Multi-Armed Bandit Load Balancing User Association in 5G Cellular HetNets

Alireza Alizadeh and Mai Vu
Department of Electrical and Computer Engineering, Tufts University, Medford, USA
Email: {alireza.alizadeh, mai.vu}@tufts.edu

***Abstract*—Using a reinforcement learning multi-armed bandit (MAB) technique, we design a centralized and a semi-distributed online algorithms, for performing load balancing user association in multi-tier heterogeneous cellular networks. The proposed algorithms guarantee user association solutions that satisfy load balancing constraints among the base stations (BSs) by employing a central load balancer (CLB). At each time step, these algorithms provide real-time associations which give the best-to-date network spectral efficiency. In the centralized approach, the CLB performs base station assignments which determine the action for each user equipment (UE) to update its reward. In the semi-distributed approach, each UE proposes an association action based on its local information and communicates with the BS for an associated reward. Numerical results show that the proposed MAB-based algorithms exhibit fast convergence and reach closely a near-optimal benchmark centralized solution.**

I. Introduction

Future wireless networks are becoming denser with coexistence of various types of base stations (BSs) operating at different frequency bands. A challenging problem in these dense networks is load balancing user association: finding the best connections between BSs and user equipment (UEs) to achieve an optimal network performance while balancing the BSs' loads (number of UEs served by each BS). This problem is more challenging in mmWave networks because of high directionality and user associations significantly alter the network interference structure [1].

In recent years, machine learning techniques have found interesting applications for user association in cellular networks [2]–[4]. A common feature of these existing works, however, is that the algorithms require a training process which usually takes a significant amount of time, rendering these approaches inapplicable to highly-dynamic dense mmWave-enabled HetNets. Furthermore, these learning algorithms do not explicitly specify how they satisfy the load balancing constraint formulated on each BS.

Multi-armed bandit (MAB) is a reinforcement learning technique in which agents explore and exploit different actions and receive rewards in order to find their best possible actions [5]. In the context of user association, a UE can be considered as an agent and selecting a BS can be considered as taking an action. Thus, user association problem can be caste as a multi-agent MAB. MAB techniques have been employed for user association, but without load balancing, where each UE tends to connect to the BS providing the highest reward, regardless of load constraints [6].

In this paper, we employ MAB techniques and propose online centralized and semi-distributed algorithms for load balancing user association. Load-balancing conditions are enforced by the novel central load balancer (CLB), which has the rewards information of all users and uses it to assign BS connections to satisfy load constraints and achieve a high total reward. The centralized and semi-distributed versions differ in where UE actions are proposed, but both algorithms use local measurements at UEs and do not require full channel state information. The proposed algorithms achieve performance close to that of a benchmark near-optimal centralized solution, and allow online implementation with fast convergence, making them suitable for highly-dynamic mmWave-enabled cellular networks.

II. System Model

We study the problem of user association in a multi-tier HetNet with MCBSs operating at a microwave (sub-6 GHz) band and SCBSs working at a mmWave band. In this section, we introduce the network, channel, and signal models.

A. Network and Channel Models

We consider the downlink of a two-tier cellular HetNet with B macro cell BSs (MCBSs), S small cell BSs (SCBSs), and K UEs. Let $\mathcal{B}$, $\mathcal{S}$, and $\mathcal{J} = \{1, \ldots, J\}$ denote the respective sets of MCBSs, SCBSs, and all BSs with $J = B + S$, and $\mathcal{K} = \{1, \ldots, K\}$ represents the set of UEs. Each BS j has a uniform planar array (UPA) antenna with M_j elements, each UE k is equipped with a single antenna at sub-6 GHz band and a uniform linear array (ULA) antenna with N_k elements at mmWave band. Each UE k requests n_k data streams from its serving

The work in this paper is supported in part by the National Science Foundation under CNS Grants 1908552.

BS such that $1 \leq n_k \leq N_k$, where the upper bound is due to the number of each UE's antennas.

In the sub-6 GHz band, the transmissions are omnidirectional and we use the well-known Gaussian MIMO channel model. Denote $\mathbf{h}_{k,j}^{\mu\mathrm{W}} \in \mathbb{C}^{M_j}$ as the channel vector between MCBS j and UE k where the entries are i.i.d. complex Gaussian random variables given by $h^{\mu\mathrm{W}} \sim \mathcal{CN}(0,1)$. In the mmWave band, the transmissions are highly directional and the simple Gaussian MIMO channel may not hold. Instead, we employ the clustered mmWave MIMO channel model used in [1].

B. Signal Model

For tier-1 working at sub-6 GHz band, the effective interfering channel on UE k from MCBS $j \in \mathcal{B}$ serving UE l is defined as

$$h_{k,l,j} = \mathbf{h}_{k,j}^{\mu\mathrm{W}} \mathbf{f}_{l,j} \tag{1}$$

where $\mathbf{f}_{l,j} \in \mathbb{C}^{M_j \times 1}$ is the linear precoder (transmit beamforming vector) at MCBS j intended for UE l. If $l = k$, this defines the effective channel between MCBS j and UE k as $h_{k,j} = \mathbf{h}_{k,j}^{\mu\mathrm{W}} \mathbf{f}_{k,j}$.

Similarly, for tier-2 operating at mmWave band, the effective interfering channel on UE k from SCBS $j \in \mathcal{S}$ serving UE l is defined as

$$\mathbf{H}_{k,l,j} = \mathbf{W}_k^* \mathbf{H}_{k,j}^{\mathrm{mmW}} \mathbf{F}_{l,j} \tag{2}$$

where $\mathbf{F}_{l,j} \in \mathbb{C}^{M_j \times n_l}$ is the linear precoder at SCBS j intended for UE l, and $\mathbf{W}_k \in \mathbb{C}^{N_k \times n_k}$ is the linear combiner (receive beamforming matrix) of UE k. If $l = k$, (2) becomes the effective channel between SCBS $j \in \mathcal{S}$ and UE k which includes both beamforming vectors/matrices at the BS and UE, and can be expressed as $\mathbf{H}_{k,j} = \mathbf{W}_k^* \mathbf{H}_{k,j}^{\mathrm{mmW}} \mathbf{F}_{k,j}$.

Thus, the received signals at UE k connected to MCBS $j \in \mathcal{B}$ can be written as

$$y_k^{\mu\mathrm{W}} = \sum_{j \in \mathcal{B}} h_{k,j} s_{k,j} + z_k \tag{3}$$

where $s_{k,j} \in \mathbb{C}$ is the data symbol intended for UE k with $\mathbb{E}[s_{k,j}^* s_{k,j}] = P_{k,j}$, and $z_k \in \mathbb{C}$ is the complex additive white Gaussian noise at UE k with $z_k \sim \mathcal{CN}(0, N_0)$, and N_0 is the noise power.

Similarly, the received signals at UE k connected to SCBS $j \in \mathcal{S}$ is given by

$$\mathbf{y}_k^{\mathrm{mmW}} = \sum_{j \in \mathcal{S}} \mathbf{H}_{k,j} \mathbf{s}_{k,j} + \mathbf{W}_k^* \mathbf{z}_k \tag{4}$$

where $\mathbf{s}_{k,j} \in \mathbb{C}^{n_k}$ is the data stream vector for UE k consisting of mutually uncorrelated zero-mean symbols with $\mathbb{E}[\mathbf{s}_{k,j}^* \mathbf{s}_{k,j}] = P_{k,j}$, and $\mathbf{z}_k \in \mathbb{C}^{N_k}$ is the complex additive white Gaussian noise vector at UE k, with $\mathbf{z}_k \sim \mathcal{CN}(\mathbf{0}, N_0 \mathbf{I}_{N_k})$.

C. User Association and Transmission Rate

In [1] we showed that the dependency between user association and interference structure must be considered in mmWave cellular systems since mmWave channels are fast time-varying and have short coherence time, and also the interference structure depends on the highly directional links between BSs and UEs.

The connections between UEs and BSs can be defined by an *association vector* as

$$\beta^{(t)} \triangleq [\beta_1^{(t)}, ..., \beta_K^{(t)}]^T \tag{5}$$

where $\beta_k^{(t)}$ represents the index of BS to whom user k is associated with during time slot t. We define $\mathcal{K}_j^{(t)}$ as the *activation set* of BS j which is a subset of $\mathcal{K}$ and represents the set of active UEs in BS j at time slot t. Thus, the relationship between the activation set of BS j and the association vector can be expressed as

$$\mathcal{K}_j^{(t)} = \{k : \beta_k^{(t)} = j\} \tag{6}$$

The *load balancing* constraint for BS j is given by

$$|\mathcal{K}_j^{(t)}| \leq q_j \tag{7}$$

where q_j is the maximum number of UEs that BS j can serve simultaneously, and the *quota vector* of BSs is $\mathbf{q} = [q_1, ..., q_J]$.

The instantaneous rate received by each UE is a function of user associations. When UE k is connected to MCBS $j \in \mathcal{B}$ operating at sub-6 GHz band, its *instantaneous rate* (in bps/Hz) at time slot t is

$$R_{k,j}^{\mu\mathrm{W}}(\beta^{(t)}) = \log_2 \left(1 + \frac{P_{k,j} h_{k,j} h_{k,j}^*}{v_{k,j}} \right) \tag{8}$$

where $v_{k,j}$ is the interference plus noise value given as

$$v_{k,j} = \sum_{\substack{l \in \mathcal{K}_j^{(t)} \\ l \neq k}} P_{l,j} h_{k,l,j} h_{k,l,j}^* + \sum_{\substack{i \in \mathcal{B} \\ i \neq j}} \sum_{l \in \mathcal{K}_i^{(t)}} P_{l,i} h_{k,l,i} h_{k,l,i}^* + N_0$$

The instantaneous rate of UE k connected to SCBS $j \in \mathcal{S}$ operating at mmWave band is given by

$$R_{k,j}^{\mathrm{mmW}}(\beta^{(t)}) = \log_2 \left| \mathbf{I}_{n_k} + \mathbf{V}_{k,j}^{-1} P_{k,j} \mathbf{H}_{k,j} \mathbf{H}_{k,j}^* \right| \tag{9}$$

where $|.|$ denotes the determinant operator and $\mathbf{V}_{k,j}$ is the interference and noise covariance matrix given as

$$\mathbf{V}_{k,j} = \sum_{\substack{l \in \mathcal{K}_j^{(t)} \\ l \neq k}} P_{l,j} \mathbf{H}_{k,l,j} \mathbf{H}_{k,l,j}^* + \sum_{\substack{i \in \mathcal{S} \\ i \neq j}} \sum_{l \in \mathcal{K}_i^{(t)}} P_{l,i} \mathbf{H}_{k,l,i} \mathbf{H}_{k,l,i}^* + N_0 \mathbf{W}_k^* \mathbf{W}_k$$

Next, we can express the network *sum-rate* as

$$r(\beta^{(t)}) = \sum_{k \in \mathcal{K}} R_{k,\beta_k^{(t)}} \tag{10}$$

to be used as a measure of network performance.

Algorithm 1: UCB Load Balancing Assignment

```
Input: Time step $t$, matrix of number of BS selection $\mathbf{T}^{(t-1)}$,
       reward matrix $\mathbf{\Gamma}$, BSs' quota vector $\mathbf{q}$
Apply UCB formula: $\mathbf{\Gamma} \leftarrow \mathbf{\Gamma} + \sqrt{\frac{2\ln t}{\mathbf{T}^{(t-1)}}}$;
while $\mathbf{\Gamma}$ has nonzero entries do
    $[k, j] = \arg\max_{l\in\mathcal{K}, i\in\mathcal{J}} \Gamma_{l,i}$;
    if $q_j > 0$ then
        Associate UE $k$ with BS $j$: $\beta_k^{(t)} = j$;
        Update BS's quota: $q_j \leftarrow q_j - 1$;
        Zero out row $k$ in reward matrix $\mathbf{\Gamma}$;
    else
        Zero out column $j$ in reward matrix $\mathbf{\Gamma}$;
    end
end
Output: Association vector $\beta$
```

III. Centralized Online MAB User Association

Using the MAB technique, we introduce an online centralized load balancing algorithm for fast user association in 5G cellular networks. The goal is to adapt and learn an association vector $\beta^\star$ which specifies the best connections between BSs and UEs. In the proposed algorithm, each UE k takes an action (selects BS j) and receives an instantaneous reward $R_{k,j}$. This reward can be simply obtained based on a local measurement at UE, the mechanism for which is readily available for other purposes like handover [7]. Then, it updates the corresponding reward based on the following updating rule [5]

$$\Gamma_{k,j} \leftarrow \Gamma_{k,j} + \alpha(R_{k,j} - \Gamma_{k,j}) \tag{11}$$

where $\Gamma_{k,j}$ is the *reward* of UE k received from connection with BS j, and α is the learning rate. The *reward matrix* $\mathbf{\Gamma}$ is defined as a $K \times J$ matrix including rewards of all UE-BS pairs.

A. Load Balancing Assignment

In a multi-agent MAB user association, each UE takes an action, receive an instantaneous reward, and updates its reward. At time step t, each UE can pick the best BS based on its updated reward vector containing rewards from the connection with each BS. However, due to the load balancing constrains in (7), the decision of each UE depends on the decisions of other UEs. A collision can happen if the number of UEs simultaneously picking the same BS is more than its quota allows. In order to avoid collision, a CLB is required to collect the reward vectors of all UEs and determine user associations based on load balancing constraints of all BSs.

A load balancing assignment algorithm produces a load balanced association vector based on the most recent reward matrix (collected from all UEs) and the quota of BSs. Because of this gathering of information from all UEs and BSs, it needs to be executed by a central entity, the CLB. In this paper, we propose a load balancing assignment scheme based on the Upper-Confidence-Bound (UCB) action selection method [5]. This assignment scheme guarantees a balance between exploiting the current best action and exploring other possible actions for all UEs.

B. UCB Load Balancing Assignment by the CLB

In the UCB action selection approach, each UE k wants to be associated with the BS which provides the highest possible reward by selecting a BS as follows [5]

$$j = \arg\max_{i\in\mathcal{J}} \left(\Gamma_{k,i} + \sqrt{\frac{2\ln t}{T_{k,i}^{(t-1)}}}\right) \tag{12}$$

where t is the time step, and $T_{k,i}^{(t-1)}$ represents the number of times UE k has been associated with BS i up to and including time step $t-1$. This mechanism guarantees a certain and diminishing amount of exploration during the learning process. If there were no quotas on the BSs, then each user k can directly implement the resulting choice of (12) as the association decision for the next learning step. With load balancing constraints, however, we need to modify these decisions in order to satisfy the BSs' quotas.

We propose the following BS load balancing assignment scheme to be performed at the CLB which has the knowledge of the entire reward matrix $\mathbf{\Gamma}$ and also matrix $\mathbf{T}$ (made up of elements $T_{k,i}$). The assignment algorithm repeatedly performs the following two steps:

1) Select the following UE-BS pair

$$[k, j] = \arg\max_{l\in\mathcal{K}, i\in\mathcal{J}} \left(\Gamma_{l,i} + \sqrt{\frac{2\ln t}{T_{l,i}^{(t-1)}}}\right) \tag{13}$$

2) Zero out row k of $\mathbf{\Gamma}$, and zero out column j if the quota of BS j is full, to form a new $\mathbf{\Gamma}$.

until it has identified associations for all users. In other words, an association occurs according to (13) by selecting the UE-BS connection with the highest reward overall. After an association happens, the association vector β is updated, the corresponding row from $\mathbf{\Gamma}$ is zeroed out, and the quota of serving BS is updated. If a BS runs out of quota, we zero out the corresponding column from $\mathbf{\Gamma}$. These steps are repeated until $\mathbf{\Gamma} = \mathbf{0}$. At this point, the balanced association vector β is complete and specifies the associations of all UEs. A summary of UCB load balancing assignment is given in Alg. 1.

C. Centralized MAB User Association Algorithm

Using the load balancing scheme above, we propose a centralized online user association algorithm based on MAB technique. The proposed algorithm can be implemented online and can track network dynamics including small-scale (instantaneous) and large-scale

```
Algorithm 2: Centralized MAB User Association
Input: UEs' learning rates $\alpha_k$, randomly generated reward
       matrix $\mathbf{\Gamma}$, BSs' quota vector $\mathbf{q}$, initial association
       vector $\beta^{(0)}$, initial matrix of number of BS selection
       $\mathbf{T}^{(0)} = \mathbf{0}$
for $t = 1 : T$ do
    Each UE $k$:
    - Connects to BS $j = \beta_k^{(t-1)}$;
    - Receives reward $R_{k,j}$ from BS $j$ and reports it to CLB;
    Central load balancer (CLB):
    - $\Gamma_{k,j} \leftarrow \Gamma_{k,j} + \alpha(R_{k,j} - \Gamma_{k,j}), \ \forall k$;
    - $T_{k,j}^{(t)} = T_{k,j}^{(t-1)} + 1, \ \forall k$;
    - Executes Alg. 1 to obtain association vector $\beta^{(t)}$ and
      informs the UEs for learning purpose;
    CLB calculates sum-rate $r(\beta^{(t)})$;
    if $r(\beta^{(t)}) > r(\beta^{(t-1)})$ then
        $\beta^\star = \beta^{(t)}$;
        Informs UEs about new best associations $\beta^\star$;
        Each UE $k$ connects with BS $\beta_k^\star$ for transmission;
    end
end
Output: Best association vector $\beta^\star$ up to time step $T$
```

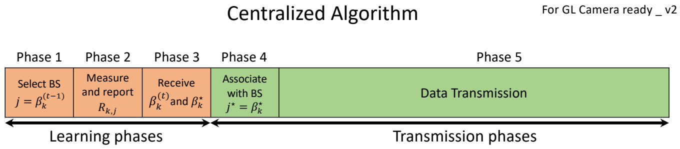
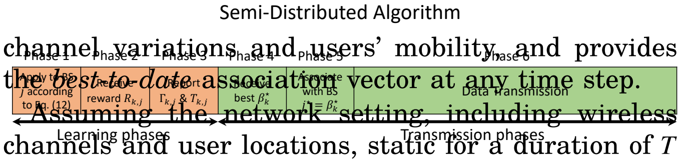

Centralized Algorithm
For GL Camera ready _ v2
Phase 1
Phase 2
Phase 3
Phase 4
Phase 5
Select BS $j = \beta_k^{(t-1)}$
Measure and report $R_{k,j}$
Receive $\beta_k^{(t)}$ and $\beta_k^\star$
Associate with BS $j^\star = \beta_k^\star$
Data Transmission
Learning phases
Transmission phases

Fig. 1. Learning and transmission phases at UE k in the centralized MAB algorithm during a single time slot t.


Semi-Distributed Algorithm
Phase 1
Phase 2
Phase 3
Phase 4
Phase 5
Phase 6
Apply to BS j according to Eq. (12)
Receive reward $R_{k,j}$
Report $\Gamma_{k,j}$ & $T_{k,j}$
Receive best $\beta_k^\star$
Associate with BS $j^\star = \beta_k^\star$
Data Transmission
Learning phases
Transmission phases

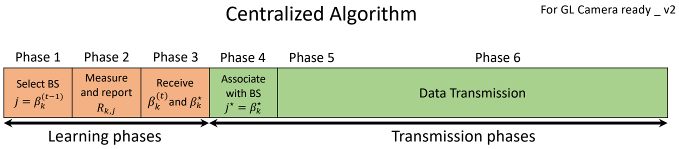
channel variations, and users' mobility, and provides the *best-to-date* association vector at any time step.

Assuming the network setting, including wireless channels and user locations, static for a duration of T learning steps (as in a block fading channel), the algorithm works as follows. In each time step, every user follows a five-phase operation: (i) performing association for learning purposes, (ii) measuring and reporting the associated reward, (iii) receiving $\beta_k^{(t)}$ and the best-to-date association $\beta_k^\star$, (iv) performing association for transmission, (v) carrying out data transmission (see Fig. 1). The first three phases are dedicated for learning which use current association result (instead of the best-to-date) to allow sufficient learning exploration, whereas the last two phases are for actual data transmission which use the best-to-date association in order to achieve the highest data rate.

In each time step $t \leq T$, during the first two phases, each UE k connects to its assigned-for-learning-purposes BS $j = \beta_k^{(t-1)}$, measures an instantaneous reward $R_{k,j}$ and reports it to the CLB. Then, in the third phase, the UE receives $\beta_k^{(t)}$ to be used in the next learning time step, and the best-to-date association $j^\star = \beta_k^\star$. The UE implements this best-to-date association in phase four and maintains it for data transmission in the fifth phase. After receiving the $R_{k,j}$ in phase two, the CLB updates the reward matrix $\mathbf{\Gamma}$ and the time matrix $\mathbf{T}$. Then, it executes the UCB load balancing assignment (Alg. 1) to obtain association vector $\beta^{(t)}$ and informs all UEs of their next learning step connections. The CLB also compares the new sum-rate resulting from $\beta^{(t)}$ with the current best-to-date value. If the new sum-rate is higher, CLB updates the best-to-date association vector as $\beta^\star = \beta^{(t)}$, and informs all UEs of this new $\beta^\star$.

When $t > T$, or the network setting changes, the CLB resets its best-to-date values and re-starts the process. This online centralized algorithm is shown in Alg. 2.

```
Algorithm 3: Semi-dist. MAB User Association
Input: UEs' learning rate $\alpha$, BSs' quota vector $\mathbf{q}$, randomly
       generated reward matrix $\mathbf{\Gamma}$, initial matrix of number
       of BS selection $\mathbf{T}^{(0)} = \mathbf{0}$
for $t = 1 : T$ do
    $\mathbf{q}^{\text{temp}} = \mathbf{q}$;
    Each UE $k$:
    - Applies to best BS in its reward vector as in (12);
    if $q_j^{temp} > 0$ then
        UE $k$ receives new reward $R_{k,j}$ from BS $j$;
        BS $j$ updates its quota: $q_j^{\text{temp}} \leftarrow q_j^{\text{temp}} - 1$;
    else
        BS $j$ rejects UE $k$;
        UE $k$ receives new reward $R_{k,j} = 0$
```


Centralized Algorithm
For GL Camera ready _ v2
Phase 1
Phase 2
Phase 3
Phase 4
Phase 5
Phase 6
Select BS $j = \beta_k^{(t-1)}$
Measure and report $R_{k,j}$
Receive $\beta_k^{(t)}$ and $\beta_k^\star$
Associate with BS $j^\star = \beta_k^\star$
Data Transmission
Learning phases
Transmission phases

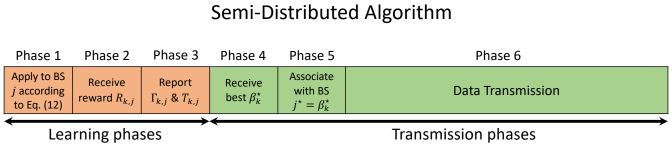

Semi-Distributed Algorithm
Phase 1
Phase 2
Phase 3
Phase 4
Phase 5
Phase 6
Apply to BS j according to Eq. (12)
Receive reward $R_{k,j}$
Report $\Gamma_{k,j}$ & $T_{k,j}$
Receive best $\beta_k^\star$
Associate with BS $j^\star = \beta_k^\star$
Data Transmission
Learning phases
Transmission phases

Fig. 2. Learning and transmission phases at UE k in the semi-distributed MAB algorithm during a single time slot t.

IV. Semi-distributed Online MAB User Association

A. Distributed User Association

Distributed user association approaches are of significant interest for future cellular networks which have short channel coherence time and low-latency requirements. Distributed approaches provide low-complexity solutions with minimal signaling overhead between network entities. Distributed algorithms performance, however, is usually worse than that of centralized algorithms since association decisions are made based on local and not global information.

For load balancing user association, the difficulty in implementing a fully-distributed algorithm comes from the fact that the association decision of each individual UE based on their local information does not guarantee load balancing. This drawback is due to the lack of information about the association of other UEs. Considering the proposed centralized MAB

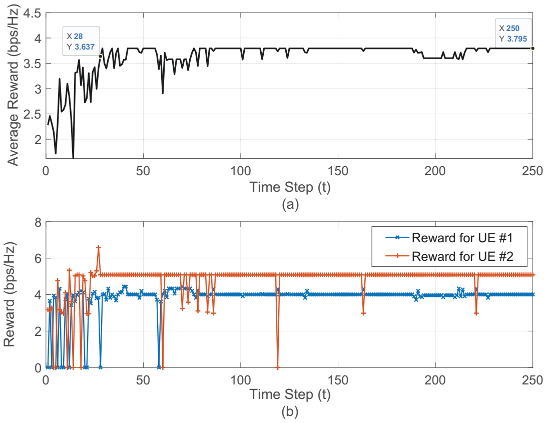


Fig. 3. a) Average received rewards of UEs, and b) Reward of two typical UEs versus time step for the centralized MAB algorithm.

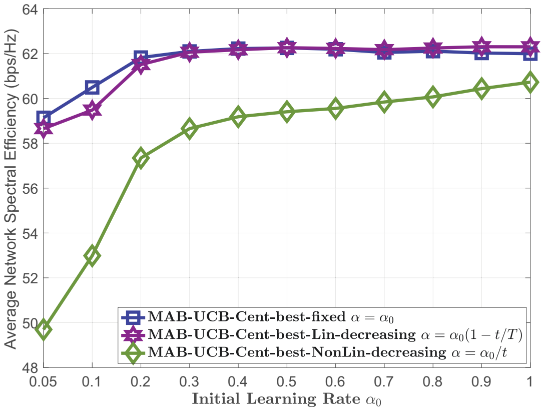


Fig. 4. Effect of learning rate α on the average network spectral efficiency of MAB centralized algorithm ($T = 50$).

algorithm in Sec. III, each UE can perform its learning procedure in a distributed fashion, but we still need a central entity to track the association of all UEs and enforce the load balancing constraints. This idea leads us to a semi-distributed MAB user association algorithm introduced next.

B. Semi-distributed MAB User Association Algorithm

In a semi-distributed algorithm, instead of receiving an action from the CLB, each UE proposes an action based on its locally updated reward vectors. At each time step, each UE follows a six-phase operation: (i) applying to a BS, (ii) receiving a reward, (iii) reporting updated reward, (iv) receiving best-to-date association from CLB, (v) performing association for transmission, and (vi) carrying out data transmission (see Fig. 2).

In particular, each UE k uses the UCB formula in (12) to find best BS providing highest reward. Then, each UE executes an *apply-response mechanism*, in which it applies to its best BS and receives an instantaneous reward. The reward will be a positive value if the BS has enough quota, but will be zero if the BS is fully loaded. Based on this instantaneous reward, the UE updates its local reward value according to (11) and also the number of times it has applied to that BS. Then, each UE reports these updates to the CLB. In this algorithm, similar to Alg. 2, the CLB is responsible for balancing the loads of BSs by performing the *while loop* in Alg. 1 using the updated rewards, keeping track of the best-to-date association $\beta^{\star}$, and informing UEs about their best-to-date load-balanced associations. The UEs then use this best-to-date association for data transmission.

This algorithm is semi-distributed in the sense that each UE updates its reward based on its own decision, instead of the CLB updating rewards as in Alg. 2. This online semi-distributed algorithm is given in Alg. 3.

V. Numerical Results

We evaluate the performance of the proposed MAB algorithms in the downlink of a 5G HetNet with $J = 4$ BSs, $K = 18$ UEs, and BSs' quota vector $\mathbf{q} = [9, 3, 3, 3]$. The network includes 1 MCBS operating at 1.8 GHz and 3 SCBSs operating at 28 GHz. The channels for sub-6 GHz links and the mmWave links are generated as described in Sec. II-B. We assume each mmWave link is composed of 5 clusters with 10 rays per cluster. In order to implement 3D beamforming, each BS is equipped with a UPA of size 8×8 ($M_j = 64$), and each UE is equipped with an antenna module designed for sub-6 GHz band, and a 4×1 ULA of antennas designed for mmWave band ($N_k = 4$). Each UE can receive one data stream at sub-6 GHz and two data streams at mmWave band ($n_k = 2$). Also, we assume that the transmit power of MCBS is 10dB higher than that for SCBSs. Network nodes are deployed in a 300×300 m^2 square where the BSs are placed at specific locations and the UEs are distributed randomly according to a homogeneous Poisson point process (PPP).

Fig. 3 depicts the average received rewards of UEs and reward of two typical UEs with respect to time steps for the centralized MAB algorithm. Subfigure (a) shows a clear trend of that the average reward increasing as time step grows. The small number of time steps required for reaching near maximum average is encouraging, where the average reward reaches to 95% of the maximum in only 28 time steps. This result indicates that online implementation of the proposed algorithm can reach close to its best performance even in highly dynamic networks. Subfigure (b) shows the reward for two typical UEs where deep valleys in the curves indicate the result of sub-optimal actions.

The effect of learning rate α on the average spectral efficiency of the centralized MAB algorithm is depicted in Fig. 4. This figure shows three cases for learning rate

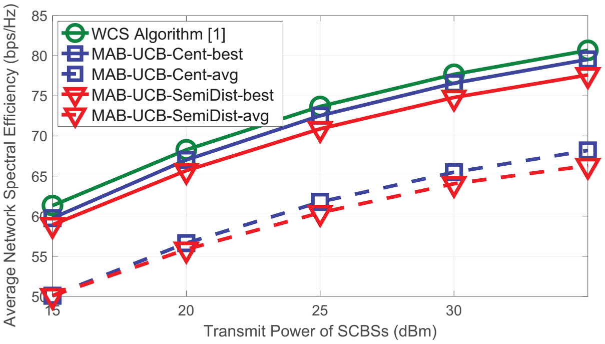


Fig. 5. Average network spectral efficiency of the proposed centralized and semi-distributed MAB algorithms with $T = 50$, in comparison with the non-learning near-optimal centralized WCS algorithm (Note that WCS has been shown in [1] to outperform existing non-learning algorithms such as those in [8].)

α: 1) fixed $\alpha = \alpha_0$, 2) linear decrement $\alpha^{(t)} = \alpha_0(1-t/T)$, and 3) nonlinear decrement $\alpha^{(t)} = \alpha_0/t$. In the first case, α is fixed throughout the learning algorithm, however, in the last two cases α is decreasing from the initial value α_0 as the number of time steps increases according to the given functions. The results indicate that fixed learning rate α leads to slightly better performance than the linear and nonlinear decrement cases. For next simulations, we pick fixed value $\alpha = 0.3$.

Fig. 5 compares the performance of the proposed MAB user association algorithms with a benchmark as the (non-learning) centralized WCS algorithm [1], where the "avg" curves are average performance of the learned solutions (instead of best-to-date) at each time step. This figure shows that selecting the best-to-date association vector achieves a performance close to that of WCS algorithm. In particular, centralized and semi-distributed MAB algorithms with best-to-date association vector achieve 98% and 96% of the solution provided by WCS algorithm, respectively .

Fig. 6 depicts the best-to-date average network spectral efficiency versus time for the centralized MAB algorithm for two network sizes. The centralized algorithm only takes 25 time steps to reach 95% of the maximum efficiency for the smaller network, however, it needs around 55 time steps to achieve the same efficiency for the larger network. In Fig. 7, we study the effect of increasing network size on the number of time steps required to achieve 95% of the maximum sum-rate. The figure shows that the centralized algorithm requires fewer number of time steps to achieve the same performance as the semi-distributed algorithm.

VI. Conclusion

Using MAB techniques, we proposed a centralized and a semi-distributed load balancing user association algorithms. The algorithms explicitly satisfy the load balancing constraints by employing a central load balancer to associate UEs with BSs based on their quotas.

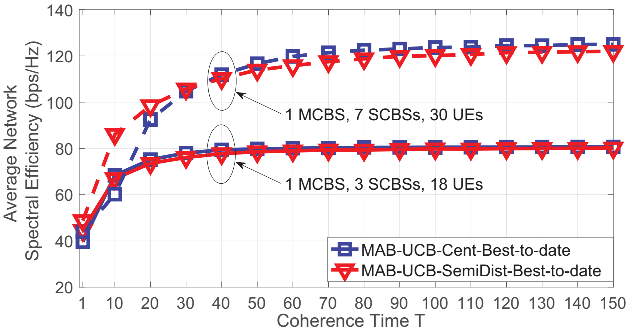


Fig. 6. Effect of increasing coherence time on the average network spectral efficiency of proposed MAB algorithms.

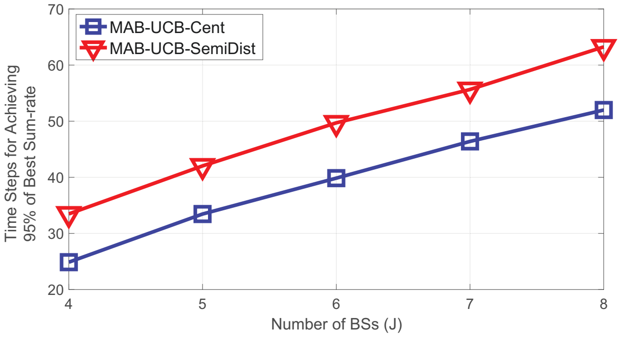


Fig. 7. Effect of increasing network size (with $\mathbf{q} = \{9, 3, ..., 3\}$) on the number of time steps required to achieve 95% of the best sum-rate.

Moreover, the proposed algorithms can be implemented online and adapt user associations to the network dynamics. Our simulations showed that the learning process in these algorithms is fast and efficient. We also observed that performance of these algorithms reaches closely to that of the benchmark near-optimal WCS algorithm. These features make our proposed algorithms potentially suitable for online user association in highly-dynamic HetNets.

References

[1] A. Alizadeh and M. Vu, "Load balancing user association in millimeter wave mimo networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 6, pp. 2932–2945, 2019.

[2] A. Zappone et al., "User Association and Load Balancing for Massive MIMO through Deep Learning," in *Proc. 52nd Asilomar Conf. on Signals, Systems, and Computers*, Oct. 2018.

[3] M. Sana et al., "Multi-Agent Deep Reinforcement Learning Based User Association for Dense mmWave Networks," in *Proc. IEEE GLOBECOM*, Dec. 2019.

[4] D. Li et al., "User Association and Power Allocation Based on Q-Learning in Ultra Dense Heterogeneous Networks," in *Proc. IEEE GLOBECOM*, Dec. 2019.

[5] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. The MIT Press, 2015.

[6] S. Maghsudi et al., "Distributed user association in energy harvesting dense small cell networks: A mean-field multi-armed bandit approach," *IEEE Access*, vol. 5, pp. 3513–3523, 2017.

[7] 3GPP, "5G NR; Requirements for support of radio resource management," TS 38.133, Oct. 2018, v. 15.3.0.

[8] D. Bethanabhotla et al., "Optimal user-cell association for massive MIMO wireless networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 3, pp. 1835–1850, 2016.