User Grouping and Power Allocation in NOMA Systems: A Reinforcement Learning-Based Solution

Rebekka Olsson Omslandseter[1], Lei Jiao[1], Yuanwei Liu[2], and B. John Oommen[1,3]

[1] University of Agder, Grimstad, Norway {rebekka.o.omslandseter,lei.jiao}@uia.no
[2] Queen Mary University of London, London, UK yuanwei.liu@qmul.ac.uk
[3] Carleton University, Ottawa, Canada oommen@scs.carleton.ca

Abstract. In this paper, we present a pioneering solution to the problem of user grouping and power allocation in Non-Orthogonal Multiple Access (NOMA) systems. There are two fundamentally salient and difficult issues associated with NOMA systems. The first involves the task of grouping users together into the pre-specified time slots. The subsequent second phase augments this with the solution of determining how much power should be allocated to the respective users. We resolve this with the first reported Reinforcement Learning (RL)-based solution, which attempts to solve the partitioning phase of this issue. In particular, we invoke the Object Migration Automata (OMA) and one of its variants to resolve the user grouping problem for NOMA systems in stochastic environments. Thereafter, we use the consequent groupings to infer the power allocation based on a greedy heuristic. Our simulation results confirm that our solution is able to resolve the issue accurately, and in a very time-efficient manner.

Keywords: Learning Automata · Non-Orthogonal Multiple Access · Object Migration Automata · Object Partitioning

1 Introduction

The Non-Orthogonal Multiple Access (NOMA) paradigm has been proposed and promoted as a promising technique to meet the future requirements of wireless capacity [6]. With NOMA, the diversity of the users' channels and power is exploited through Successive Interference Cancellation (SIC) techniques in receivers [1]. This technology introduces questions concerning users who are ideally supposed to be grouped together, so as to obtain the maximum capacity gain. Additionally, the power level of the signal intended for each user is a crucial component for the successful SIC in NOMA operations. Consequently, it is an accepted fact that the performance of NOMA is highly dependent on both the grouping of the users and the power allocation.

The user grouping and power allocation problems in NOMA systems are, in general, intricate. First of all, the user grouping problem, in and of itself, introduces a combinatorially difficult task, and is infeasible as the number of users increases. This is further complicated by the channel conditions, and the random nature of the users' behaviors in communication scenarios. For this reason, the foundation for grouping, and consequently, for power allocation, can change rapidly. It is, therefore, necessary for a modern communication system to accommodate and adapt to such changes.

The user grouping in NOMA systems, is akin to a classic problem, i.e., to the Object Partitioning Problem (OPP). The OPP concerns grouping "objects" into sub-collections, with the goal of optimizing a related objective function, so as to obtain an optimal grouping [3]. Our goal is utilize Machine Learning (ML) techniques to solve this, and in particular, the ever-increasing domain of Reinforcement Learning (RL), and its sub-domain, of Learning Automata (LA). When it concerns RL-based solutions for the OPP, the literature reports many recent studies to solve Equi-Partitioning Problems (EPPs). EPPs are a sub-class of the OPP, where all the groups are constrained to be of equal size. Among these ML solutions, the Enhanced Object Migration Automata (EOMA) performs well for solving different variants of EPPs [2]. They can effectively handle the stochastic behavior of the users, and are thus powerful in highly dynamic environments, similar to those encountered in the user grouping phase in NOMA systems.

Moving now to the second phase, the task of allocating power to the different users of a group in NOMA systems, further complicates the NOMA operation. However, a crucial observation is that the problem resembles a similar well-known problem in combinatorial optimization, i.e., the *Knapsack Problem* (KP). KPs, and their variants, have been studied for decades [4], and numerous solutions to such problems have been proposed. Among the numerous solutions, it is well known that many fundamental issues can be resolved by invoking a greedy solution to the KP. This is because a greedy solution can be exquisite to a highly complex problem, and can quickly utilize a relation among the items, to yield a near-optimal allocation of the resources based on this relation. The power allocation problem in NOMA systems can be modeled as a variation of a KP, and this can yield a near-optimal solution based on such a greedy heuristic.

In this paper, we concentrate on the problem's stochastic nature and propose an adaptive RL-based solution. More specifically, by invoking a technique within the OMA paradigm, we see that partitioning problems can be solved even in highly stochastic environments. They thus constitute valuable methods for handling the behavior of components in a NOMA system. In particular, we shall show that such methods are compelling in resolving the task of grouping the users. Indeed, even though the number of possible groupings can be exponentially large, the OMA-based scheme yields a remarkably accurate result within *a few hundred iterations*. This constitutes the first phase of our solution. It is pertinent to mention that the solution is unique, and *that we are not aware of any analogous RL-based solution for this phase of NOMA systems*.

The second phase groups users with different channel behaviors, and allocates power to the respective users. Here, we observe that the power allocation problem can be mapped onto a variation of a KP. Although the types of reported KPs are numerous, our specific problem is more analogous to a *linear* KP. By observing this, we are able to resolve the power allocation by solving a linear (in the number of users) number of algebraic equations, all of which are also *algebraically linear*. This two-step solution constitutes a straightforward, but comprehensive strategy. Neither of them, individually or together, has been considered in the prior literature.

The paper is organized as follows. In Section 2, we depict the configuration of the adopted system. Then, in Section 3, we formulate and analyze the optimization problem. Section 4 details the proposed solution for the optimization problem. We briefly present numerical results in Section 5, and conclude the paper in Section 6.

2 System description

Consider a simplified single-carrier down-link cellular system that consists of one base station (BS) and K users that are to be divided into N groups for NOMA operation. NOMA is applied to each group, but different groups are assigned to orthogonal resources. For example, one BS assigns a single frequency band to the K users. The users are to be grouped in N groups, each of which occupies a time slot. User k is denoted by U_k where $k \in \mathcal{K} = \{1,2,\ldots,K\}$. Similarly, the set of groups are denoted by $\mathcal{G} = \{g_n\}$, $n \in \mathcal{N} = \{1,2,\ldots,N\}$, where g_n is the set of users inside the n-th group. The groups are mutually exclusive and collectively exhaustive, and thus, $g_n \cap g_o = \emptyset$ with $n \neq o$. When a User U_k belongs to Group n, we use the notation $U_{n,k}$ to refer to this user and its group. We adopt the simplified notation U_k to refer to a user when the user's group is trivial or undetermined. Thus, if we have 4 users in the system, User 1 and User 3 could belong to Group 1, and User 2 and User 4 belong to group 2. In this case, when we want to refer to User 1 without its group, we use U_1. Likewise, when we want to mention User 4 belonging to Group 2, we apply $U_{2,4}$. For mobility, the users are expected to move within a defined area. The user behavior in a university or an office building are examples of where the user behavior coincides with our mobility model.

2.1 Channel Model

The channel model coefficient for U_k is denoted by $h_k(t)$ and refers to the channel fading between the BS and U_k along time. The channel coefficient is generated based on the well-recognized mobile channel model, which statistically follows a Rayleigh distribution [8]. The parameters of the channel configuration will be detailed in the section describing the numerical results. Note that the LA solution to be proposed can handle a non-stationary stochastic process, and the solution proposed in this work is distribution-independent. Therefore, the current Rayleigh distribution can be replaced by any other channel model, based on the application scenario and environment.

2.2 Signal Model

Based on the NOMA concept, the BS sends different messages to the users of a group in a single time slot via the same frequency band. Consequently, the received signal y_k at time t for $U_{n,k}$ can be expressed as

$$y_k(t) = \sqrt{p_{n,k}}h_k(t)s_k + \sum_{e=1}^{|g_n|-1} \sqrt{p_{n,e}}h_k(t)s_e + n_k, \tag{1}$$

where e is the index of the users in the set $g_n \backslash U_{n,k}$, which is the complementary set of $U_{n,k}$ in g_n. $|g_n|$ returns the number of users in g_n. The received signal $y_k(t)$ has three parts, including the signal intended for $U_{n,k}$, the signal from all users other than $U_{n,k}$ in the same group, and the additive white Gaussian noise (AWGN) $n_k \sim \mathcal{CN}(0,\sigma_k^2)$ [10]. The transmitted signal intended for $U_{n,k}$ and $U_{n,e}$ is given by s_k and $s_e \sim \mathcal{CN}(0,1)$ respectively. $p_{n,k}$ is the allocated power for $U_{n,k}$, and the total power budget for group g_n is given by P_n.

The BS' signals are decoded at the users through SIC by means of channel coefficients in an ascending order [9]. As a result, through SIC, a user with a good channel quality can remove the interference from the users of poor channel quality, while users of poor channel quality decode their signals without applying SIC. Hence, for the User $U_{n,k}$, successful SIC is applied when $|h_{n,w}(t)|^2 \leq |h_{n,k}(t)|^2$ fulfills, where w is the index of the users that have lower channel coefficients than User k in the user Group g_n.

3 Problem Formulation

In this section, we formulate the problem to be solved. The problem is divided into two sub-problems. Specifically, in the first problem, we cluster the users into categories based on the time average of the channel coefficients. In the second step, we group the users based on the learned categories and solve the resultant power allocation problem.

3.1 Problem Formulation for the Clustering Phase

To initiate discussions, we emphasize that the channel coefficients of the users in a group need to be as different as possible so as to achieve successful NOMA operation. To group the users with different coefficients, we thus first cluster the users with *similar* coefficients, and then select one user from each cluster to formulate the groups. The first problem, the clustering problem, is formulated in this subsection. The problem for user grouping, together with power allocation, is formulated later.

The criterion that we have used for clustering the users is the *time average* of the channel coefficients, $\overline{h_k(t)}$. The reason motivating this is because the user grouping is computationally relatively costly, and the fact that the environment may change rapidly, i.e., $h(t)$ might change after channel sounding. If we cluster the users according to the time average, we can reduce the computational cost, and at the same time, capture the advantages of employing NOMA statistically.

We consider clustering users to clusters of the same size, where the number of the clusters is $L_c = K/N$, and where L_c and N are integers[4]. Let q_c be the set of users in Cluster c, where $c \in [1,2,\ldots,L_c]$ is the index of the cluster. Clearly, for the clustering problem, the difference of coefficients in each cluster needs to be minimized, and the problem can be formulated as

$$\min_{\{\varphi_{c,k}\}} \quad \sum_{c=1}^{L_c} \sum_{k=1}^{K} \varphi_{c,k} |\overline{h_k(t)} - E_c|, \tag{2a}$$

$$\text{s.t.} \quad \sum_{c=1}^{L_c} \sum_{k=1}^{K} \varphi_{c,k} = K, \quad c \in \mathcal{C}, k \in K, \tag{2b}$$

$$\sum_{k=1}^{K} \varphi_{c,k} = N, \quad \forall c, \tag{2c}$$

[4] In reality, if L_c is not an integer, we can add dummy users to the system so as to satisfy this constraint. *Dummy users* are virtual users that are not part of the real network scenario, but are needed for constituting equal-sized partitions in the clustering phase.

where $\varphi_{c,k}$ is an indicator function showing the relationship of users and clusters, as $\varphi_{c,k} = 1$ when U_k belongs to Cluster c, and 0 otherwise. Additionally, the mean value of the channel fading in each cluster is denoted by the parameters E_c and δ, which are given by $E_c = \frac{1}{\delta}\sum_{k=1}^{K} \overline{h_k(t)}\varphi_{c,k}$, and $\delta = \sum_{k=1}^{K} \varphi_{c,k}$ respectively. To explain the above equation, we mention that Eq. (2a) states the objective function. Specifically, for all the clusters, we want to minimize the difference of channel coefficients between the users within each cluster. Eq. (2b) and Eq. (2c), state a description of the variable $\varphi_{c,k}$ for User k in c. Hence, the sum of the variables $\varphi_{c,k}$ needs to be equal to the number of users, meaning that all the users need to be a part of a single cluster, and in each cluster, there needs to be an equal number of users.

The result of the clustering problem, i.e., the $\{\varphi_{c,k}\}$ that minimizes the objective function, formulates L_c sets of users, each of which has exactly N users.

3.2 Problem Formulation for the Power Allocation

From the output of the clustering, we know which users that are similar. Thus, when we take one user from each cluster and construct N groups, the size of each group is L_c. Without loss of generality, we can assume that the average channel coefficients are sorted in ascending order, i.e., $\overline{h_1(t)} \leq \overline{h_2(t)} \leq \ldots \leq \overline{h_K(t)}$. If we consider user grouping and power allocation based on average channel coefficients, the reduces to:

$$\max_{\{g_n\},\{p_{n,k}\}} \quad R = \sum_{k=1}^{K} b\log_2\left(1 + \frac{p_{n,k}|\overline{h_{n,k}}|^2}{I_{n,k} + \sigma^2}\right) \tag{3a}$$

$$\text{s.t.} \quad g_n \cap g_o = \emptyset, \;\; n \neq o, \;\; n,o \in \mathcal{N}, \tag{3b}$$

$$\sum_{j,\forall U_j \in g_n} p_{n,j} \leq P_n, \;\; n \in \mathcal{N}, \tag{3c}$$

$$R_{n,j}(t) \geq R_{QoS}, \;\; j \in \mathcal{K}, n \in \mathcal{N}, \tag{3d}$$

$$\overline{h_i(t)} > \overline{h_j(t)}, \;\; \forall i > j, \;\; i,j \in \mathcal{K}, \tag{3e}$$

$$|g_n \cap q_c| = 1, \;\; \forall c, \; \forall n, \tag{3f}$$

$$\sum_{j,\forall U_j \in g_n} \tau_{n,j} = L_c, \;\; \forall n, \tag{3g}$$

$$\sum_{n,\forall n \in \mathcal{N}} \sum_{j,\forall U_j \in g_n} \tau_{n,j} = NL_c. \tag{3h}$$

In Eq. (3a), $I_{n,k} = \sum_{\substack{j, \\ \forall j>k, \; \{U_j, \, U_k\} \in g_n}} |\overline{h_k}|^2 p_{n,j}$ is the interference to User k in Group n. In Eq. (3b), we state that the groups need to be disjoint. Hence, one user can only be in one group. In (3c), we address the constraint for the power budget. The QoS constraint is given in Eq.(3d), where $R_{n,j}(t)$ is the achievable data rate for User j in Group n, and Eq. (3e) gives the SIC constraint. The constraint in Eq. (3f) specifies that only a single user is selected to formulate a group from each cluster. In Eq. (3g), we introduce an indicator $\tau_{n,k}$, stating whether U_k is in Group n, as $\tau_{n,k} = 1$ when U_k belongs to Group n, and 0 otherwise. Furthermore, all users should belong to a certain group, which is given in Eq. (3h). Table 1 summarizes the notation.

Notation	Description	Notation	Description
h, $h(t)$	Channel coefficient, and h for t	$p_{n,k}$	Allocated power for $U_{n,k}$
h_k, $h_{n,k}$	h for U_k and $U_{n,k}$	n_k, σ^2	AWGN at U_k and Gaussian noise
$\overline{h_k(t)}$, $\overline{h_{n,k}(t)}$	The mean of h for U_k	$h_k(t)$, $h_{n,k}(t)$	h for U_k and $U_{n,k}$ at t
K	Total number of users	$I_{n,k}$	Interference from other users to $U_{n,k}$
N	Total number of groups	E_c	Mean of channel fading in q_c
$\mathcal{K}$	Set of user indexes	R	Total data rate (capacity)
$\mathcal{N}$	Set of group indexes	S	Number of states per action
$\mathcal{G}$	Set of groups	R_{QoS}	Minimum required data rate for a user
g_n	Set of users inside the n-th group	v_U, v_L	Mobility factor and speed of light
$\lvert g_n \rvert$	Number of users inside g_n	f_c, f_d	Carrier and Doppler frequency
U_k, $U_{n,k}$	User k and user k in group n	L_c	Number of clusters
$g_n \backslash U_{n,k}$	The complementary set of users in set g_n	q_c	Set of users in cluster c
$\emptyset$	Empty set	$\mathcal{C}$	Set of clusters
y_k, $y_k(t)$	Signal from BS at U_k and U_k at time t	$\varphi_{c,k}$	Indicator of whether U_k is in cluster c
s_k	Transmitted signal intended for U_k	δ	Number of $\varphi_{c,k} = 1$ for a cluster
P_n	Power budget for g_n	b	Channel bandwidth
$\tau_{n,k}$	Indicator of whether U_k is in group n	r_k	Rank of U_k
Δ_t	Time period for average of h	Υ_k	Ranking category of user k
ε_k	Index of the current state of user k	Θ_k	Cluster of U_k
$Q = (U_a, U_b)$	Input *query* of users to the EOMA	W, $Mbps$	Watt and Megabits per second
r	Rank	c	Index of the set of clusters

Table 1. Summary of notations

4 Solution to User Grouping and Power Allocation

The problem of grouping and power allocation in NOMA systems is two-pronged. Therefore, in Section 4.1, we only consider the first issue of the two, namely the grouping of users. We will show that our solution can handle the stochastic nature of the channel coefficients of the users, while also being able to follow changes in their channel behaviors over time. This will ensure that the system will be able to follow the nature of the channels in a manner that is similar to what we will expect in a real system. Thereafter, in Section 4.2, we will present our solution to the power allocation problem. Once the groups have been established in Section 4.1, we can utilize these groups to allocate power among them either instantaneously, or over a time interval using a greedy solution to the problem.

4.1 Clustering Through EOMA

The family of OMA algorithms are based on *tabula rasa* Reinforcement Learning. Without any prior knowledge of the system parameters, the channels, or the clusters, (as in our case), the OMA self-learns by observing, over time, the Environment that it interacts with. For our problem, the communication system constitutes the Environment, which can be observed by the OMA through, e.g., channel sounding. By gaining knowledge from the system behavior and incrementally improving through each interaction with the Environment, the OMA algorithms are compelling mechanisms for solving complex and stochastic problems. In the OMA, the users of our system need to be represented as abstract objects. Therefore, as far as the OMA is concerned, the users are called “objects”. The OMA algorithms require a number of states per action, indicated by S. For the LA, an action is a solution that the algorithm can converge to. In our system, the actions are the different clusters that the objects may belong to. Hence,

based on the current state of an object, we know that object's action, which is equal to its current cluster in our system. Therefore, each object, or user in our case, has a given state indicated by $\varepsilon_k = \{1,2,...,SL_c\}$, where ε_k denotes the current state of U_k, S is the number of states per action, and L_c is the number of clusters. Clearly, because we have L_c clusters, the total number of possible states is SL_c. To indicate the set of users inside Cluster c, where $c \in [1,2,\ldots,L_c]$, we have q_c. The cluster for a given User, k, is represented by Θ_k, where the set of clusters is denoted by $\mathcal{C}$ and $\Theta_k \in \mathcal{C} = \{q_1, q_2, \ldots, q_{L_c}\}$.

Algorithm 1 Clustering of Users

Require: $h_k(t)$ for all users K
  while not converged **do** // Converged if all users are in the two innermost states of any action
    for all K **do**
      Rank the users from 1 to K // 1 is given to the user with lowest h (K to the highest)
    end for
    for $\frac{K}{N}$ pairs (U_a, U_b) of K **do** // The pairs are chosen uniformly from all possible pairs
      if $\Upsilon_a = \Upsilon_b$ **then** // If U_a and U_b have the same ranking category
        if $\Theta_a = \Theta_b$ **then** // If U_a and U_b are clustered together in the EOMA
          Process Reward
        else // If U_a and U_b are not clustered together in the EOMA
          Process Penalty
        end if
      end if
    end for
  end while // Convergence has been reached

The states are central to the OMA algorithms, and the objects are moved in and out of states as they are penalized or rewarded in the Reinforcement Learning process. When all objects have reached the two innermost states of an action, we say that the algorithm has converged. When convergence is attained, we consider the solution that the EOMA algorithm has found to be sufficiently accurate. In the EOMA, the numbering of the states follows a certain pattern. By way of example, consider a case of three possible clusters: the first cluster of the EOMA has the states numbered from 1 to S, where the innermost state is 1, the second innermost state is 2, and the boundary state is S. The second cluster has the innermost state $S+1$ and the second innermost state $S+2$, while the boundary state is $2S$. Likewise, for the third cluster, the numbering will be $2S+1$ for the innermost and $2S+2$ for the second innermost state, while $3S$ is the boundary state.

Algorithm 1 presents the overall operation for the clustering of the users. The functionality for reward and penalty, as the EOMA interacts with the NOMA system, are given in Algorithms 2 and 3 respectively. In the algorithms, we consider the operation in relation to a pair of users U_a and U_b, and so $Q = \{(U_a, U_b)\}$. The EOMA considers users in pairs (called *queries*, denoted by Q). Through the information contained in their pairwise ranking, we obtain a clustering of the users into the different channel categories. For each time instant, Δ_t, the BS obtains values of $h_k(t)$ through channel sounding, and we use the average of Δ_t samples as the input to the EOMA ($\overline{h_k(t)}$). The

BS then ranks the users, indicated by $r_k = \{1, 2, ..., K\}$, where each U_k is given a single value of r_k for each Δt. For the ranks, $r_k = 1$ is given to the user that has the lowest channel coefficient compared to the total number of users, and $r_k = K$ is given to the user with the highest channel coefficient of the users. The others are filled in between them with ranks from worst to best. Furthermore, the values of these ranks corresponds to ranking categories, denoted by Υ_k for U_k, where $\Upsilon_k = \{r \in [1, N] = 1, r \in [1+N, 2N] = 2, r \in [1+2N, 3N] = 3, \ldots, r \in [K-N+1, K] = L_c\}$. In this way, even if the users have similar channel conditions, they will be compared, and the solution can work on finding the current best categorization of the K users for the given communication scenario. As depicted in Algorithm 1, we check the users' ranking categories in a pairwise manner. If the users in a pair (query) are in the same ranking category, they will be sent as a query to the EOMA algorithm. The EOMA algorithm will then work on putting the users that are queried together in the same cluster, which, in the end, will yield clusters of users with similar channel coefficients. More specifically, if two users have the same ranking category, they are sent as a query to the EOMA and the LA is rewarded if these two users are clustered together (penalized if they are not together).

Algorithm 2 Process Reward

Require: $Q = (U_a, U_b)$ // A query (Q), consisting of U_a and U_b
Require: The state of U_a (ε_a) and U_b (ε_b)
  if $\varepsilon_a \; mod \; S \neq 1$ **then** // U_a not in innermost state
    $\varepsilon_a = \varepsilon_a - 1$ // Move U_a towards innermost state
  end if
  if $\varepsilon_b \; mod \; S \neq 1$ **then** // U_b not in innermost state
    $\varepsilon_b = \varepsilon_b - 1$ // Move U_b towards innermost state
  end if
  return The next states of U_a and U_b

When the algorithm converges to obtain the groups that are needed for the power allocation, we rank the users within each cluster based on $\overline{h_k(t)}$ that was obtained in the clustering process, and then formulate the groups that consist of one user from each cluster with the same rank.

4.2 Power Allocation Through a Greedy Solution

Once the grouping of the users has been established, we can allocate power to different users in such a way that the joint data rate (R) is maximized. There are numerous ways of power allocation in various communication scenarios [5, 10]. The power allocation can be replaced by any other algorithm and will not change the nature of the Reinforcement Learning procedure. However, in this paper, we will consider the problem of power allocation as a variation of the KP, and solve it through a greedy solution.

Our aim for the greedy solution is that of maximizing the total data rate of the system. Thus, the weakest user will always be limited to the minimum required data

Algorithm 3 Process Penalty

Require: $Q = (U_a, U_b)$ // A query (Q), consisting of U_a and U_b
Require: The state of U_a (ε_a) and U_b (ε_b)
  if $\varepsilon_a \ mod \ S \neq 0$ and $\varepsilon_b \ mod \ S \neq 0$ **then** // Neither of the users are in boundary states
    $\varepsilon_a = \varepsilon_a + 1$, $\varepsilon_b = \varepsilon_b + 1$ // Move U_a and U_b towards boundary state
  else if $\varepsilon_a \ mod \ S \neq 0$ and $\varepsilon_b \ mod \ S = 0$ **then** // U_b in boundary state but not U_a
    $\varepsilon_a = \varepsilon_a + 1$, $temp = \varepsilon_b$
    $x =$ unaccessed user in cluster of U_a which is closest to boundary state
    $\varepsilon_x = temp$, $\varepsilon_b = \varepsilon_a$
  else if $\varepsilon_b \ mod \ S \neq 0$ and $\varepsilon_a \ mod \ S = 0$ **then** // U_a in boundary state but not U_b
    $\varepsilon_b = \varepsilon_b + 1$, $temp = \varepsilon_a$
    $x =$ unaccessed user in cluster of U_b closest to boundary state
    $\varepsilon_x = temp$, $\varepsilon_a = \varepsilon_b$
  else // Both users are in boundary states
    $\varepsilon_y = \varepsilon_{\{a \ or \ b\}}$ // y equals a or b with equal probability, and y is the staying user
    $\varepsilon_z = \varepsilon_{\{a \ or \ b\}}$ // z is the moving user, and is a if b was chosen as y (b if a was chosen)
    $temp = \varepsilon_z$
    $x =$ unaccessed user in cluster of U_y closest to boundary state
    $\varepsilon_x = temp$
    $\varepsilon_z = \varepsilon_y$ // Move U_z to cluster of U_y
  end if
  return The next states of U_a and U_b

rate. The heuristic involves allocating the majority of the power to the users with higher values of h, and this will result in a higher sum rate for the system. Consequently, the stronger users are benefited more from the greedy solution than those with weaker channel coefficients. However, the weak users' required data rate is ensured and can be adjusted to the given scenario. The formal algorithms are not explicitly given here in the interest of brevity, and due to space limitations. They are included in [7].

5 Numerical Results

The techniques explained above have been extensively tested for numerous numbers of users, power settings etc., and we give here the results of the experiments. In the interest of brevity, and due to space limitations, the results presented are a very brief summary of the results that we have obtained. More detailed results are included in [7] and in the Doctoral Thesis of the First Author.

We employed Matlab for simulating the values of the channel coefficient, h. Additionally, we invoked a Python script for simulating the LA solution to the user grouping and the greedy solution to power allocation. The numerical results for the power allocation solution are based on the results obtained from the EOMA clustering and grouping. For the simulations, we used a carrier frequency of $5.7\,GHz$ and an underlying Rayleigh distribution for the corresponding values of $h(t)$. For the mobility in our model, we utilized a moving pace corresponding to the movement inside an office building, i.e., $v_U = 2\ km/h$. We sampled the values of h according to $\frac{1}{2f_d}$, where f_d is the Doppler frequency and f_c is the carrier frequency. The Doppler frequency can be expressed as

$f_d = f_c(\frac{v_U}{v_L})$ and v_L is the speed of light. Therefore, in the following figures, we use "Sample Number" as the notation on the X-axis. Fig. 1 illustrates the snap-shot of h values for four users and the principle for the simulation when the number of users increased.

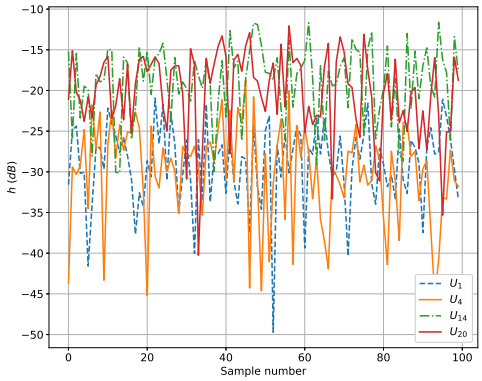

Fig. 1. Example of the simulated $h(t)$ for four different users. In the interest of clarity, and to avoid confusion, we did not plot all the 20 users.

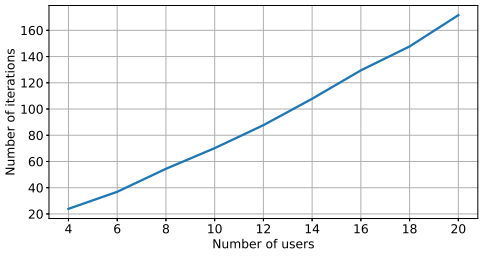

Fig. 2. A plot of the average number of iterations needed before convergence, as a function of number of users, where there were two users in each group. This was obtained by executing 100 independent experiments.

For evaluating the simulation for the clustering phase, we recorded whether or not the LA were able to determine the clusters that corresponded to the minimized difference between the users in a cluster, based on the users' mean values of h in the simulations. Remarkably, in the simulation, the EOMA yielded a 100 % accuracy in which the learned clustering was identical to the unknown underlying clustering in every sin-

gle run for the example provided with $-10dB$ difference between values of h within the different clusters. This occurred for groups of sizes $4, 6, 8, 10, 12, 14, 16, 18$ and 20, where the number of users in a group was equal to two. The difference between the users can be replaced by any "equivalent metric", and it should be mentioned that these values were only generated for testing the solution, since in a real scenario, the "True partitioning" is always unknown. The number of iterations that it took the EOMA to achieve 100 % accuracy for the different number of users is depicted in Fig. 2. Notably, the EOMA retains its extremely high accuracy as the number of users increased, and yielded 100 % accuracy both for 4 users as well as 20 users.

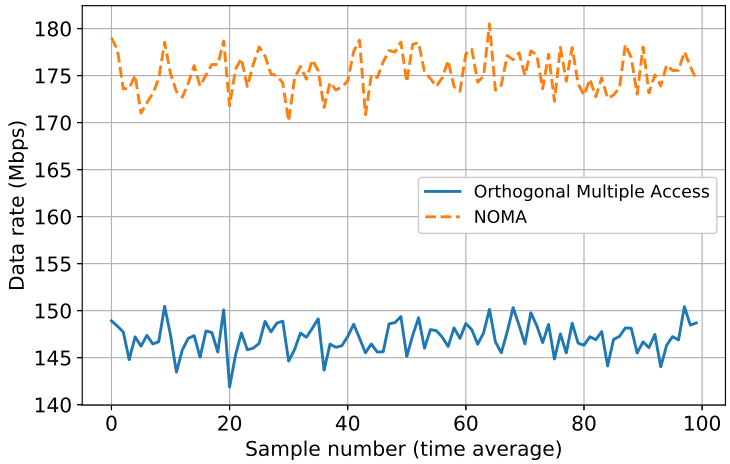

Fig. 3. Data rate for orthogonal multiple access compared to NOMA for time averages of h. Based on averages over 500 samples of h.

The simulation for the greedy solution to the power allocation phase, was carried out based on the groups established in the LA solution. Again, in the interest of brevity, we only report the results for the cases with 20 users in total and 2 users in each group, and more extensive results are included in [7] and in the Doctoral Thesis of the First Author. The optimal power allocation for 2 users in a group was obtained when we gave the minimum required power to the user with smaller h value and then allocated the rest to the user with lager h value. The optimality of the greedy algorithm for the 2 user-group case was verified by an alternate independent exhaustive search. For illustrating the advantages of our NOMA greedy solution when NOMA was employed, we compared it with the data rate that would be achieved with orthogonal multiple access[5].

[5] The achieved data rate for User k in Group n in orthogonal multiple access is given by $R_{n,k} = \frac{1}{2}\log_2\left(1+\frac{P_n|h_{n,k}|^2}{\sigma^2}\right)$. The factor $\frac{1}{2}$ is due to the multiplexing loss when 2 users share the orthogonal resource.

In Fig. 3, we depict the results obtained for the greedy NOMA solution together with the orthogonal multiple access, for an average over $\Delta t = 5$ samples of h. Further, with regard to the parameters used, the data rate for the simulations depicted in Fig. 3 was based on the following configuration: The minimum required data rate was configured to 2.0 $Mbps$, the noise to 10^{-8} W, the bandwidth to 1 MHz, and the power level for all groups to 0.125 W. As illustrated, the simulation results obtained show that the greedy solution to the power allocation is higher than the data rate achieved with orthogonal multiple access. From the graph in Fig. 3 we see that the average difference between the orthogonal multiple access and NOMA was approximately 28.17 $Mbps$.

6 Conclusions

In this paper, we have proposed a novel solution to the user grouping and power allocation problems in NOMA systems, where we have considered the stochastic nature of the users' channel coefficients. The grouping has been achieved by using the *tabula rasa* Reinforcement Learning technique of the EOMA, and the simulation results presented demonstrate that a 100 % accuracy for finding clusters of similar $h(t)$ over time can be obtained within a limited number of iterations. With respect to power allocation, we proposed a greedy solution, and again the simulation results confirm the advantages of the NOMA solution. Our solutions offer flexibility, as both the grouping and the power allocation phases, can be used as stand-alone components of a NOMA system.

References

1. Cui, J., Ding, Z., Fan, P., Al-Dhahir, N.: Unsupervised Machine Learning-Based User Clustering in Millimeter-Wave-NOMA Systems. IEEE Transactions on Wireless Communications **17**(11), 7425–7440 (Nov 2018)
2. Gale, W., Das, S., Yu, C.T.: Improvements to an Algorithm for Equipartitioning. IEEE Transactions on Computers **39**(5), 706–710 (May 1990)
3. Glimsdal, S., Granmo, O.: A Novel Bayesian Network Based Scheme for Finding the Optimal Solution to Stochastic Online Equi-Partitioning Problems. In: 2014 13th International Conference on Machine Learning and Applications. pp. 594–599 (Dec 2014)
4. Kellerer, H., Pferschy, U., Pisinger, D.: Knapsack problems. Springer, Berlin (2004)
5. Liu, Y., Elkashlan, M., Ding, Z., Karagiannidis, G.K.: Fairness of User Clustering in MIMO Non-Orthogonal Multiple Access Systems. IEEE Communications Letters **20**(7), 1465–1468 (July 2016)
6. Liu, Y., Qin, Z., Elkashlan, M., Ding, Z., Nallanathan, A., Hanzo, L.: Nonorthogonal Multiple Access for 5g and Beyond. Proceedings of the IEEE **105**(12), 2347–2381 (Dec 2017)
7. Omslandseter, R. O., Jiao, L., Liu, Y. Oommen, B. J.: An Efficient and Fast Reinforcement Learning-Based Solution to the Problem of User Grouping and Power Allocation in NOMA Systems. Unabridged version of this paper. To be submitted for publication.
8. Pätzold, M.: Mobile Radio Channels. Wiley, Chichester, 2nd ed. edn. (2012)
9. Pischella, M., Le Ruyet, D.: Noma-Relevant Clustering and Resource Allocation for Proportional Fair Uplink Communications. IEEE Wireless Communications Letters **8**(3), 873–876 (June 2019)
10. Xing, H., Liu, Y., Nallanathan, A., Ding, Z., Poor, H.V.: Optimal Throughput Fairness Tradeoffs for Downlink Non-Orthogonal Multiple Access Over Fading Channels. IEEE Transactions on Wireless Communications **17**(6), 3556–3571 (June 2018)