly, and M. Park, "IEEE 802.11ac: From channelization to multi-user MIMO," *IEEE Commun. Mag.*, vol. 51, no. 10, pp. 84–90, Oct. 2013.

[5] Y. Zeng, I. Pefkianakis, K.-H. Kim, and P. Mohapatra, "MU-MIMO-aware AP selection for 802.11ac networks," in *Proc. ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, Jul. 2017, p. 19.

[6] S. Sur, I. Pefkianakis, X. Zhang, and K.-H. Kim, "Practical MU-MIMO user selection on 802.11ac commodity networks," in *Proc. 22nd Annu. ACM Conf. Mobile Comput. Netw.*, Oct. 2016, pp. 122–134.

[7] Y. Du, E. Aryafar, P. Cui, J. Camp, and M. Chiang, "SAMU: Design and implementation of selectivity-aware MU-MIMO for wideband WiFi," in *Proc. 12th Annu. IEEE Int. Conf. Sens., Commun., Netw.*, Jun. 2015, pp. 229–237.

[8] W.-L. Shen, K. C.-J. Lin, M.-S. Chen, and K. Tan, "SIEVE: Scalable user grouping for large MU-MIMO systems," in *Proc. IEEE Int. Conf. Comput. Commun.*, Apr. 2015, pp. 1975–1983.

[9] N. Anand, J. Lee, S.-J. Lee, and E. W. Knightly, "Mode and user selection for multi-user MIMO WLANs without CSI," in *Proc. IEEE Int. Conf. Comput. Commun.*, Apr. 2015, pp. 451–459.

[10] N. A. Jagadeesan and B. Krishnamachari, "Software-defined networking paradigms in wireless networks: A survey," *ACM Comput. Surv.*, vol. 47, no. 2, pp. 27:1–27:11, Nov. 2014.

[11] W.-L. Shen *et al.*, "Rate adaptation for 802.11 multiuser MIMO networks," in *Proc. 18th Annu. Int. Conf. Mobile Comput. Netw.*, Aug. 2012, pp. 29–40.

[12] A. Rico-Alvarino and R. W. Heath, Jr., "Learning-based adaptive transmission for limited feedback multiuser MIMO-OFDM," *IEEE Trans. Wireless Commun.*, vol. 13, no. 7, pp. 3806–3820, Jul. 2014.

[13] R. Karmakar, S. Chattopadhyay, and S. Chakraborty, "Dynamic link adaptation for high throughput wireless access networks," in *Proc. IEEE Int. Conf. Adv. Netw. Telecommun. Syst.*, Dec. 2015, pp. 1–6.

[14] R. Karmakar, S. Chattopadhyay, and S. Chakraborty, "SmartLA: Reinforcement learning-based link adaptation for high throughput wireless access networks," *Comput. Commun.*, vol. 110, pp. 1–25, Sep. 2017.

[15] M. Kim, J. I. Takada, and Y. Konishi, "Novel scalable MIMO channel sounding technique and measurement accuracy evaluation with transceiver impairments," *IEEE Trans. Instrum. Meas.*, vol. 61, no. 12, pp. 3185–3197, Dec. 2012.

[16] S. K. Jayaweera and H. V. Poor, "Capacity of multiple-antenna systems with both receiver and transmitter channel state information," *IEEE Trans. Inf. Theory*, vol. 49, no. 10, pp. 2697–2709, Oct. 2003.

[17] İ. E. Telatar, "Capacity of multi-antenna Gaussian channels," *Eur. Trans. Telecommun.*, vol. 10, no. 6, pp. 585–595, 1999.

[18] *IEEE Standard for Information Technology—Local and Metropolitan Area Networks—Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 4: Enhancements for Very High Throughput for Operation in Bands Below 6 GHz*, IEEE Standard 802.11ac-2013, 2013, pp. 1–425.

[19] H. Lou, M. Ghosh, P. Xia, and R. Olesen, "A comparison of implicit and explicit channel feedback methods for MU-MIMO WLAN systems," in *Proc. IEEE 24th Int. Symp. Pers., Indoor Mobile Radio Commun.*, Sep. 2013, pp. 419–424.

[20] P. Wang and L. Ping, "On maximum eigenmode beamforming and multi-user gain," *IEEE Trans. Inf. Theory*, vol. 57, no. 7, pp. 4170–4186, Jul. 2011.

[21] T. Okamoto and H. Hirata, "Constrained optimization using the Lagrangian method and the improved discrete gradient chaos model," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2009, pp. 3909–3916.

[22] G. Gordon and R. Tibshirani, "Karush-Kuhn-tucker conditions," *Optimization*, vol. 10, no. 36, p. 725, 2012.

[23] A. M. Lyapunov, *Stability of Motion*, vol. 30. New York, NY, USA: Academic, 1966.

[24] C. J. C. H. Watkins, "Learning from delayed rewards," Ph.D. dissertation, Univ. Cambridge, Cambridge, U.K., May 1989.

[25] H. Shi, R. V. Prasad, E. Onur, and I. G. M. M. Niemegeers, "Fairness in wireless networks: Issues, measures and challenges," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 5–24, 1st Quart., 2014.

Raja Karmakar received the B.Tech. degree in computer science and engineering from the Government College of Engineering and Leather Technology, Kolkata, India, and the M.Eng. degree in software engineering from Jadavpur University, Kolkata, where he is currently pursuing the Ph.D. degree. He is currently an Assistant Professor with the Department of Information Technology, Techno International Newtown, Kolkata. He is a Student Member of the IEEE.

His research areas include wireless networks, specifically high-throughput wireless access networks and mobile computing.

Samiran Chattopadhyay received the B.Tech. and M.Tech. degrees from IIT Kharagpur, in 1987 and 1989, respectively, and the Ph.D. degree from Jadavpur University, in 1993. He is currently a Professor with the Department of Information Technology, Jadavpur University. He received the Gold Medal from IIT Kharagpur, for being first in the institute. He has two decades of experience in serving reputed industry houses such as Computer Associates, Interra Systems India, Agilent, and Motorola, as a Technical Consultant.

Prof. Chattopadhyay focuses on algorithms for security, bio informatics, distributed and mobile computing, and middleware.

Sandip Chakraborty received the B.Eng. degree from Jadavpur University, Kolkata, India, and the M.Tech. and Ph.D. degrees from IIT Guwahati, India. He is currently an Assistant Professor with IIT Kharagpur, India.

His research areas include computer systems, and networks and distributed computing.