Noname manuscript No.
(will be inserted by the editor)

User Grouping and Power Allocation in NOMA Systems: A Novel Semi-Supervised Reinforcement Learning-based Solution

Rebekka Olsson Omslandseter ·
Lei Jiao ·
Yuanwei Liu ·
B. John Oommen

Received: date / Accepted: date

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. The problem is highly pertinent because NOMA is a well-recognized technique for future mobile radio systems. The salient and difficult issues associated with NOMA systems involve the task of grouping users together into the pre-specified time slots, which are augmented with the question of determining how much power should be allocated to the respective users. This problem is, in and of itself, NP-hard. Our solution is the first reported Reinforcement Learning (RL)-based solution, which attempts to resolve parts of this issue. In particular, we invoke the Object Migration Automaton (OMA) and one of its variants to resolve the grouping in NOMA systems. Furthermore, unlike the solutions reported in the literature, we do not assume prior knowledge of the channels' distributions, nor of their coefficients, to achieve the grouping/partitioning. Thereafter, we use the consequent groupings to heuristically infer the power allocation. The simulation results that we have obtained confirm that our learning scheme can follow the dynamics of the channel coefficients efficiently, and that the solution is able to resolve the issue dynamically.

R. O. Omslandseter and L. Jiao
Address of the first two authors: Department of Information and Communication Technology, University of Agder, Jon Lilletuns vei, 4879 Grimstad, Norway. E-mail: rebekka.o.omslandseter@uia.no, lei.jiao@uia.no.

Yuanwei Liu
Address: School of Electronic Engineering and Computer Science, Queen Mary University of London, Mile End Road, London, E1 4NS, U.K. E-mail: yuanwei.liu@qmul.ac.uk.

B. John Oommen
Chancellor's Professor; *Life Fellow: IEEE* and *Fellow: IAPR*. Address: School of Computer Science, Carleton University, Ottawa, Canada: K1S 5B6. This author is also an *Adjunct Professor* with the University of Agder in Grimstad, Norway. E-mail: oommen@scs.carleton.ca.

Keywords Learning Automata · Non-Orthogonal Multiple Access · Object Migration Automaton · Object Partitioning · Semi-Supervised Reinforcement Learning

1 Introduction

The Non-orthogonal Multiple Access (NOMA) paradigm has been established as a promising technique to meet the future requirements of wireless capacity [1]. As user demands are increasing due to the ever-increasing range of applications and technologies, (such as the Internet of things) (IoT), NOMA constitutes a valuable solution, as more users can be multiplexed together in the same orthogonal Resource Block (RB) [2]. With NOMA, the diversity of the user's channel and power are exploited through Successive Interference Cancellation (SIC) techniques in receivers [3]. The RB sharing introduces questions as to which users are the most ideal candidates to be grouped together, so as to obtain the maximum gain of capacity. Additionally, the power level of the signal intended for each user is a crucial component for successful SIC in NOMA operation. Therefore, the performance of NOMA is highly dependent on both the user grouping *and* the subsequent power allocation.

The user grouping and power allocation problems in NOMA systems are, in general, inter-twined and intricate. The user grouping problem, in and of itself, introduces a combinational in-feasibility when the number of users increases. In addition, users in NOMA systems can suffer from both inter- and intra-channel interference, constituting the non-convex property of power allocation in NOMA systems [3]. Furthermore, the channel conditions and user behavior in communication scenarios have a random nature, complicating the problem. Consequently, the foundation for grouping, and thus for power allocation, can change rapidly. Therefore, in addition to searching for instantaneous optimization, it is necessary for a modern communication system to accommodate and adapt to such changes.

In recent years, the fields of Machine Learning (ML), including RL, have been exploding, which provides new opportunities for facilitating communication systems with more capacity in terms of automation. With some insight, one sees that the problem of user grouping in NOMA systems, is similar in nature and with regard to the solution space, to a classic problem, namely, to the Object Partitioning Problem (OPP). The OPP concerns grouping objects into sub-collections and attempts to optimize a related objective function to obtain a near-optimal grouping [4]. When it concerns RL-based solutions for the OPP, many recent studies have been carried out for Equi-Partitioning Problems (EPPs). EPPs are a subset of OPPs, where all the groups (referred to as partitions) need to be equi-sized.

Among the ML solutions, the learning automata based Enhanced Object Migration Automata (EOMA) is a technique that performs well for solving different variants of EPPs [5]. Further advancements to the EOMA algorithm, that can be found in [6] and [7], are the PEOMA and TPEOMA respectively.

These OMA-based algorithms can partition a set of users into disjoint groups through the use of RL (or more precisely, Learning Automata (LA)) instances, which can handle stochastic behavior. These are powerful techniques applicable in highly dynamic environments, dealing with problem instances that are akin to the underlying ones encountered by users in NOMA systems.

The task of allocating power to the different users of a group in NOMA systems further complicates the NOMA operations. Depending on the objective of power allocation, the formulation and complexity can be quite different. Various solutions can be adopted for distinct problems, and heuristics-based solutions can be quite pertinent to such a highly complex problem.

1.1 Motivation of this Paper

In communications, a particular distribution for the channel fading is often assumed, e.g., Rayleigh fading, which involves a stochastic process in which we observe the channel, as time proceeds. When the channel coefficient, h, is assumed to follow a specified distribution that is time-invariant, it is equivalent to assuming that the stochastic process follows a random and stationary process. In previous solutions, although channel fading was assumed to be following a certain random distribution, user grouping and power allocation were traditionally carried out based on an instantaneous sample from the distribution, and thus a constant, h, was assumed to have been known, and was utilized for optimization. In other words, the stochastic behavior of channels was not handled in the prior grouping and power allocation.

Although channel sounding is not an expensive operation, it may not be carried out frequently enough to follow the instantaneous changes of the channels in certain systems. Therefore, basing a system on the most recent channel sounding result for optimization, may not be a statistically-prudent strategy. Furthermore, due to the complexity of optimization problems, in practice, the system may not prefer to solve the optimization problem frequently based on, e.g., instantaneous channel sounding results every time when they are available. In addition, the overall statistics of the channel may even change over time due to, for example, mobility. Therefore, we need a more reasonable base for the channel coefficients for optimization, and at the same time, require a more computationally-effective and adaptive solution.

1.2 Contributions of this Paper

In this paper, we study the problem considering the issue's stochastic nature, and propose an adaptive solution based on RL. To be specific, by incorporating a technique from within the OMA paradigm, partitioning problems can be solved even in environments with a highly stochastic nature. Hence, OMA algorithms constitute valuable methods for handling the behavior of components in a NOMA system. In particular, we shall show that such methods are powerful in resolving the task of grouping the users. Indeed, even though the number of possible groupings can be exponentially large, the OMA schemes can yields a remarkably accurate result, which can be achieved within a few hundred iterations! This constitutes the first phase of our solution.

The second phase groups users with different channel behaviors and allocates power to the respective users. For power allocation, we adopt the objective of maximizing the sum rate, and propose two heuristic-based algorithms. Other objective functions and the corresponding solutions may also be employed for power allocation, depending on the system's demand. This two-step solution constitutes a straightforward but comprehensive strategy, which has not been considered in the prior literature.

Our proposed solution handles random stationary environments and can learn from the environment and adjust user groupings adaptively, based on the time-averaged values of the channel's coefficients. Instead of grouping users and allocating power to users based on instantaneous measurements, which is not practical due to the complexity and stochastic nature of the problem domain, the proposed solution employs user grouping according to the time average of the communication environment. Further, based on the obtained groups, heuristic-based schemes resolve power allocation in NOMA systems. In addition, when the statistics of the environment changes, the RL algorithm can follow them so that the groupings of the users can be updated. The beauty of the proposed algorithm is that it requires no prior knowledge of the channel, *and* that the learning and adaptation are carried out while the communication system is in operation.

The reader will also observe that the problem is two-pronged. Firstly, it involves the grouping of the users, and thereafter, secondly, the corresponding power allocations. With respect to the first prong, our solution converges arbitrarily close to the optimal clustering. This, of course, does not address the power allocation problem. To address the second prong, we have resorted to straightforward heuristic-based algorithms.

Our contributions can thus be summarized as follows:

1. We study the user grouping and power allocation problem in stochastic environments. This real-life scenario is hardly addressed in the literature.
2. Through a two-step solution, the user grouping and power allocation problems are solved through a RL technique and heuristic solutions, respectively. The solution is adaptive to changes in the environment. Additionally, without prior knowledge of the channel, the system can learn the knowledge when the system is in operation, and thus both these solutions, can be implemented on the fly.
3. We provide fairly extensive simulation results to illustrate the effectiveness and the strength of RL for problems in NOMA systems.

1.3 Organization of the Paper

The paper is organized as follows[1]. First of all, Section 2 summarizes the related work in the research area of NOMA and LA. In Section 3, we depict the configuration of the adopted system, and in Section 4, the optimization problem is formulated and analyzed. Section 5 details the proposed solution for the optimization problem. Numerical results are illustrated in Section 6, before we conclude the paper in Section 7.

2 The State of the Art

In this paper, we present a solution to user grouping and power allocation in NOMA systems through the use of LA, and specifically the OMA-based partitioning algorithms. Therefore, in this section, we present the state of the art of both NOMA and LA in relation to partitioning.

2.1 NOMA

Recently, NOMA technology has attracted a great attention and research effort [1,2,8]. Substantial research has been devoted to the field of NOMA through the recent past, and user grouping and power allocation are among the many problems that have been researched.

A power allocation algorithm for NOMA networks was introduced in [9] so as to assure the fairness for users. Thereafter, in a single cell scenario, the physical layer security was studied [10]. The sum-rate and outage probability for the downlink were analyzed in [11]. For the uplink, a power back-off method was investigated in [12]. The aforementioned research effort mainly focused on the intra-cell interference analysis. A dense multiple cell network for NOMA techniques considering inter-cell interference, where both uplink and downlink transmissions were evaluated, was studied in [13]. The study of the mmWave networks with NOMA was carried out in [14] and [15] with a focus on random beamforming without considering the locations of the users. Thereafter, the “the beamforming” strategy and power allocation coefficients were jointly optimized for maximizing the throughput [16]. In addition, stochastic geometry, which is able to characterize the communication distances between transceivers by providing a spatial framework, has also been utilized in NOMA [13,17] to model the locations of primary and secondary NOMA receivers.

In terms of user grouping and the corresponding ML techniques applied in NOMA, the study is still in its infancy. In [18], a dynamic method for classifying users for power allocation was investigated. The channel coefficients were sorted from high to low, and were assigned to channels, such that the difference between the users in each group increased. In [3], the authors utilized a K-means scheme for user grouping based on geolocation, where they

[1] The notation for the paper is given below, so as to not distract from the content itself.

Table 1 Table of notations.

Notation	Description
h	Channel coefficient
h_k, $h_k(t)$, $h_{n,k}$, $h_{n,k}(t)$	h for U_k and $U_{n,k}$, and for U_k and $U_{n,k}$ at t
$\overline{h_k(t)}$, $\overline{h_{n,k}(t)}$	The mean of the channel coefficient for U_k
K	Total number of users
N	Total number of groups
$\mathcal{K}$	Set of user indexes
$\mathcal{N}$	Set of group indexes
$\mathcal{G}$	Set of groups
g_n	Set of users inside the n-th group
$\|g_n\|$	Number of users inside g_n
U_k, $U_{n,k}$	User k and user k in group n
$g_n \backslash U_{n,k}$	The complementary set of users in set g_n
$\emptyset$	Empty set
y_k, $y_k(t)$	Signal from BS at U_k and U_k at time t
s_k	Transmitted signal intended for U_k
P_n, $P_n(t)$	Power budget for g_n, and P_n considering t
$\tau_{n,k}$	Indicator of whether U_k is in group n
Δ_t	Time period for considering average of h
S	Number of states per action
ϵ_k	Index of the current state of user k
$Q = (U_a, U_b)$	Input *query* of users to the EOMA
f_c, f_d	Carrier and Doppler frequency
W	Number of combinations
$p_{n,k}$, $p_{n,k}(t)$	Allocated power for $U_{n,k}$ and considering t
n_k, σ^2	AWGN at U_k and Gaussian noise power
$\Gamma_k(t)$, $\Gamma_{n,k}$, $\Gamma_{n,k}(t)$	SINR of user U_k and $U_{n,k}$, and considering t
$I_{n,k}(t)$	Interference from other users to $U_{n,k}$ at t
b	Bandwidth of the channel
R, $R(t)$	Total data rate, and R considering t
$R_k, R_{n,k}$, $R_{n,k}(t)$	Data rate of U_k and $U_{n,k}$, and considering t
R_{QoS}	Minimum required data rate for a user
B_K	The K-th Bell number
$\left\{ {K \atop \kappa} \right\}$	Stirling number of the second kind
L_c	Number of clusters
q_c	Set of users in cluster c
$\mathcal{C}$	Set of clusters
$\varphi_{c,k}$	Indicator of whether U_k is in cluster c
δ	Number of $\varphi_{c,k} = 1$ for a cluster
$U'_{c,g,k}$	User k in cluster c and group g in (13)
r_k	Rank
Υ_k	Ranking category of user k
E_c	Mean of channel fading in q_c
Θ_k	Cluster of U_k
γ	Precision in exhaustive search
v_U, v_L	Mobility factor of users and speed of light
α_c	Action in the LA for cluster c

considered the grouping of NOMA systems in, for example, school halls. Maximum weight matching was adopted in [2] to build non-disjoint groups per RB and for subsequently allocating power, given the obtained groups. In [19], proportional fairness (PF) was applied through an exhaustive search to allocate groups to the RBs. Groups of size two were investigated in [8] by invoking the Hungarian algorithm, and in [20] through PF for power allocation.

The above studies did not explore the stochastic nature of the wireless communication environment, and the channel coefficients for the different users were assumed to be known. Therefore, the optimization results based on the known channel coefficients were valid only when the coefficients were not far from reality. In practice, channel coefficients may vary along time rapidly, and channel sounding may not be frequent enough to capture the instantaneous change of mobile radio channels in the stochastic environment. In such situations, appropriate channel coefficients are to be used as the basis for user grouping and power allocation.

2.2 Learning Automata and Object Migration Automata

An LA[2] is a decision-making algorithm that can sequentially learn, the optimal action from among a set of actions, in a stochastic environment. At each time instant, an action is chosen by the LA, and serves as the input to the environment. The environment then responds to the action chosen by the LA, by a feedback that is usually a reward or a penalty to the action. Based on both the response of the environment and the current state of the LA, the LA adjusts its action selection strategy for future interactions. Initial LA was designed in [21] with a fixed structure, where the state update and the decision functions were time-invariant. Later, Variable Structure Stochastic Automata (VSSA) were developed, such as the linear reward-penalty scheme, the linear reward-inaction scheme, the linear inaction-penalty scheme, and the linear reward-penalty scheme [22], [23]. Schemes that apply nonlinear functions have also been designed and analyzed [22], [23], [24], where the updating functions can either be of continuous or discretized [25], [26]. In addition, the Markovian representation of the states in LA can be either absorbing or ergodic [22], [27], where the latter adapts better to non-stationary environments where the reward probabilities are time-variant. The state-of-the-art of the field of LA, is reported in [28], [29].

LA can be applied to solve many different problems, including OPPs in a random environment. The OPP is, in general, NP-hard, and a special case of the OPP, in which the number of objects in each group is equal, is the EPP. The benchmark solutions for the EPP have involved the classic field of LA [30], namely, the OMA and its variations. A comprehensive review of the previously proposed solutions for the OPP/EPP can be found in [6].

As the wireless communication system operates in a stochastic environment, and the NOMA technology involves multiple user groupings for SIC, it is natural to apply the most recent RL-based solution for such an application in order to improve the system's performance in the stochastic environment. In this study, we confirm the potential of applying the OMA in optimizing the

[2] This section can be moved to the Introduction (Section 1), if recommended by the Referees.

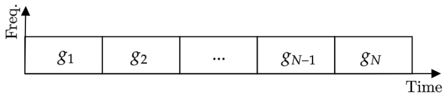

Fig. 1 Groups are assigned to orthogonal resources. The users within a group employ NOMA.

system's performance in the stochastic environment for the NOMA, which, to the best of our knowledge, has not been previously addressed in the literature.

3 System description

Consider a simplified single-carrier downlink cellular system that consists of one base station (BS) and K users that are to be divided into N groups for NOMA operation. 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. Each user can only be in one of the groups implying that $g_n \cap g_o = \emptyset$ 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 corresponding group. The simplified notation U_k is adopted to refer to a user, when the group of the user is trivial or undetermined. For example, if we have 4 users in the system, we could have user 1 and user 3 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 the notation U_1. Likewise, when we want to mention user 4 belonging to group 2, we apply $U_{2,4}$.

In this paper, we will consider the case of a single BS and with K users connected to that BS. NOMA is applied to each group, but different groups are assigned to orthogonal resources. One realization of such a communication scenario is shown in Fig. 1. In this scenario, the BS has assigned a frequency band to the K users. The users are to be grouped in N groups, each of which occupies a time slot. Here the orthogonal resource is the set of time slots, but it could just as well be other orthogonal resources, such as frequency bands. For mobility, the users are expected to move within a defined area. The users' behavior in a university or an office building are examples of where their behavior coincides with the mobility model utilized in this paper.

3.1 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 [31]. The parameters of the channel configuration will be detailed in the sections listing the numerical results. Note that our proposed LA solution can handle non-stationary stochastic processes, and so, it is distribution-independent. Therefore, the current Rayleigh distribution can be replaced by any other channel model, based on the application scenario and the environment.

3.2 Signal Model

In NOMA, each user in a group may suffer from both intra- and inter-group interference [3]. Intra-group interference refers to interference from other users within the same group, and inter-group interference refers to interference from users in other groups that employ the same band at the same time. In this study, we consider the case that different groups adopt orthogonal resources such that the inter-group interferences are non-existent.

Based on the NOMA protocol, 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_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 to $U_{n,k}$, the signal from all users other than $U_{n,k}$ in the same group, and the additive white Gaussian noise (AWGN), where $n_k \sim \mathcal{CN}(0, \sigma_k^2)$ [32]. The transmitted signal intended for $U_{n,k}$ and $U_{n,e}$ are given by s_k and $s_e \sim \mathcal{CN}(0,1)$ respectively. $p_{n,k}$ is the allocated power for $U_{n,k}$. Further, the total power budget for group g_n is given by P_n.

The BS' signals are decoded at the users through the SIC by using the channel coefficients in an ascending order [2]. As a result, through SIC, a user with a good channel quality can remove the interference from the users possessing poor channel qualities, while users with poor channel qualities decode their signals without applying SIC[3]. Hence, for user $U_{n,k}$, successful SIC is applied when the following requirement fulfills,

$$|h_{n,w}(t)|^2 \leq |h_{n,k}(t)|^2, \tag{2}$$

where w is the index of the users that have lower channel coefficients than user k in the user group g_n. Note that the channel coefficient for a certain user may be quite different in distinct frequency levels. In this study, similar to assumptions in [33] and [34], we assume that the ranking of channel coefficients along time, on average, keep the same for all the users. Indeed, the differences of channel coefficients are due to the various distances among the users to the BS, which makes the assumption in this paper valid. Eq. (2) implies that $U_{n,k}$

[3] We assume that differences between the time average of channel coefficients are due to the distinct geolocations of the users. Although the channel coefficient for a user may vary in different frequency bands, it is assumed that the ranking of the time averages of the coefficients among the various users maintains the same order in the different bands due to their distinct geolocations. With this assumption, the heterogeneity in different frequency bands will not influence the results of the user grouping.

is able to remove the messages to the users with lower channel coefficients in its group via SIC and then retrieve its own message. Hence, $U_{n,k}$ considers users with higher channel coefficients in its group as interference in the decoding process.

4 Problem Formulation

In this section, we formulate the problem to be solved. The first problem, which we refer to as the main problem, is formulated when instantaneous channel coefficients, say, at time t_0, are considered for grouping and power allocation. Here time t_0 is the time instant when the channel sounding is employed for the current round of user grouping and power allocation. Ideally, if the optimization of user grouping and power allocation is quick enough, and if the packet is short enough, the channel coefficients can be considered as a constant. In addition, if it is possible to re-group users and allocate power to them more often than the changes of channel coefficients, the system becomes adaptive and can thus be operated always in an optimized manner.

In reality, though, by studying the complexity of the main problem, we show that it is computationally very costly to solve. Therefore, it is not practical for the BS to follow the instantaneous channel condition as the basis for user grouping. For this reason, the main problem is divided into two sub-problems, where in the first sub-problem, the user channels are clustered based on the time averages of the coefficients, and subsequently, in the second sub-problem, the power allocation is considered.

4.1 Problem Formulation of Overall System

The objective of the main problem is to maximize the overall data rate given channel coefficients $h_k(t_0)$ and power budget. To take advantage of NOMA, the K users need to be divided into N groups, and for each group, the users share the same resource block (in time and frequency). Additionally, power allocation is to be optimized according to the channel conditions for each user in that group. To ease in the formulation, without loss of generality, we assume that the channel coefficients are sorted in ascending order, i.e., $h_1(t_0) \leq h_2(t_0) \leq \ldots \leq h_K(t_0)$[4].

Following the description given in [3], for the n-th group, after deployment of SIC, the SINR (signal-to-interference-plus-noise ratio) of user k in g_n is given by

$$\mathit{\Gamma}_{n,k}(t_0) = \frac{p_{n,k}|h_{n,k}(t_0)|^2}{I_{n,k}(t_0) + \sigma^2}, \tag{3}$$

[4] This assumption applies only in the problem formulation with instantaneous channel coefficients at t_0. Understandably, the channel coefficients will change along time due to the stochastic behavior, and the ranking of instantaneous channel coefficients belonging to different users may change from time to time due to channel fading.

where $p_{n,k}$ is the power allocated to $U_{n,k}$. Clearly, it is true that for any group n, $\sum_{j,\forall U_j \in g_n} p_{n,j} \leq P_n$ holds, where P_n denotes the power budget of g_n, with σ^2 denoting the Gaussian noise power. Parameter $I_{n,k}(t)$ represents the intra-group interference from other users to $U_{n,k}$, as

$$I_{n,k}(t_0) = \sum_{\substack{j, \\ \forall j>k,\ \{U_j,\ U_k\} \in g_n}} |h_k(t_0)|^2 p_{n,j}. \tag{4}$$

Given that the SIC requirement is fulfilled, the achievable data rate of user k is

$$R_{n,k}(t_0) = b \log_2 \left(1 + \Gamma_{n,k}(t_0)\right), \tag{5}$$

where b is the bandwidth of the channel. Therefore, the total achievable data rate for the system is expressed as

$$R(t_0) = \sum_{k=1}^{K} b \log_2 \left(1 + \Gamma_{n,k}(t_0)\right). \tag{6}$$

Based on the above analysis, we can formulate the optimization problem as follows.

$$\begin{aligned}
\max_{\{g_n\},\{p_{n,k}\}} \quad & R(t_0) && \text{(7a)} \\
\text{s.t.} \quad & g_n \cap g_o = \emptyset, \quad n \neq o, \quad n, o \in \mathcal{N} && \text{(7b)} \\
& \sum_{j,\forall U_j \in g_n} p_{n,j} \leq P_n, \quad \forall n \in \mathcal{N}, && \text{(7c)} \\
& R_{n,k}(t_0) \geq R_{QoS}, \quad k \in \mathcal{K}, && \text{(7d)} \\
& h_i(t_0) > h_j(t_0), \quad \forall i > j, \quad i, j \in \mathcal{K}. && \text{(7e)}
\end{aligned}$$

We now explain the significance of each of the above equations. The constraints in (7b) say that each user can only be part of one group. In (7c), we state that the sum of powers for a certain group needs to be less than or equal to P_n, which guarantees that the total power of the group is within the power constraint. The constraints in (7d) concern the demands on the Quality of Service (QoS) for each user. Hence, the data rate of a user needs to be above a specified required value, R_{QoS}, ensuring the QoS to all users, and addressing the fairness issue [32]. Finally, Eq. (7e) addresses the SIC requirement.

4.2 Complexity Considerations

We shall now consider the complexity of the associated problem. Considering the users and their placement into different groups, the minimum number of combinations possible is a Bell number, without even reckoning with the task of power allocation, where the Bell number gives the number of possible partitions of a set. In our case, we have B_K options where B_K is the K-th Bell number. Our task is to partition K users into κ non-empty sets. Considering that κ, without any additional constraints, can range from 1 to K, the consecutive Bell number for K is given by

$$B_K = \sum_{\kappa=1}^{K} \left\{ {K \atop \kappa} \right\}, \tag{8}$$

where $\left\{ {K \atop \kappa} \right\}$ is the Stirling numbers of the second kind [35]. Consequently, it follows that

$$\left(\frac{K}{e \ln K}\right)^K < B_K < \left(\frac{K}{e^{1-\lambda} \ln K}\right)^K, \tag{9}$$

which is exponential with regard to K [35], and $\lambda > 0$. Because the number of possible combinations to solve such a maximization problem (as stated in (7)) increases drastically with the number of users, and since the power allocation problem is non-convex [3], finding an optimal solution to the problem, based on instantaneous $h_k(t_0)$, is computationally hard. Further, when the system is to be adaptive to the changes in the environment, computations are to be carried out very frequently. Therefore, it is more practical to aim for a compromised solution, where the problem is divided into two sub-problems.

Specifically, in the first problem, we cluster the users into categories based on the time averages of the channel coefficients, and in the second step, we group the users based on the obtained categories, and proceed to solve the power allocation phase. The rationale behind such a computational paradigm is that we capture the long time average of the channel coefficients for the purpose of *grouping*, and thereafter, the power allocation is based on the available instantaneous values of h, or a time average computed over a certain number of channel sounding samples of h. Thus, as the grouping is considered more costly than the power allocation, by doing the grouping based on the mean, and being able to do power allocation more often at a minimal cost once the groups have been established, the computational effort is kept low.

Since there is no known solution for the general OPP, we further simplify the model for the solution, to consider the equi-partitioning of the users, namely the EPP. When all the groups are of equal sizes, we have the combination number W as:

$$W = \frac{K!}{\left(\frac{K}{N}!\right)^N N!}, \tag{10}$$

where $\frac{K}{N}$ is an integer. As a result of the above, we observe that the partitioning problems are still characterized by a combinatorial issue, but the number is significantly smaller than the Bell numbers. Note that the problem is still more complicated than just finding an optimal partitioning once and for all, because the environment changes stochastically.

4.3 Problem Formulation of Clustering

For NOMA, the channel coefficients of the users in a group need to be as different as possible to attain to a successful NOMA operation. To group the users with different coefficients, we first want to cluster the users with similar channel coefficients, and then select a single user from each cluster in order to formulate the groups. In other words, we want to cluster similar users together first, and then group them by selecting one user from each cluster. The first problem is the clustering problem, which is formulated in this subsection. The problem for user grouping, together with power allocation, is formulated in the next subsection.

The criterion for user clustering involves the *time averages* of the channel coefficients, denoted by $\overline{h_k(t)}$. In other words, the users that have similar average channel coefficients will be clustered together. The reason behind this is that the task of grouping the users is relatively costly in terms of computation, and it is, further, not cost-efficient to apply it based on the channel's instantaneous values. If we cluster the users according to the time averages, we can efficiently reduce the computational cost, and at the same time, capture the advantages of statistically employing NOMA.

In this study, as mentioned, we consider clustering users to groups of the same size. Therefore the number of the clusters is $L_c = \frac{K}{N}$, where L_c and N are integers[5]. Let q_c be the set of users in cluster c, where $c \in \mathcal{C} = \{1, \dots, L_c\}$ is the index of the set of clusters. Clearly, for the clustering problem, the differences between the coefficients in each cluster needs to be minimized, and the problem can be formulated as

$$\min_{\{\varphi_{c,k}\}} \quad \sum_{c=1}^{L_c} \sum_{k=1}^{K} \varphi_{c,k} |\overline{h_k(t)} - E_c|, \tag{11a}$$

$$\text{s.t.} \quad \sum_{c=1}^{L_c} \sum_{k=1}^{K} \varphi_{c,k} = K, \quad c \in \mathcal{C}, k \in K, \tag{11b}$$

$$\sum_{k=1}^{K} \varphi_{c,k} = N, \quad \forall c, \tag{11c}$$

where $\varphi_{c,k}$ is an indicator showing the relationship between the users and the clusters, and is given by

$$\varphi_{c,k} = \begin{cases} 1, \text{when } U_k \text{ belongs to cluster } c. \\ 0, \text{otherwise.} \end{cases} \tag{12}$$

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, where the objective function is stated in Eq. (11a). Specifically, for all the clusters, we want to minimize the difference of the channel coefficients between the users inside each of them. Eqs. (11b) and (11c) give a description of the variable $\varphi_{c,k}$ for user k in c. Hence, the sum of the variable $\varphi_{c,k}$ needs to be equal to the number of users, implying that all the users need to be a part of one 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, can be re-arranged in an $L_c \times N$ matrix, as

$$\begin{array}{c} \\ 1 \\ 2 \\ \cdots \\ L_c \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & \dots & N \end{array} \\ \begin{bmatrix} U'_{1,1,k} & U'_{1,2,k} & \cdots & U'_{1,N,k} \\ U'_{2,1,k} & U'_{2,2,k} & \cdots & U'_{2,N,k} \\ \cdots & \cdots & \cdots & \cdots \\ U'_{L_c,1,k} & U'_{L_c,2,k} & \cdots & U'_{L_c,N,k} \end{bmatrix} \end{array}, \tag{13}$$

[5] In reality, if L_c is not an integer, we can add dummy users to the system to satisfy the requirement. *Dummy users* are virtual users that are not part of the real network scenario, but are needed for constituting an equal size for all the clusters.

indicating L_c clusters with N users in each cluster. The user in the matrix is indexed by $U'_{c,g,k}$, where c is the index of the cluster, and g is the index of a user in a certain group, while k indicates the user (the position of a user k, in the matrix, is an independent term). Note that $U_{n,k}$ is different from $U'_{c,g,k}$ as the indexes are different.

For the next step, the problem is then to group users together by selecting one user from each cluster, and to then allocate power to them in order to achieve the maximized sum of data rate.

4.4 Problem Formulation of Power Allocation

As the grouping and power allocation tasks are inter-twined, we must consider these two aspects jointly when we study the maximization of the data rate. Although from the first step, we have obtained information on which users have similar channel coefficients, we observe that the power allocation problem remains unresolved. Therefore, we need to consider the power allocation of the users inside each cluster so as to be able to solve both the grouping and power allocation in NOMA.

Clearly, from the output of the clustering, we know which users are similar, and when we take one user from each cluster and construct N groups, the size of each group will be L_c. Without loss of generality, we can assume that the average channel coefficients are sorted in ascending order, i.e., $\overline{h_1(t)} \leq \overline{h_2(t)} \leq \ldots \leq \overline{h_K(t)}$ (similar to [33] and [34]). If we now consider user grouping and power allocation based on the average channel coefficients, the problem can be formulated as

$$\max_{\{g_n\},\{P_n\}} \quad R \tag{14a}$$

$$\text{s.t.} \quad g_n \cap g_o = \emptyset, \quad n \neq o, \quad n, o \in \mathcal{N}, \tag{14b}$$

$$\sum_{j,\forall U_j \in g_n} p_{n,j} \leq P_n, \quad n \in \mathcal{N}, \tag{14c}$$

$$R_{n,j}(t) \geq R_{QoS}, \quad j \in \mathcal{K}, n \in \mathcal{N}, \tag{14d}$$

$$\overline{h_i(t)} > \overline{h_j(t)}, \quad \forall i > j, \quad i, j \in \mathcal{K}, \tag{14e}$$

$$|g_n \cap q_c| = 1, \quad \forall c, \ \forall n, \tag{14f}$$

$$\sum_{j,\forall U_j \in g_n} \tau_{n,j} = L_c, \quad \forall n, \tag{14g}$$

$$\sum_{n,\forall n \in \mathcal{N}} \sum_{j,\forall U_j \in g_n} \tau_{n,j} = N L_c. \tag{14h}$$

In Eq. (14a), the parameter R is calculated based on Eq. (6) when $h_k(t)$ is replaced by $\overline{h_k(t)}$, indicating that this rate is based on the average channel coefficients. Further, in (14b), we state that the groups need to be disjoint. Hence, any user can only be in a single group. In Eq. (14c), we address the constraint for the power budget. The QoS constraint is given in (14d), which ensures the fairness among the users. The SIC constraint is given in Eq. (14e). The constraint in Eq. (14f) specifies that only a single user is selected to formulate a group from each cluster. Finally, in Eq. (14g), we introduce an indicator $\tau_{n,k}$, stating whether U_k is in group n, as

$$\tau_{n,k} = \begin{cases} 1, \text{when } U_k \text{ belongs to group } n. \\ 0, \text{otherwise}. \end{cases} \tag{15}$$

This constraint states that each group has L_c users. Furthermore, all users should belong to a certain group, which is given in Eq. (14h).

The solution to this problem will provide the grouping of users along with their corresponding power allocations. Note that the power allocation is calculated based on the average of the channel coefficients. When communication happens, these coefficients may be different. Thus, the power allocation can be done for time averages of the channel sounding, or for instantaneous values.

Comparing the problem in Eq. (7) with the sub-problems in Eqs. (11) and (14), we record the following differences. (a) In Eq. (7), the instantaneous channel coefficients are followed for grouping and optimization. However, in Eqs. (11) and (14), the clustering and grouping of users are based on the respective time-averaged values. (b) In Eq. (7), the groups of users may have different sizes, while in Eqs. (11) and (14), the sizes of all the groups are equal. Indeed, since following the instantaneous channel coefficients is both costly and impractical, the task of following the average values becomes a reasonable and feasible foundation for the grouping. Considering that the equi-partitioning of the users reduces the complexity of the original problem, practical and adaptive solutions based on RL algorithms can be proposed. In the next section, we will propose a two-step solution corresponding to the sub-problems, based on the adaptive Tabula rasa RL paradigm.

5 Solution to User Grouping and Power Allocation

The problem of grouping and power allocation in NOMA systems is two-pronged. Therefore, in Section 5.1, we only consider the first issue of the two, namely the clustering and the grouping of the users. We will show that our solution can handle the stochastic nature of the channel coefficients of the users, while it is also able to follow changes in their channel behaviors over time, ensuring that the system can prudently follow the nature of the channels that are similar to what we will expect in a real system. More specifically, we will categorize users into clusters based on similar channel coefficients for long-term fading. Because the values of h for different users are stochastic, we need a method for capturing the long-time average of the channels. Therefore, we exploit a ML algorithm from within the OMA family for clustering them. Specifically, we utilize the Enhanced Object Migration Automata (EOMA) to capture the users' similarities, for the purposes of categorization. Thereafter, as mentioned above, the users are grouped by taking a single user from each cluster so that users within any one group have distinct channel coefficients. Once the groups have been established in Section 5.1, we can utilize these groups to allocate power among them either instantaneously or for a time interval using heuristic-based solutions.

5.1 Clustering Through EOMA

To solve problems in stochastic environments, the field of LA has shown itself to be a powerful tool because fast and accurate convergence can be achieved while the computational complexity remains low [36]. OMA algorithms are part of the LA paradigm, and can solve partitioning problems by having LA instances cooperate to find the best partitioning [37]. With some insight, we see that the family of OMA algorithms are based on Tabula rasa RL. Without any prior knowledge of the system parameters, channels, or clusters in our case, the OMA self-learns by observing the environment that it interacts with, over time. 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 improving incrementally through each interaction with the environment, the OMA algorithms prove themselves to be compelling mechanisms for solving complex and stochastic problems. The OMA algorithms attempt to learn the "true partitioning" in different grouping scenarios, based on the elements from the respective groups that are accessed together. The "true partitioning" that occurs in nature is always unknown, but assuming that the true partitioning consists of equi-sized clusters, the OMA algorithms can find these with a high accuracy [6], [7], [38].

In the OMA, the users of our system need to be represented as abstract objects. Therefore, in OMA terms, the users are called objects. The OMA algorithms require a number of states per action, indicated by S. An action in LA is a solution that the algorithm can converge to. In our system, the actions are the different clusters that objects can be assigned to. Hence, based on the current state of an object, we know that object's action, which is precisely its current cluster in the system. Therefore, each object, or user in our case, has a given state indicated by $\epsilon_k = \{1, 2, ..., SL_c\}$, 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 Lc 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 $\Theta_k \in \mathcal{C} = \{q_1, q_2, \ldots, q_{L_c}\}$.

The states are the foundational memory components of the OMA algorithms, and the objects are moved in and out of states as they are penalized or rewarded in the RL process. In this paper, we utilize the EOMA variant of OMA, where each object has an equal number of possible states. We say that the algorithm has converged when all the objects have reached the two innermost states of an action. When convergence has been attained, we reckon that the solution that the EOMA algorithm has found, is sufficiently accurate.

The requirement for convergence can be tuned through the number of states introduced to the system. Consequently, introducing a deeper state space increases the solution's accuracy of finding the "true partitioning". However, the time before convergence is achieved, also increases with the number of states. Therefore, the state depth, given by S, is a trade-off between time and accuracy. In Fig. 2, we describe the state numbering of any two actions. By way of example, consider a scenario with three possible clusters: the first cluster of the EOMA will have the state numbering from 1 to S, where the innermost state is 1, and the second innermost state is 2, while the boundary state is S. For the second cluster, it will have 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 whilst $3S$ for the boundary state.

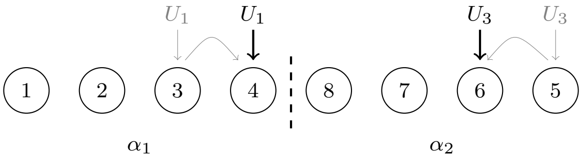

Fig. 2 An example of the states of the EOMA, for the updates for the penalty for U_1 and U_3 when they have the same ranking category.

Algorithm 1 Clustering of Users

```
Require: $\overline{h_k(t)}$ for all users $K$ over $\Delta_t$
  while not converged do {Convergence is reached when all users are in the two most internal states of any action}
    for all $K$ do
      Rank the users from 1 to $K$ {Index 1 is given to the user with lowest $\overline{h_k(t)}$ ($K$ 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}
```

Algorithm 1 gives the overall operation for the clustering of the users. The functionalities on receiving a reward or a penalty, as the EOMA is interacting with the NOMA system, are given in Algorithms 2 and 3. In the algorithms, we consider the operation in relation to a pair of two users U_a and U_b ($Q = (U_a, U_b)$). The EOMA considers users in pairs (called *queries*, denoted by Q). Through information about their pairwise rankings, we work towards a clustering of the users into the different channel categories. For each time instant $\varDelta_t$, the BS obtains values of $h_k(t)$ through channel sounding, and we use the average of the samples of $\varDelta_t$ as input to the EOMA $(\overline{h_k(t)})$. Note that when $\varDelta_t = 1$, the instantaneous values are utilized as the input.

The BS proceeds to rank the users, indicated by $r_k = \{1, 2, ..., K\}$, where each U_k is given a single value of r_k for each $\varDelta t$. For the ranks, $r_k = 1$ is given to the user that has the lowest channel coefficient compared to the total number of users; $r_k = K$ is given to the user with the highest channel coefficient of the users, and 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 $\varUpsilon_k$ for U_k, where $\varUpsilon_k = \{r \in [1, N] = 1, r \in [1 + N, 2N] = 2, r \in [1 + 2N, 3N] = 3, \ldots, r \in [K - N + 1, K] = L_c\}$. In this way, even if the users have similar channel conditions, they will be compared, and the solution can work on determining 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 precisely, if two users have the same ranking categories, they are sent as a query to the EOMA and the LA is rewarded if these two users are clustered together, and penalized if they are not.

Algorithm 2 Process Reward

```
Require: $Q = (U_a, U_b)$ {A query ($Q$), consisting of $U_a$ and $U_b$}
Require: The state of $U_a$ ($\epsilon_a$) and $U_b$ ($\epsilon_b$)
  if $\epsilon_a \mod S \neq 1$ then {$U_a$ not in innermost state}
    $\epsilon_a = \epsilon_a - 1$ {Move $U_a$ towards innermost state}
  end if
  if $\epsilon_b \mod S \neq 1$ then {$U_b$ not in innermost state}
    $\epsilon_b = \epsilon_b - 1$ {Move $U_b$ towards innermost state}
  end if
  return The next states of $U_a$ and $U_b$
```

As an example, let us consider four users in a NOMA communications scenario for $\varDelta_t = 5$. The users should be grouped into two groups. Therefore, we need to categorize them into two clusters based on their channel conditions: one with weak channel conditions (ground truth in this example: $\varUpsilon_1 = 1$ and $\varUpsilon_2 = 1$) and the other with strong channel conditions (ground truth in this example: $\varUpsilon_3 = 2$ and $\varUpsilon_4 = 2$). First, when $h_k(5)$ for the different users are obtained, we rank them according to $\overline{h_k(5)}$. Then, we consider the arbitrary pair $Q = (U_1, U_3)$, ranked $r_1 = 2$ giving $\varUpsilon_1 = 1$ and $r_3 = 2$ giving $\varUpsilon_3 = 1$ ($r = \{3, 4\}$ resulting in $\varUpsilon = 2$). Additionally, their current states are $\epsilon_1 = 3$ and $\epsilon_3 = 5$. For this scenario the state depth for each action is four, meaning that we have 8 states in total ($SL_c = 8$). Following the descriptions given in

Algorithm 3 Process Penalty

```
Require: $Q = (U_a, U_b)$ {A query ($Q$), consisting of $U_a$ and $U_b$}
Require: The state of $U_a$ ($\epsilon_a$) and $U_b$ ($\epsilon_b$)
  if $\epsilon_a \mod S \neq 0$ and $\epsilon_b \mod S \neq 0$ then {Neither of the users are in boundary states}
    $\epsilon_a = \epsilon_a + 1$ {Move $U_a$ towards boundary state}
    $\epsilon_b = \epsilon_b + 1$ {Move $U_b$ towards boundary state}
  else if $\epsilon_a \mod S \neq 0$ and $\epsilon_b \mod S = 0$ then {$U_b$ in boundary state and $U_a$ not in boundary state}
    $\epsilon_a = \epsilon_a + 1$
    $temp = \epsilon_b$
    $x$ = unaccessed user in cluster of $U_a$ closest to boundary state
    $\epsilon_x = temp$
    $\epsilon_b = \epsilon_a$
  else if $\epsilon_b \mod S \neq 0$ and $\epsilon_a \mod S = 0$ then {$U_a$ in boundary state and $U_b$ not in boundary state}
    $\epsilon_b = \epsilon_b + 1$
    $temp = \epsilon_a$
    $x$ = unaccessed user in cluster of $U_b$ closest to boundary state
    $\epsilon_x = temp$
    $\epsilon_a = \epsilon_b$
  else {Both users are in boundary states}
    $\epsilon_y = \epsilon_{\{a\ or\ b\}}$ {$y$ equals $a$ or $b$ with equal probability}
    $temp = \epsilon_y$
    $x$ = unaccessed user in cluster of $U_{\not y}$ closest to boundary state
    $\epsilon_x = temp$
    $\epsilon_y = \epsilon_{\not y}$ {Move chosen user ($y$) to cluster of $\not y$}
  end if
  return The next states of $U_a$ and $U_b$
```

[6], [7], [39], or the same concept that we observe in Algorithms 2 and 3, we understand that the objects are currently not clustered together. Therefore, we will penalize them according to Algorithm 3. A visualization of the example is depicted in Fig. 2.

Algorithm 4 Get Groups

```
Require: The users and information about their cluster obtained by the EOMA
  for all clusters in $\mathcal{C}$ do
    Rank the users from 1 to $Lc$ based on $\overline{h(t)}$ {$r = 1$ to the user with lowest value ($r = Lc$ to the highest)}
  end for
  for Number of groups ($N$) do
    for all $r$ do
      for all clusters in $\mathcal{C}$ do {Each group will consist of one user from each cluster with the same rank}
        $g_n$ = One user from each cluster with rank $r$
      end for
    end for
  end for
  return The users' groups
```

When the algorithm has converged, the users in distinct actions constitute different clusters. We will then invoke Algorithm 4, to obtain the groups that are needed for the power allocation, where the users are selected into groups based on their ranking of $\overline{h_k(t)}$ within each cluster. Again we use the ranking information of the users, where users with the same rank in each cluster are put together. Thus, all the users with the same rank in each of their respective clusters, will form a group.

5.2 Power Allocation via Heuristic-based Solutions

Once the grouping of the users has been established, we can allocate power to the different users in such a way that the joint data rate (R) is maximized. For a group with $\frac{K}{N} = L_c$ users and power budget P_n, the problem can be expressed by:

$$\max \sum_{k=1}^{K} b \log_2 \left(1 + \mathit{\Gamma}_{n,k}\right), \tag{16a}$$

$$\text{s.t.} \ \sum_{k=1}^{K} p_{n,k} \leq P_n, \quad \forall n, n \in \mathcal{N}, \tag{16b}$$

$$0 \leq p_{n,k}, \quad \forall n, n \in \mathcal{N}, \forall k, k \in \mathcal{K}, \tag{16c}$$

$$R_{Qos} \leq R_k, \quad \forall k, k \in \mathcal{K}, \tag{16d}$$

where $\mathit{\Gamma}_{n,k} = \frac{p_{n,k}|h_{n,k}|^2}{|h_{n,k}|^2(P_n - \sum_{\forall i \leq k} p_{n,i}) + \sigma^2}$. Our goal is to determine the power to the different users within each group so as to maximize the total data rate of all the groups.

Note the the objective function for the optimization may be changed to also work with other functions, such as that of maximizing the minimum data rate within a group. There are also numerous ways of power allocation in

Algorithm 5 Greedy solution for the power allocation

```
Require: $h_{n,k}$ for all users $K$ {Requires the value of $h_{n,k}$ for $t$ $\left(h_{n,k}(t)\right)$ or $\Delta_t$ $\left(\overline{h_{n,k}(\Delta_t)}\right)$}
Require: $R_{QoS}$ {The minimum required data rate}
Require: $\mathcal{G}$ {The groups established in Algorithm 4}
  for all $g_n$, in $\mathcal{G}$ do
    for all users, $i$ from 1 to the size of $g_n$ do {Ordered, where user 1 has the lowest $h$ ($L_c$ has the highest $h$)}
      Solve for $p_{n,i}$ using $R_{QoS} = B \log_2 \left(1 + \frac{p_{n,i}|h_{n,i}|^2}{|h_{n,i}|^2(P_n - \sum_{\forall k \leq i} p_{n,k}) + \sigma^2}\right)$ {The feasibility check}
    end for
    if $P_n - \sum_i p_{n,i} \geq 0$ then
      $p_{n,Lc} = P_n - \sum_{j, \forall p_{n,j} \in g_n \setminus p_{n,L_c}} p_{n,j}$ {The problem is feasible, and we give the remaining power to the strongest user}
    end if
  end for
```

Algorithm 6 Channel-coefficient based solution for power allocation

```
Require: $h_{n,k}$ for all users $K$ {Requires the value of $h_{n,k}$ for $t$ $\left(h_{n,k}(t)\right)$ or $\Delta_t$ $\left(\overline{h_{n,k}(\Delta_t)}\right)$}
Require: $R_{QoS}$ {The minimum required data rate}
Require: $\mathcal{G}$ {The groups established in Algorithm 4}
  for all $g_n$, in $\mathcal{G}$ do
    for all users, $i$, in $g_n$ do {Ordered, where user 1 has the lowest $h$ ($L_c$ has the highest $h$)}
      if $i$ is 1 then {For the weakest user}
        $R_{n,1} = R_{QoS}$ {Data rate of weakest user}
        Solve for $p_{n,1}$ using $R_{n,1} = B\log_2\left(1 + \frac{p_{n,1}|h_{n,1}|^2}{|h_{n,1}|^2(P_n - p_{n,1}) + \sigma^2}\right)$
      else if $i$ is $L_c$ then {For the strongest user}
        $p_{n,Lc} = P_n - \sum_{j,\forall p_{n,j} \in g_n \setminus p_{n,L_c}} p_{n,j}$ {We give the remaining power to the strongest user}
      else {For the user between the weakest and strongest user}
        $\Gamma_{n,i} = \frac{p_{n,i}|h_{n,i}|^2}{|h_{n,i}|^2(P_n - \sum_{\forall k \leq i} p_{n,k}) + \sigma^2}$
        Solve for $p_{n,i}$ using $\frac{\Gamma_{n,i}}{\Gamma_{n,i-1}} = \frac{|h_{n,i}|^2}{|h_{n,i-1}|^2}$ {SINR of user $i$ is based on a relation between $U_i$ and $U_{i-1}$}
      end if
    end for
  end for
  return The power for the different users in the different groups
```

various communication scenarios [40],[32]. The power allocation schemes can be replaced by any other algorithm and will not change the nature of the RL procedure. However, in this paper, we will consider two heuristic-based algorithms, namely, the greedy algorithm and the channel coefficient based algorithm for maximizing the sum rate.

For the greedy solution, we allocate to the users with limited power to just fulfill the minimum required data rate, and give the remaining power to the user with the best channel coefficient, as presented in Algorithm 5. Given Eqs. (5) and (6), allocating the majority of power to the users with higher values of h 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. Nevertheless, the weak users' required data rate is ensured, and can be adjusted to the given scenario. The greedy solution can also be used for checking the feasibility. When all users are given power values that are just sufficient to fulfill the QoS requirement, and if the total power is sufficient, we deem the solution obtained as being "feasible".

The main drawback of the greedy solution is that the data rates among the users may be highly unbalanced. To mitigate this drawback, we propose another solution based on a relation between the values of $|h|^2$ of the different users, when they are a part of an established group. This solution is depicted in Algorithm 6. As observed in the algorithm, we base a linear algebraic system on the SINR (Eq. (3)), intra-group interference (Eq. (4)) and the data rate (Eq. (5)) for optimizing the sum rate of the system given by Eq. (6). Firstly, the data rate of the user with the weakest h is ensured by setting its data rate equal to R_{QoS}. Once we know the data rate of the weakest user, we can calculate the power that needs to be given to that particular user. Consequently, we then find the power for the users between the weakest and strongest users for the given group (based on $h_{n,i}$), where we use the SINR of the previous user times the relation between h-values for the previous user and the user for whom we are finding the SINR. Once we have the SINR for the user in-between the weakest and strongest, we can calculate its needed power based on Eq. (3). The strongest user is then given the remaining power by subtracting the power allocated for the other users from the total power budget of its group. The process is repeated for all the groups.

6 Numerical Results

For the experiments reported here, we used MatLab for simulating the values of h. Additionally, a Python script was utilized 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 $5.7GHz$ and an underlying Rayleigh distribution for the corresponding values of $h(t)$. For mobility in our model, we utilized a moving pace corresponding to the movement inside an office building. Thus, we assumed a mobility factor of $2\,\frac{km}{h} = v_U$ for the users' receivers. We sampled the values of h according to $\frac{1}{2f_d}$, where f_d is the Doppler frequency, where the latter is expressed as $f_d = f_c(\frac{v_U}{v_L})$, and v_L is the speed of light.

In the figures given below, we use "Sample Number" (Δ_t) as the notation on the X-axis. The numerical results for the sub-problems will be presented separately in Sections 6.1 and 6.2.

6.1 Results for Grouping

For evaluating the simulation of clustering and grouping, we base our accuracy on whether or not the LA found the clusters that correspond to the minimized difference between the users in a cluster, based on the users' given mean values of h in the simulations. Remarkably, if it is provided with such pairwise inputs, the EOMA yielded a 100% accuracy in which the learned clustering was identical to the unknown underlying clustering in every single run for the example provided with $-30dB$ difference between values of h, and for groups of size $4, 6, 8, 10, 12, 14, 16, 18$ and 20, where the number of users in a group was equal to two. The difference between the users can be replaced by any other equivalent condition, and these values are only generated for testing the solution. The reader should observe that in a real-life scenario, the "true partitioning" is always unknown. The number of iterations that it took for the EOMA to achieve 100% accuracy for the different numbers of users is depicted in Fig. 3. These results were based on h values that are shown in Fig. 4. In the

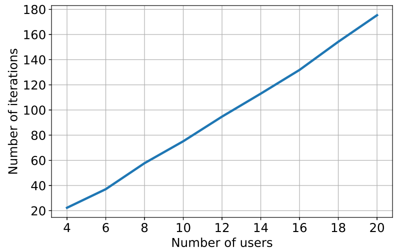

Fig. 3 Graph showing the average number of required iterations (Δ_t) before convergence is reached for different number of users, and groups of size two, based on the average of 100 independent experiments.

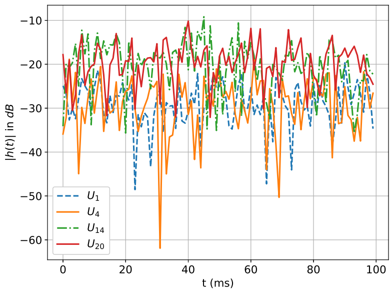

Fig. 4 Example of the simulated $|h(t)|$ for four different users in dB scale.

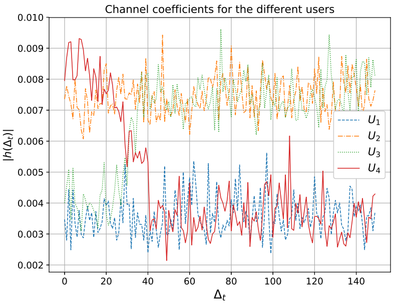

Fig. 5 Graph showing $|h(\Delta_t)|$ as a function of time Δ_t, where U_3 and U_4 changes distinctly around $40\Delta_t$.

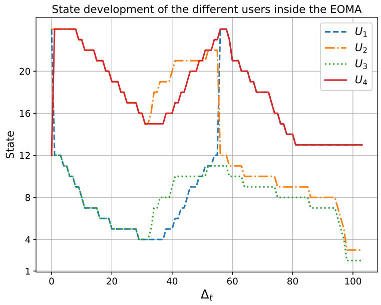

Fig. 6 The changes of states in the EOMA for different users over Δ_t, where the LA starts changing clusters around $40\Delta_t$.

interest of simplicity of presentation, the plot shown in Fig. 4 is for 4 users. For the case of 20 users, the lines become hard to distinguish, even though the principles used in the simulation of h are the same. Notably, the EOMA retains its accuracy as the number of users increases, and yields 100% accuracy both for four users as well as 20 users[6].

When we formulated the problem in Section 3, for conciseness, we assumed that the ranking of the average values of the channel coefficients were kept the same. In fact, the proposed EOMA algorithm can follow the changes adaptively even if the mean values vary along time. Here in Figs. 5 and 6, we demonstrate that the EOMA is able to follow the changes adaptively when the users' channel coefficients vary along time. As depicted in Fig. 5, a change in channel coefficients happens at around $\Delta_t \approx 40$. From Fig. 6, we can observe that the

[6] In these simulations, we used $S = 8$. The way that we obtained the solution's accuracy was in terms of whether or not the EOMA found the clusters that it should have found, based on the mean values of the different users in the NOMA system.

EOMA quickly detects the change even as early as around $\Delta_t \approx 40$, and updates its clustering after approximately 20 samples. More specifically, we can observe from Fig. 5 that the most similar users are initially $U1$ and $U3$, and $U2$ and $U4$, which change to $U1$ and $U4$, and $U2$ and $U3$ after around 40 samples. To show the updates of the states in the EOMA, the states are depicted as a function of time in Fig. 6. Initially, the objects are randomly located in boundary states (State 24 and State 12 in this figure). We can observe that the users with similar coefficients move to the same cluster after a few interactions, and they eventually move to deeper states. When the environment changes at around time instant 40, they move shallower instead of deeper. Eventually, the user clusters are updated in accordance to the new environment, and the states move deeper and deeper again. The LA converges once all the objects are in the innermost states (1 or 2, or 14 or 13), which we can observe from the end of the lines in the plot. It is important to note that, in this example, we have utilized a state depth of 12 in the EOMA. With a shallower state space, we could have followed the channels more instantaneously.

6.2 Results for Power Allocation

The simulation of the greedy solution and the channel-coefficient based solution to power allocation was done based on the groups established in the LA solution. For demonstrating the results of our approaches, we compared our solutions with those obtained by an exhaustive grid search. The exhaustive grid search was implemented in a step size of 0.001 ($\gamma = 0.001$). It was carried out for the same groups and the same values obtained from h through channel sounding as for the greedy and the channel-coefficient based solutions. We tested both the cases of doing power allocation based on instantaneous values, and on a time average ($\Delta t = 5$). In Fig. 7, we depict the results of the three approaches, for an average of $\Delta_t = 5$ samples of h.

As illustrated, the results of the greedy solution coincide with that of the exhaustive search, which means that giving more power to the user with strong channel coefficient, indeed, optimizes the sum rate of the system in the current configuration. The results of the channel-coefficient based solution have a better fairness among the users, where this is at a cost of attaining to a sub-optimal solution in terms of sum rate. The computations required for the greedy or the channel-coefficient based solution, depend on the number of users in each group, where $2L_c$ computations are needed for each group. By way of comparison, for L_c users in each group, we need to test $\left(\frac{P_n}{\gamma}\right)^{L_c}$ combinations for an exhaustive search.

Because the LA-based adaptive grouping solution accomplishes the partitioning of the users in favor of NOMA technology (i.e., users with distinct channel coefficients are grouped together), once the group is formulated, the objective of the power allocation may be changed to any other interesting form for NOMA, and thus different solutions can be further developed. In

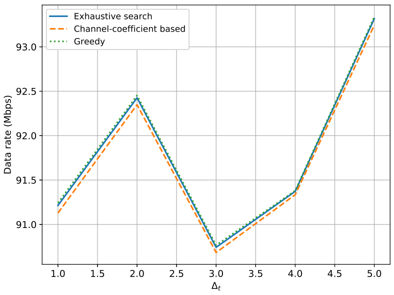

Fig. 7 Data rate for exhaustive, greedy and channel-based solution for groups of three users. Based on averages over 5 samples of $|h(t)|$.

other words, the objective function of power allocation is not constrained to be the one requiring "sum-rate maximization".

7 Conclusions

In this paper, we have proposed a novel solution to the user grouping and power allocation problems in NOMA systems, also taking into consideration the stochastic nature of the users' channel coefficients. The grouping has been achieved by using the tabula rasa RL technique specified by the EOMA, and the simulation results presented show that a 100% accuracy for finding clusters of similar $h(t)$ over time, can be obtained in a limited number of iterations. In addition, our solution is able to follow the changes of $h(t)$, which makes our solution for grouping adaptive to changes in users' channel conditions, and the corresponding changes for their group associations. For power allocation, we proposed two solutions for the sum rate maximization with a QoS constraint for users. Our two-step solution offers flexibility with regard to both the grouping and power allocation phases, and can be used as stand-alone components of a NOMA system.

Acknowledgements We are very grateful to the anonymous Referees of the previous version of the paper, who suggested various modifications and changes. Their suggestions have enhanced the quality of this present version.

References

1. Y. Liu, Z. Qin, M. Elkashlan, Z. Ding, A. Nallanathan, and L. Hanzo, "Nonorthogonal Multiple Access for 5G and Beyond," *Proceedings of the IEEE*, vol. 105, no. 12, pp. 2347–2381, Dec. 2017.
2. M. Pischella and D. Le Ruyet, "NOMA-Relevant Clustering and Resource Allocation for Proportional Fair Uplink Communications," *IEEE Wireless Commun. Lett.*, vol. 8, no. 3, pp. 873–876, Jun. 2019.

3. J. Cui, Z. Ding, P. Fan, and N. Al-Dhahir, “Unsupervised Machine Learning-Based User Clustering in Millimeter-Wave-NOMA Systems,” *IEEE Trans. Wireless Commun.*, vol. 17, no. 11, pp. 7425–7440, Nov. 2018.
4. S. Glimsdal and O. Granmo, “A Novel Bayesian Network Based Scheme for Finding the Optimal Solution to Stochastic Online Equi-Partitioning Problems,” in *13th International Conference on Machine Learning and Applications*, Dec. 2014, pp. 594–599.
5. W. Gale, S. Das, and C. T. Yu, “Improvements to an Algorithm for Equipartitioning,” *IEEE Trans. Comput.*, vol. 39, no. 5, pp. 706–710, May 1990.
6. A. Shirvani and B. J. Oommen, “On Invoking Transitivity to Enhance the Pursuit-Oriented Object Migration Automata,” *IEEE Access*, vol. 6, pp. 21 668–21 681, 2018.
7. ——, “On Utilizing the Pursuit Paradigm to Enhance the Deadlock-Preventing Object Migration Automaton,” in *International Conference on New Trends in Computing Sciences (ICTCS)*, Oct. 2017, pp. 295–302.
8. M. A. Sedaghat and R. R. Müller, “On User Pairing in Uplink NOMA,” *IEEE Trans. Wireless Commun.*, vol. 17, no. 5, pp. 3474–3486, May 2018.
9. S. Timotheou and I. Krikidis, “Fairness for Non-Orthogonal Multiple Access in 5g Systems,” *IEEE Signal Process. Lett.*, vol. 22, no. 10, pp. 1647–1651, 2015.
10. Y. Liu, Z. Qin, M. Elkashlan, Y. Gao, and L. Hanzo, “Enhancing the Physical Layer Security of Non-Orthogonal Multiple Access in Large-Scale Networks,” *IEEE Trans. Wireless Commun.*, vol. 16, no. 3, pp. 1656–1672, 2017.
11. Z. Ding, Z. Yang, P. Fan, and H. V. Poor, “On the Performance of Non-Orthogonal Multiple Access in 5G Systems with Randomly Deployed Users,” *IEEE Signal Process. Lett.*, vol. 21, no. 12, pp. 1501–1505, 2014.
12. N. Zhang, J. Wang, G. Kang, and Y. Liu, “Uplink Nonorthogonal Multiple Access in 5G Systems,” *IEEE Commun. Lett.*, vol. 20, no. 3, pp. 458–461, 2016.
13. Y. Liu, Z. Qin, M. Elkashlan, A. Nallanathan, and J. A. McCann, “Non-Orthogonal Multiple Access in Large-Scale Heterogeneous Networks,” *IEEE J. Sel. Areas Commun.”*, vol. 35, no. 12, pp. 2667–2680, 2017.
14. Z. Ding, P. Fan, and H. V. Poor, “Random Beamforming in Millimeter-Wave NOMA Networks,” *IEEE Access*, vol. 5, pp. 7667–7681, 2017.
15. J. Cui, Y. Liu, Z. Ding, P. Fan, and A. Nallanathan, “Optimal User Scheduling and Power Allocation for Millimeter Wave NOMA Systems,” *IEEE Trans. Wireless Commun.*, vol. 17, no. 3, pp. 1502–1517, 2017.
16. L. Zhu, J. Zhang, Z. Xiao, X. Cao, D. O. Wu, and X. Xia, “Joint Power Allocation and Beamforming for Non-Orthogonal Multiple Access (NOMA) in 5G Millimeter Wave Communications,” *IEEE Trans. Wireless Commun.*, vol. 17, no. 9, pp. 6177–6189, Sep. 2018.
17. Y. Liu, Z. Ding, M. Elkashlan, and H. V. Poor, “Cooperative Non-Orthogonal Multiple Access with Simultaneous Wireless Information and Power Transfer,” *IEEE J. Sel. Areas Commun.”*, vol. 34, no. 4, pp. 938–953, 2016.
18. Y. Yin, Y. Peng, M. Liu, J. Yang, and G. Gui, “Dynamic User Grouping Based NOMA Over Rayleigh Fading Channels,” *IEEE Access*, vol. 7, pp. 110 964–110 971, 2019.
19. X. Chen, A. Benjebbour, A. Li, and A. Harada, “Multi-User Proportional Fair Scheduling for Uplink Non-Orthogonal Multiple Access (NOMA),” in *IEEE 79th Vehicular Technology Conference (VTC Spring)*, May 2014, pp. 1–5.
20. F. Liu, P. Mähönen, and M. Petrova, “Proportional Fairness-Based User Pairing and Power Allocation for Non-Orthogonal Multiple Access,” in *IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Aug. 2015, pp. 1127–1131.
21. M. L. Tsetlin, “Finite Automata and Modeling the Simplest Forms of Behavior,” in *Mathematics in Science and Engineering*, 1973, vol. 102, pp. 3–83.
22. S. Lakshmivarahan, *Learning Algorithms: Theory and Applications.* New York: Springer, 1981.
23. K. S. Narendra and M. A. L. Thathachar, *Learning Automata: An Introduction.* Courier Corporation, May 2013.
24. S. Lakshmivarahan and M. A. L. Thathachar, “Absolutely Expedient Algorithms for Stochastic Automata,” *IEEE Trans. Syst. Man Cybern.*, vol. 3, pp. 281–286, 1973.

25. B. J. Oommen and M. Agache, “Continuous and Discretized Pursuit Learning Schemes: Various Algorithms and Their Comparison,” *IEEE Trans. Syst. Man Cybern.: B Cybern.*, vol. 31, no. 3, pp. 277–287, 2001.
26. X. Zhang, O.-C. Granmo, and B. J. Oommen, “Discretized Bayesian Pursuit - A New Scheme for Reinforcement Learning,” in *Proceedings of IEA-AIE 2012*, Dalian, China, Jun. 2012, pp. 784–793.
27. A. S. Poznyak and K. Najim, *Learning Automata and Stochastic Optimization.* Springer, 1997, vol. 3.
28. A. Yazidi, X. Zhang, L. Jiao, and B. J. Oommen, “The Hierarchical Continuous Pursuit Learning Automation: A Novel Scheme for Environments With Large Numbers of Actions,” *IEEE Trans. Neural. Netw. Learn. Syst.*, vol. 31, no. 2, pp. 512–526, 2020.
29. X. Zhang, L. Jiao, B. J. Oommen, and O. Granmo, “A Conclusive Analysis of the Finite-Time Behavior of the Discretized Pursuit Learning Automaton,” *IEEE Trans. Neural. Netw. Learn. Syst.*, vol. 31, no. 1, pp. 284–294, 2020.
30. A. Shirvani, “Novel Solutions and Applications of the Object Partitioning Problem,” Ph.D. dissertation, Carleton University, 2018.
31. M. Pätzold, *Mobile Radio Channels*, 2nd ed. Chichester: Wiley, 2012.
32. H. Xing, Y. Liu, A. Nallanathan, Z. Ding, and H. V. Poor, “Optimal Throughput Fairness Tradeoffs for Downlink Non-Orthogonal Multiple Access Over Fading Channels,” *IEEE Trans. Wireless Commun.*, vol. 17, no. 6, pp. 3556–3571, Jun. 2018.
33. J. Kang and I. Kim, “Optimal User Grouping for Downlink NOMA,” *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 724–727, Oct. 2018.
34. L. Zhu, J. Zhang, Z. Xiao, X. Cao, and D. O. Wu, “Optimal User Pairing for Downlink Non-orthogonal Multiple Access (NOMA),” *IEEE Wireless Commun. Lett.*, vol. 8, no. 2, pp. 328–331, Apr. 2019.
35. D. Berend and T. Tassa, “Improved Bounds on Bell Numbers and on Moments of Sums of Random Variables,” *Probability and Mathematical Statistics*, vol. 30, no. 2, pp. 185–205, 2010.
36. O. Granmo, B. J. Oommen, S. A. Myrer, and M. G. Olsen, “Learning Automata-Based Solutions to the Nonlinear Fractional Knapsack Problem With Applications to Optimal Resource Allocation,” *IEEE Trans. Syst. Man Cybern.: B Cybern.*, vol. 37, no. 1, pp. 166–175, Feb. 2007.
37. B. J. Oommen, “Stochastic Searching on the Line and its Applications to Parameter Learning in Nonlinear Optimization,” *IEEE Trans. Syst. Man Cybern.: B Cybern.*, vol. 27, no. 4, pp. 733–739, 1997.
38. M. Stege, J. Jelitto, N. Lohse, M. Bronzel, and G. Fettweis, “A Stochastic Vector Channel Model-Implementation and Verification,” in *IEEE 50th Vehicular Technology Conference*, vol. 1, Sep. 1999, pp. 97–101 vol.1.
39. W. Gale, S. Das, and C. T. Yu, “Improvements to an Algorithm for Equipartitioning,” *IEEE Trans. Comput.*, vol. 39, no. 5, pp. 706–710, May 1990.
40. Y. Liu, M. Elkashlan, Z. Ding, and G. K. Karagiannidis, “Fairness of User Clustering in MIMO Non-Orthogonal Multiple Access Systems,” *IEEE Commun. Lett.*, vol. 20, no. 7, pp. 1465–1468, Jul. 2016.