Power Allocation in Cache-Aided NOMA Systems: Optimization and Deep Reinforcement Learning Approaches

Khai Nguyen Doan, *Student Member, IEEE*, Mojtaba Vaezi, *Senior Member, IEEE*, Wonjae Shin, *Member, IEEE*, H. Vincent Poor, *Fellow, IEEE*, Hyundong Shin, *Senior Member, IEEE*, and Tony Q. S. Quek, *Fellow, IEEE*

***Abstract*— This work exploits the advantages of two prominent techniques in future communication networks, namely caching and non-orthogonal multiple access (NOMA). Particularly, a system with Rayleigh fading channels and cache-enabled users is analyzed. It is shown that the caching-NOMA combination provides a new opportunity of cache hit which enhances the cache utility as well as the effectiveness of NOMA. Importantly, this comes without requiring users' collaboration, and thus, avoids many complicated issues such as users' privacy and security, selfishness, etc. In order to optimize users' quality of service and, concurrently, ensure the fairness among users, the probability that all users can decode the desired signals is maximized. In NOMA, a combination of multiple messages are sent to users, and the defined objective is approached by finding an appropriate power allocation for message signals. To address the power allocation problem, two novel methods are proposed. The first one is a divide-and-conquer-based method for which closed-form expressions for the optimal resource allocation policy are derived making this method simple and flexible to the system context. The second one is based on deep reinforcement learning method that allows all users to share the full bandwidth. Finally, simulation results are provided to demonstrate the effectiveness of the proposed methods and to compare their performance.**

***Index Terms*— Caching, non-orthogonal multiple access (NOMA), deep reinforcement learning, deep learning, power allocation, interference cancellation.**

I. Introduction

THE dramatic growth in the number and capabilities of mobile devices has triggered a dramatic increase in demand for data over wireless networks [1]. This issue is causing a massive load on the backhaul of such networks, especially in systems with densely deployed access points [2], and hence, seriously affecting user quality of service (QoS). Proactive caching techniques provide a promising solution to this problem, in which, besides caching at BS [3], proactively forwarding content to users' devices has been shown to achieve significant benefits [4]–[6]. An obvious advantage of this technique is that it offers opportunities for users to retrieve desired content right from their devices, which considerably reduces the number of transmission sessions needed at peak traffic times, thus, saving peak power and bandwidth.

Non-orthogonal multiple access (NOMA) is another solution for enhancing system capacity and user experiences. NOMA outperforms its counterpart, orthogonal multiple access (OMA), in many contexts by achieving higher power efficiency and lower spectrum usage [7]–[9]. One of the well-known methods to deploy NOMA is based on the power domain in which multiple users' signals are superposed with different power levels while users share a common radio resource of time and frequency [10]–[12]. Subsequently, successive interference cancellation (SIC) is applied by the receivers to decode the desired information.

The deployment of NOMA has been considered in an attempt to find optimal power allocation policies [13], [14]. However there are very few works investigating potential benefits of caching in NOMA context. When enhancing the system performance with the involvement of caching, the situation will be different and more complicated. The reason is that users now are not only affected by the channel conditions, but also the cache placement at the time of generating requests. Because the cached content can be used to eliminate (part of)

Manuscript received March 5, 2019; revised July 16, 2019 and September 20, 2019; accepted September 22, 2019. Date of publication October 15, 2019; date of current version January 15, 2020. This work was supported in part by the U.S. National Science Foundation under Grants CCF-0939370 and CCF-1513915, and in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT (NRF-2019R1C1C1006806). The associate editor coordinating the review of this article and approving it for publication was Y. Liu. *(Corresponding author: Wonjae Shin.)*

K. N. Doan is with the Department of Information Systems Technology and Design, Singapore University of Technology and Design, Singapore 487372 (e-mail: nguyenkhai_doan@mymail.sutd.edu.sg).

M. Vaezi is with the Department of Electrical and Computer Engineering, Villanova University, Villanova, PA 19085 USA (e-mail: mvaezi@villanova.edu).

W. Shin is with the Department of Electronics Engineering, Pusan National University, Busan 46241, South Korea, and also with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544 USA (e-mail: wjshin@pusan.ac.kr).

H. V. Poor is with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544 USA (e-mail: poor@princeton.edu).

H. Shin is with the Department of Electronics and Radio Engineering, Kyung Hee University, Yongin 17104, South Korea (e-mail: hshin@khu.ac.kr).

T. Q. S. Quek is with the Department of Information Systems Technology and Design, Singapore University of Technology and Design, Singapore 487372, and also with the Department of Electronic Engineering, Kyung Hee University, Yongin 17104, South Korea (e-mail: tonyquek@sutd.edu.sg).

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

Digital Object Identifier 10.1109/TCOMM.2019.2947418

the interference in the superposed signal. In terms of these techniques combination, [15] jointly considered the advantages of caching and NOMA. This work designed a power allocation method to ensure that the most popular files could be obtained by a predefined number of content servers. In the recent work [16], the authors narrowed their analysis to a specific case when the user with weaker channel cached information of the user with stronger channel. In [17], the authors focused on minimizing the power consumption in the system. From another point of view, designing a power allocation policy to maximize the users' QoS as well as to guarantee fairness among users is necessary. Moreover, exploiting users' cached content for interference cancellation can improve users achievable rates and should be paid sufficient attention. However, these points have not been jointly considered in the aforementioned works and many of the previous works [18], [19]. In addition, in NOMA systems without caching, power is typically allocated in inverse order of users' channel conditions [20]. However, the user having the worst channel condition may experience the lowest level of interference thanks to the interference cancellation capability offered by caching technique. Therefore, power allocation schemes in non-caching NOMA systems may not be optimal anymore in this context. Thus, it is useful to find resource allocation policies in which the assumptions on the order of channels and the order of allocated power is relaxed. Inspired by these, we propose two power allocation methods to maximize the users' success probability defined as the probability that all users successfully decode their desired signals in a cache-enabled NOMA system. Furthermore, each user can cache a variety of different content items, resulting in many different situations, and our methods are to work in all of those cases. The main contributions of this work are listed as follows:

- We propose a method in which users are paired and the user pairs are separated by orthogonal subchannels to reduce interference. Then, closed-form expressions of power sharing for every user are derived. This method makes use of the channel gain distribution knowledge, users' cached and requested content items at the instant time to maximize the success probability.
- Also, from the spectrum efficiency perspective, another solution is proposed that allows all users to share the full bandwidth. In this context, we formulate the problem as a mixed-integer programming problem which can be solved by existed mixed-integer programming algorithms [21]–[23].
- In order to avoid time-consuming iterative algorithms for solving the formulated optimization problem, a deep-learning-based power allocation method is proposed. Regarding this, we follow a reinforcement learning approach that improves the performance by observing the accuracy of each applied power allocation pattern. We then propose a dual deep neural network model to deal with the nosiness/randomness in the collected training data.

The remainder of this work is organized as follows. Section II describes the system model under consideration. Section III presents our first power allocation method. The second power allocation method is described in Section IV where we first formulate the problem as a solvable mixed-integer programming and then cast it into another form to fit into our designed learning scheme. Subsequently, our simulation results are shown in Section V. Finally, Section VI concludes our work.

II. System Model

We consider a system consisting of K users served by a BS having a content server. Each user's device has a cache with finite capacity and we assume that users cache an entire file rather than a partition of it. With the cache, users typically fetch and store a set of files during an off-peak time called *caching phase*. In this work, we assume that the caching phase has already taken place and consider the next stage called *requesting phase*. In this phase, each user requests for a file in the server. In addition, since files are sent to users for caching by the BS, it has information about files placed in users' caches.

In a downlink NOMA framework in which a superposed signal is transmitted by the BS to users, the SIC is employed at each user to decode the desired signal. Let $P_{\max}$ be the total transmission power and α_i be the portion allocated to the signal of user i where $0 \le \alpha_i \le 1, \forall i = 1, \ldots, K$. With SIC, user i will decode a sequence of signals from the one with strongest to weakest power until the ith signal is decoded. This is because, firstly, strong signals are always easier to be decoded than weak signals. Secondly, decoding and removing stronger signals from the interference will increase the SINR associated with weaker signals making them easier to be decoded. The BS can acknowledge users about decoding order by adding an additional field to the sending information. The model for the signal received at user i is

$$y_i = \frac{h_i}{d_i^{\gamma/2}} \sum_j x_j \sqrt{\alpha_j P_{\max}} + n_i \tag{1}$$

where x_j is the signal corresponding to file j; h_i and d_i are the channel coefficient and the distance between user i and the BS, respectively; γ is the pathloss exponent; and n_i is the Gaussian noise with mean 0 and variance σ^2. Denote $\rho = \frac{P_{\max}}{\sigma^2}$ the signal-to-noise ratio (SNR), and $\beta_i = d_i^{\gamma}\sigma^2$. h_i follows Rayleigh distribution with parameter σ_i, and thus $|h_i|^2$ follows exponential distribution with parameter $\lambda_i = 2/\sigma_i^2$. Note that only the distribution of channels is known by the BS. Let $\mathcal{C}$ be the index set of files cached by user i, then, user i can decode the signal j (signal associated with user j) when

$$\frac{|h_i|^2 \alpha_j P_{\max}}{|h_i|^2 \sum_{k \notin \mathcal{C}, \alpha_k \le \alpha_j} \alpha_k P_{\max} + \beta_i} \ge \epsilon_j \tag{2}$$

where ϵ_j is the minimum SINR required to decode file j. The meaning of the interference term in (2) is that only signals bearing information which has not been cached will constitute the interference; otherwise, those signals will be removed. Thus, we have the condition $k \notin \mathcal{C}$ in the sum. Besides, with SIC, signals with stronger power will be decoded first and removed from the superposed signal. Therefore, when a user decode signal j, only weaker signals constitute the

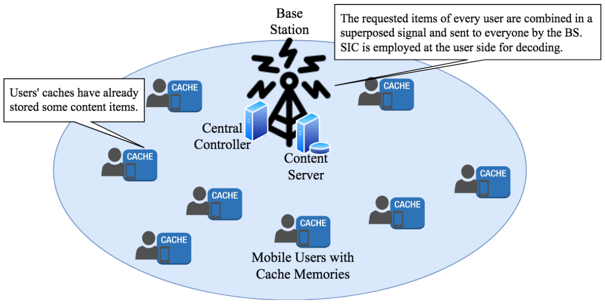

Base Station
The requested items of every user are combined in a superposed signal and sent to everyone by the BS. SIC is employed at the user side for decoding.
Users' caches have already stored some content items.
CACHE
Central Controller
Content Server
Mobile Users with Cache Memories

Fig. 1. The system model under consideration consisting of a BS with a content server serving a set of K users. Each user's device is implemented with a cache storage. At the beginning, each user has already cached some content items.

interference, and thus we have the condition $\alpha_k \leq \alpha_j$ in the range of the sum.

In summary, with SIC users may need to decode a sequence of signals before obtaining their desired ones. In this case, the failure event is said to occur if users fail to decode one of those signals. However, if some content items have already been cached, users can remove the corresponding signals from the interference without decoding, which increases the success probability. Therefore, we aim to design a power allocation policy that exploits both information about channel conditions and cache placement of users to maximize the success probability. Regarding this, we present two power allocation methods in the following two sections. The first method in Section III makes use of orthogonal channels to reduce the interference among users, while the second method in Section IV allows all users to share the whole bandwidth.

By abusing the notations we will use ϵ_j to denote the minimum SINR level required to decode file j and σ^2 for the noise power. Note that although the same notation is used for the SINR threshold, this quantity in Section III and IV are not the same. This is because the first method requires subchannel allocation, while the second method does not. If ϵ is the SINR threshold to decode a file when a user can use the whole bandwidth, and ϵ_W is that when the user can only use $1/W$ of the bandwidth, then, their relation is as follows

$$\boxed{\epsilon_W = \exp\left(W \log\left(1+\epsilon\right)\right) - 1.} \tag{3}$$

In addition, if σ^2 is the power of the Gaussian noise when users use the whole bandwidth, then σ^2/W is that when they only use $1/W$ of the bandwidth. In Fig. 1, the central controller is a computational unit placed at the BS. This unit gathers and processes information to allocate appropriate power to users' signals.

III. Method 1: Divide-and-Conquer-Based Optimization Scheme

In this method, user pairs are interference-isolated with orthogonal frequencies. The entire bandwidth is evenly divided among all pairs. The time-frequency resource allocation in a multi-carrier setting is an interesting topic; however, it is beyond the scope of this work and is a topic of further study. NOMA is then separately applied to each user pair. In this regime, the power allocation consists of two stages as in Fig. 2. The first stage is to share the total transmission power $P_{\max}$ to user pairs, so-called inter-pair power allocation stage. We denote P_i to be the power allocated to the ith pair where $\sum_{i=1}^{K/2} P_i = P_{\max}$. Then, the second stage is to allocate portions of a given power amount to users in each pair, i.e., the first user in pair i will be allocated with a portion α_i of P_i, while the second one is given $1-\alpha_i$. This stage is called intra-pair power allocation stage.[1]

Lemma 1: The whole process can be optimized by optimizing the two stages separately.

In order to prove the above lemma, we first describe the problem formulation as follows

$$\max \prod_{i=1}^{K/2} \mathcal{G}_i\left(\alpha_i, P_i\right) \tag{4}$$

$$\text{s.t.} \sum_{i=1}^{K/2} P_i = P_{\max} \tag{5}$$

$$P_i \geq 0, \forall i = 1, 2, \ldots, K/2 \tag{6}$$

$$0 \leq \alpha_i \leq 1, \forall i = 1, 2, \ldots, K/2. \tag{7}$$

where $\mathcal{G}_i\left(\alpha_i, P_i\right)$ is the success probability of the ith user pair. Through subsection III-A (specifically, expressions (10), (13), (18), (21), (29), (32), (41), (46) and (51)) it will be shown that, in all cases, the success probability of each user pair takes the following form

$$\mathcal{G}_i\left(\alpha_i, P_i\right) = \exp\left(-\frac{\Psi_i\left(\alpha_i\right)}{P_i}\right) \tag{8}$$

where $\Psi_i\left(\alpha_i\right)$ is a function of α_i and does not depend on P_i. In the above formulation, (4) is the probability that all users are success. In the subsection below, we will alternatively analyze all possible cases in which the expressions of user pair's success probability are different in different cases. However, the problem formulations in all cases share a common form which is (4)-(7). Besides that, (5) is the only coupling constraint, meanwhile α_i and $\alpha_j, \forall i \neq j$ are independent due to orthogonal subchannel assumption. Therefore, the intra-pair (regarding variables α_i) and inter-pair (regarding variables P_i) power allocation stages can be optimized separately by minimizing $\Psi_i\left(\alpha_i\right) \forall i$ separately, then solving the above problem with P_i $\forall i$ as variables and with the optimal values of α_i plugged in. In the following, the steps of optimizing α_i are given in subsection III-A, while the form of optimal value for P_i is presented in subsection III-B.

A. Intra-Pair Power Allocation Stage

In this subsection, we work with the second stage of resource assignment and consider a specific user pair. Hence, the user pair indexes in notations will be dropped out for simplicity. Let us denote P to be the total power allocated to this pair and $0 \leq \alpha \leq 1$ be the portion of P allocated to the first user, and thus, the portion of the second user is $1-\alpha$. Note that the value of P will be optimized in the first stage.

[1] The divide-and-conquer-based optimization method is also presented in [13] - a conference version of this work.

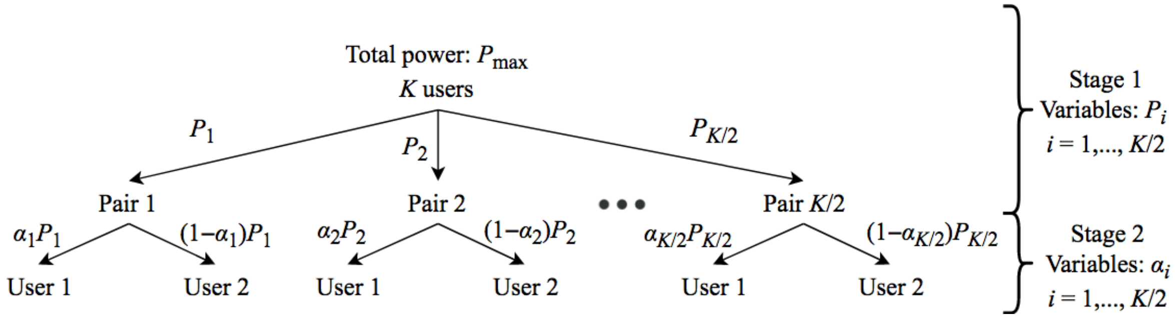


Fig. 2. The two-stage power allocation process in which users are paired and user pairs are assigned orthogonal frequencies.

Without loss of generality, we assume that user 1 requests for file f_1 and user 2 requests for file f_2. In SIC, for each specific user, the signal with stronger power will be decoded first. Therefore, the main idea is to optimize the success probability for both of the cases, $0 \leq \alpha < 0.5$ and $0.5 \leq \alpha \leq 1$, then, the better one will be chosen.

There are some situations in which the optimal power allocation is trivial, whereas some situations require solving optimization problems to find the optimal policy. Therefore, we will first clear out the trivial cases before addressing the rest. Those trivial cases and the corresponding optimal power allocation policies are as follows

1) When both users can be served locally with their caches, no over-the-air transmission is required.
2) When only one user finds his request in his own cache. We assume that is user 2, then all the power P will be allocated for user 1. User 1 can decode his file successfully when

$$\frac{|h_1|^2}{\beta_1} \geq \epsilon_1 \tag{9}$$

Because $|h_i|^2 \sim Exp\,(\lambda_i)$, the success probability is

$$\Pr\left\{|h_1|^2 \geq \epsilon_1\beta_1\right\} = \exp\left(-\frac{\lambda_1\epsilon_1\beta_1}{P}\right). \tag{10}$$

3) When both users request for the same file, but neither of them have cached it. Then, a single signal representing that file is sent with power P to both users. Both users can successfully decode that file when

$$\frac{|h_1|^2}{\beta_1} \geq \epsilon_{1,2} \tag{11}$$

$$\frac{|h_2|^2}{\beta_2} \geq \epsilon_{1,2} \tag{12}$$

where $\epsilon_{1,2}$ denotes the SINR threshold of the file requested by both users. The success probability is given by

$$\Pr\left\{|h_1|^2 \geq \epsilon_{1,2}\beta_1, |h_2|^2 \geq \epsilon_{1,2}\beta_2\right\} = \exp\left(-\frac{\epsilon_{1,2}}{P}\left(\lambda_1\beta_1 + \lambda_2\beta_2\right)\right). \tag{13}$$

To this end, it remains to consider four more-complicated cases. Without loss of generality, we assume that

$$\zeta = \frac{\lambda_1\epsilon_1\beta_1}{\lambda_2\epsilon_2\beta_2} \geq 1. \tag{14}$$

The remaining cases are listed as follows:

1) *C1*: User 1 has cached f_2 and user 2 has had a cache miss.
2) *C2*: User 1 has had a cache miss and user 2 has cached f_1.
3) *C3*: User 1 has cached f_2 and user 2 has cached f_1.
4) *C4*: Both users have had cache misses.

where the term "cache miss" implies that users have not cached any file in the coming signal. Hereafter, we will analyze and derive the optimal power allocation for each case.

In case $C1$, user 1 is capable of eliminating the interference from the superposed signal by exploiting the cached f_2. Thus, user 1 can decode the desired file when

$$\frac{|h_1|^2\alpha P}{\beta_1} \geq \epsilon_1. \tag{15}$$

For the second user, if $0.5 \leq \alpha \leq 1$, user 2 has to decode f_1 with signal of f_2 as interference, remove f_1 from the superposed signal and then decode f_2. If $0 \leq \alpha \leq 0.5$, he can decode f_2 directly with f_1 as interference. These points can be expressed as follows

- If $0.5 \leq \alpha \leq 1$

$$\frac{|h_2|^2\alpha P}{|h_2|^2(1-\alpha)P + \beta_2} \geq \epsilon_1 \tag{16}$$

$$\frac{|h_2|^2(1-\alpha)P}{\beta_2} \geq \epsilon_2. \tag{17}$$

With simple manipulating steps and the fact that $|h_i|^2 \sim Exp\,(\lambda_i)$, the success probability can be expressed as

$$p_1^{C_1} = \exp\left(-\frac{1}{P} \times \tilde{p}_1^{C_1}\right), \tag{18}$$

where

$$\tilde{p}_1^{C_1} = \frac{\lambda_1\epsilon_1\beta_1}{\alpha} + \max\left(\frac{\lambda_2\epsilon_2\beta_2}{1-\alpha}, \frac{\lambda_2\epsilon_1\beta_2}{(1+\epsilon_1)\alpha - \epsilon_1}\right). \tag{19}$$

- If $0 \leq \alpha \leq 0.5$

$$\frac{|h_2|^2(1-\alpha)P}{|h_2|^2\alpha P + \beta_2} \geq \epsilon_2 \tag{20}$$

which gives the following success probability

$$p_2^{C_1} = \exp\left(-\frac{1}{P}\left(\frac{\lambda_1\epsilon_1\beta_1}{\alpha} + \frac{\lambda_2\epsilon_2\beta_2}{1-(1+\epsilon_2)\alpha}\right)\right). \tag{21}$$

In this case, the optimal power allocation is given as in the following theorem.

Theorem 1: When user 1 has cached f_2 and user 2 has had a cache miss, given $\zeta \geq 1$, the optimal power allocation is

$$\alpha^* = \begin{cases} z_1^{C1}, \text{ if } g_1^{C1}\left(z_1^{C1}\right) \leq g_2^{C1}\left(z_2^{C1}\right) \\ z_2^{C1}, \text{ otherwise} \end{cases} \tag{22}$$

where g_1^{C1}, g_2^{C1}, z_1^{C1}, and z_2^{C1} are defined as follows

$$g_1^{C1}(z) = \frac{\lambda_1\epsilon_1\beta_1}{z} + \frac{\lambda_2\epsilon_2\beta_2}{1-z} \tag{23}$$

$$g_2^{C1}(z) = \frac{\lambda_1\epsilon_1\beta_1}{z} + \frac{\lambda_2\epsilon_2\beta_2}{1-(1+\epsilon_2)z} \tag{24}$$

$$z_1^{C1} = \max\left(1 - \frac{1}{\sqrt{\zeta}+1}, 1 - \frac{1}{1+\epsilon_1+\frac{\epsilon_1}{\epsilon_2}}\right) \tag{25}$$

$$z_2^{C1} = \min\left(\frac{1}{1+\epsilon_2}\left(1 - \frac{1}{\sqrt{\zeta(1+\epsilon_2)}+1}\right), 0.5\right). \tag{26}$$

Proof: Please see an appendix in Section VI.

Note that the case *C2* is not equivalent to *C1*, due to (14). In *C2*, user 2 removes the interference by using the cached content, thus, can decode the desired signal when

$$\frac{|h_2|^2(1-\alpha)P}{\beta_2} \geq \epsilon_2. \tag{27}$$

Similar to the previous case, the success conditions for user 1 are

- If $0.5 \leq \alpha \leq 1$

$$\frac{|h_1|^2\alpha P}{|h_1|^2(1-\alpha)P+\beta_1} \geq \epsilon_1. \tag{28}$$

Similarly to the manipulation in the previous case, we have success probability as

$$p_1^{C_2} = \exp\left(-\frac{1}{P}\left(\frac{\lambda_1\epsilon_1\beta_1}{(1+\epsilon_1)\alpha-\epsilon_1} + \frac{\lambda_2\epsilon_2\beta_2}{1-\alpha}\right)\right). \tag{29}$$

- If $0 \leq \alpha \leq 0.5$

$$\frac{|h_1|^2(1-\alpha)P}{|h_1|^2\alpha P+\beta_1} \geq \epsilon_2 \tag{30}$$

$$\frac{|h_1|^2\alpha P}{\beta_1} \geq \epsilon_1. \tag{31}$$

The corresponding success probability is

$$p_2^{C_2} = \exp\left(-\frac{1}{P} \times \tilde{p}_2^{C_2}\right), \tag{32}$$

where

$$\tilde{p}_2^{C_2} = \max\left(\frac{\lambda_1\epsilon_2\beta_1}{1-(1+\epsilon_2)\alpha}, \frac{\lambda_1\epsilon_1\beta_1}{\alpha}\right) + \frac{\lambda_2\epsilon_2\beta_2}{1-\alpha}. \tag{33}$$

Then, the optimal power allocation for *C2* is presented in Theorem 2.

Theorem 2: When user 1 has had a cache miss and user 2 has cached f_1, given $\zeta \geq 1$, the optimal power allocation is as follows

$$\alpha^* = \begin{cases} z_1^{C2}, \text{ if } g_1^{C2}\left(z_1^{C2}\right) \leq g_2^{C2}\left(z_2^{C2}\right) \\ z_2^{C2}, \text{ otherwise} \end{cases} \tag{34}$$

where g_1^{C2}, g_2^{C2}, z_1^{C2} and z_2^{C2} are defined as follows

$$g_1^{C2}(z) = \frac{\lambda_1\epsilon_1\beta_1}{(1+\epsilon_1)z-\epsilon_1} + \frac{\lambda_2\epsilon_2\beta_2}{1-z} \tag{35}$$

$$g_2^{C2}(z) = \frac{\lambda_1\epsilon_1\beta_1}{z} + \frac{\lambda_2\epsilon_2\beta_2}{1-z} \tag{36}$$

$$z_1^{C2} = 1 - \frac{1}{\sqrt{\zeta(1+\epsilon_1)}+1+\epsilon_1} \tag{37}$$

$$z_2^{C2} = \min\left(\frac{1}{1+\epsilon_2+\frac{\epsilon_2}{\epsilon_1}}, 0.5\right). \tag{38}$$

Next, for the case *C3*, both users can use their cached content items to remove the interference from the superposed signal, therefore, the conditions for them to successfully decode the desired signal does not depend on where α is relative to 0.5.

$$\frac{|h_1|^2\alpha P}{\beta_1} \geq \epsilon_1 \tag{39}$$

$$\frac{|h_2|^2(1-\alpha)P}{\beta_2} \geq \epsilon_2. \tag{40}$$

The success probability of this case can be derived as follows

$$p^{C_3} = \exp\left(-\frac{1}{P}\left(\frac{\lambda_1\epsilon_1\beta_1}{\alpha} + \frac{\lambda_2\epsilon_2\beta_2}{1-\alpha}\right)\right). \tag{41}$$

Then, the optimal policy for this case is simple and summarized in the next theorem.

Theorem 3: When user 1 has cached f_1 and user 2 has cached f_2, given $\zeta \geq 1$, the optimal power allocation is as follows

$$\alpha^* = 1 - \frac{1}{\sqrt{\zeta}+1}. \tag{42}$$

Finally, case *C4*, both users experience interference, and user with lower power need to decode other users' file, remove it from the superposed signal before decoding his own file. Thus, the success conditions are

- If $0.5 \leq \alpha \leq 1$

$$\frac{|h_1|^2\alpha P}{|h_1|^2(1-\alpha)P+\beta_1} \geq \epsilon_1 \tag{43}$$

$$\frac{|h_2|^2\alpha P}{|h_2|^2(1-\alpha)P+\beta_2} \geq \epsilon_1 \tag{44}$$

$$\frac{|h_2|^2(1-\alpha)P}{\beta_2} \geq \epsilon_2. \tag{45}$$

with similar manipulation steps as case $C1$, the success probability is derived as

$$p_1^{C4} = \exp\left(-\frac{1}{P} \times \tilde{p}_1^{C4}\right), \tag{46}$$

where

$$\tilde{p}_1^{C4} = \frac{\lambda_1\epsilon_1\beta_1}{(1+\epsilon_1)\,\alpha-\epsilon_1} + \max\left(\frac{\lambda_2\epsilon_1\beta_2}{(1+\epsilon_1)\,\alpha-\epsilon_1}, \frac{\lambda_2\epsilon_2\beta_2}{1-\alpha}\right). \tag{47}$$

- If $0 \le \alpha \le 0.5$

$$\frac{|h_1|^2(1-\alpha)P}{|h_1|^2\alpha P+\beta_1} \ge \epsilon_2 \tag{48}$$

$$\frac{|h_1|^2\alpha P}{\beta_1} \ge \epsilon_1 \tag{49}$$

$$\frac{|h_2|^2(1-\alpha)P}{|h_2|^2\alpha P+\beta_2} \ge \epsilon_2. \tag{50}$$

The corresponding success probability is

$$p_2^{C4} = \exp\left(-\frac{1}{P}\times\tilde{p}_2^{C4}\right), \tag{51}$$

where

$$\tilde{p}_2^{C4} = \max\left(\frac{\lambda_1\epsilon_2\beta_1}{1-(1+\epsilon_2)\,\alpha}, \frac{\lambda_1\epsilon_1\beta_1}{\alpha}\right) + \frac{\lambda_2\epsilon_2\beta_2}{1-(1+\epsilon_2)\,\alpha}. \tag{52}$$

Theorem 4 presents the optimal policy for this case.

Theorem 4: Given $\zeta \ge 1$, when both users have cache misses, the optimal power allocation policy is as follows

$$\alpha^* = \begin{cases} z_1^{C4}, \text{ if } g_1^{C4}\left(z_1^{C4}\right) \le g_2^{C4}\left(z_2^{C4}\right) \\ z_2^{C4}, \text{ otherwise} \end{cases} \tag{53}$$

where g_1^{C4}, g_2^{C4}, z_1^{C4} and z_2^{C4} are defined as follows

$$g_1^{C4}(z) = \frac{\lambda_1\epsilon_1\beta_1}{(1+\epsilon_1)\,z-\epsilon_1} + \frac{\lambda_2\epsilon_2\beta_2}{1-z} \tag{54}$$

$$g_2^{C4}(z) = \frac{\lambda_1\epsilon_1\beta_1}{z} + \frac{\lambda_2\epsilon_2\beta_2}{1-(1+\epsilon_2)\,z} \tag{55}$$

$$z_1^{C4} = 1-\min\left(\frac{1}{\sqrt{1+\epsilon_1}\left(\sqrt{\zeta}+\sqrt{1+\epsilon_1}\right)}, \frac{1}{1+\epsilon_1+\frac{\epsilon_1}{\epsilon_2}}\right) \tag{56}$$

$$z_2^{C4} = \min\left(\frac{1-\frac{1}{\sqrt{\zeta(1+\epsilon_2)}+1}}{1+\epsilon_2}, \frac{1}{1+\epsilon_2+\frac{\epsilon_2}{\epsilon_1}}, 0.5\right). \tag{57}$$

Note that the expression of z_2^{C4} is a minimum function with three arguments. Due to space limitation, the proofs of Theorem 2, 3 and 4 are omitted, however, their results can be derived in a very similar way as that of Theorem 1.

In summary, we have presented, in this subsection, the optimal power allocation for each user in a pair where user pairs are given orthogonal subchannels. The system state information constituted by users' requests and their cached content items can be gathered by the BS at the time requests are generated. Then, the current state can be defined to be one of the analyzed cases, and the corresponding power allocation is applied. In this subsection, we assume that the total power given to the considered user pair is fixed to be P, and the optimal portion shared to each user is derived. In the next subsection, we will discuss how P is defined for each pair of user. Before closing this subsection, we will point out some important observations.

B. Inter-Pair Power Allocation Stage

In this subsection, we address the power allocation for user pairs, i.e., deriving P_i for all pair i such that the success probability is maximized. As pointed out at the beginning of this subsection, when optimizing variables P_i, $\Psi_i \forall i$ are fixed at their optimal values which are obtained by substituting α_i by the results given in Theorem 1-4. Let us denote the optimal value of $\Psi_i(\alpha_i)$ (when α_i is optimized) as Ψ_i^*.

To this end, the objective function in (4) can be written as

$$\mathcal{G}(\mathbf{P}) = \exp\left(-\sum_{i=1}^{K/2}\frac{\Psi_i^*}{P_i}\right) \tag{58}$$

where $\mathbf{P} = \left[P_1, P_2, \ldots, P_{K/2}\right]$ and (4) can be replaced by

$$\min_{\mathbf{P}} \sum_{i=1}^{K/2}\frac{\Psi_i^*}{P_i}. \tag{59}$$

Solving (59) with the set of constraints (5) and (6) by applying KKT conditions gives us the following closed-form solution

$$\boxed{P_i = \frac{\sqrt{\Psi_i^*}}{\sum_{j=1}^{K/2}\sqrt{\Psi_j^*}}P_{\max}, \forall i = 1, \ldots, K/2.} \tag{60}$$

Note that each user pair is assigned $2/K$ of the available bandwidth, therefore, as mentioned at the end of Section II, the SINR thresholds required to decode files will increase as the number of user pairs increases, however, the noise power σ^2 will decrease for each pair. This is because σ^2 is inversely proportional to the number of partitioned subchannels.

In summary, the method proposed in this section helps create two separable power allocation stages. The closed-form solutions for both stages are provided in Theorem 1-4, three trivial cases and (60). These results together with the simple design of this method allow it to be flexibly applicable in various system contexts.

IV. Method 2: Deep-Reinforcement-Learning-Based Scheme

Our presented Method 1 simplifies the problem with a user pairing technique that allows us to obtain closed-form expressions. In this section, from the bandwidth efficiency perspective, we propose another method based on machine learning, which not only responds quickly upon users' requests but also allows all users to share the entire bandwidth. The application and performance of both methods will be compared and summarized in Sections V and VI. However, before discussing the technical details, we will formulate the power allocation problem as an optimization problem. This is, first, to provide a better mathematical view of our considered problem. Second, we want to show that applying iterative optimization algorithms is not suitable for this context since users require a short-delay response from the BS. These are the main motivations for proposing a learning-based approach.

A. Problem Formulation

For the ease of notation, let us number the users from 1 to K, and we also call $f_1, f_2, \ldots, f_K$ the files requested by user 1, 2, $\ldots$, K, respectively. Let $C_{ij} = 1$ if the ith user has cached f_j, and $C_{ij} = 0$, otherwise, $\forall i, j = 1, \ldots, K$. Similar to the previous section, $C_{ij}, \forall i, j$ are known prior to the power allocation process. Recall that the success probability expression is not defined if the order of $\alpha_1, \alpha_2, \ldots, \alpha_K$ is not defined, where α_i is the portion of the total power $P_{\max}$ allocated to the ith user. Therefore, we use another set of variables ψ_{ij} which, by taking value 1, implying that $\alpha_i \geq \alpha_j$ and by taking value 0, implying that $\alpha_i < \alpha_j$.

The condition for user i to successfully decodes file f_j is as follows

$$\frac{|h_i|^2 \alpha_j}{|h_i|^2 \sum_{k \neq j} C_{ik} \psi_{jk} \alpha_k + \beta_i} - (1 - C_{ij}) \epsilon_j \geq 0 \tag{61}$$

which can be rewritten as

$$\begin{cases} |h_i|^2 \geq \frac{(1-C_{ij}) \epsilon_j \beta_i}{\alpha_j - \epsilon_j \sum_{k \neq j} C_{ik} \psi_{jk} \alpha_k} \\ \alpha_j - \epsilon_j \sum_{k \neq j} C_{ik} \psi_{jk} \alpha_k > 0 \end{cases}. \tag{62}$$

In other words, if (62) are not satisfied, user i cannot decode file f_j. Maximizing the success probability is maximizing the following function

$$\exp\left(- \sum_{i=1}^{K} \lambda_i \sum_{j=1}^{K} \frac{(1 - C_{ij})(1 - \psi_{ij}) \epsilon_j \beta_i}{\alpha_j - \epsilon_j \sum_{k \neq j} C_{ik} \psi_{jk} \alpha_k} \right) \tag{63}$$

where the term $(1 - \psi_{ij})$ is added to the objective function to imply that the i-th user only need to obtain file f_j if $\alpha_i < \alpha_j$ or equivalently, $\psi_{ij} = 0$. The binary variables ψ_{ij} are to define the order of $\alpha_i, \forall i, j$. To this end, the problem formulation to maximize the success probability can be expressed as

$$\min_{\psi, \alpha} \sum_{i=1}^{K} \lambda_i \sum_{j=1}^{K} \frac{(1 - C_{ij})(1 - \psi_{ij}) \epsilon_j \beta_i}{\alpha_j - \epsilon_j \sum_{k \neq j} C_{ik} \psi_{jk} \alpha_k} \tag{64}$$

$$\text{s.t. } \psi_{ij} + \psi_{ji} = 1, \forall i, j = 1, \ldots, K \tag{65}$$

$$\psi_{ij} \in \{0, 1\}, \forall i, j = 1, \ldots, K \tag{66}$$

$$\sum_{i=1}^{K} \alpha_i = 1 \tag{67}$$

$$-1 \leq \alpha_i - \alpha_j - \psi_{ij} \leq 0, \forall i, j = 1, \ldots, K \tag{68}$$

$$\alpha_j - \epsilon_j \sum_{k \neq j} C_{ik} \psi_{jk} \alpha_k \geq \xi, \forall i, j = 1, \ldots, K \tag{69}$$

$$0 \leq \alpha_i \leq 1, \forall i = 1, \ldots, K. \tag{70}$$

where maximizing (63) is equivalent to (64). The constraint set (65) is to guarantee the consistency, i.e., as $\alpha_i < \alpha_j$ or $\psi_{ij} = 1$, then we cannot have $\alpha_j < \alpha_i$ meaning that ψ_{ji} must be 0. Constraint sets (66), (67) and (70) are due to the definition of our variables. The constraint set (68) presents the relationship between ψ_{ij} and $\alpha_i, \forall i, j$, which is $0 \leq \alpha_i - \alpha_j \leq 1$ if $\psi_{ij} = 1$ and $-1 \leq \alpha_i - \alpha_j \leq 0$ if $\psi_{ij} = 0$. Finally, the constraint set (69) is from the second inequality of (62). We have replaced 0 on the right-hand side by a small positive number ξ to convert “$>$” to ”$\geq$” making the problem easier to be solved by optimization techniques. The problem can be solved by applying searching methods or mixed-integer programming algorithms [21]–[23].

Unlike the previous section in which we can allocate orthogonal frequencies to each user pair, and since the number of users occupying a frequency is limited to two, we can derive a closed-form expression for the optimal policy. In this part, users share the same frequency and in addition, the order of $\alpha_i, \forall i$ is not defined yet, hence, we have to solve a mixed integer programming problem with a non-convex objective function as in (64)-(70). Therefore, optimization algorithms may require several iterations, which is time consuming, thus increasing the delay. In other words, by applying optimization techniques, users always have to wait for the central controller to find a suitable power allocation policy via iterative numerical algorithms.

Being motivated by the above, we proposed a learning mechanism allowing our system to improve its performance over time. This approach, after the training stage, will be able to incur no delay in term of resource allocating process.

B. Deep-Reinforcement-Learning-Based Scheme

1) Problem Re-Formulation: Before introducing a new form of problem formulation such that a learning method can be applied, we would like to mention the concept of *users' advantage* which consists of two sides, the user advantage in terms of channel conditions and cached content items. In the case without caching, before power is allocated to users, user advantage is defined only by channel conditions, i.e., a user with better channel condition is considered to have a higher advantage. Then, we allocate higher power for users having less advantage. In our problem, another type of advantage coming from users' cached content items is that users caching more files in the superposed signal have more advantage in terms of interference cancellation. Therefore, in order to allocate appropriate power, the user advantages need to be jointly exploited from both of these sides.

Following a reinforcement learning method, our central controller (at the BS) can be treated as an acting agent and the environment, here, contains all other components such as channels, SINR thresholds of files, users' requests, cached files, etc. In order to capture the aspect of users' advantages, a state is defined as follows. Let K^2-dimensional vector $\mathbf{S}$ represent a state. For the ease of understanding, this state vector can be described as a concatenation of K row vectors (each has the dimension of K) of matrix $\mathbf{S}^M$. Each row i of matrix $\mathbf{S}^M$ is associated with a user, whose jth element is

$$\mathbf{S}_{ij}^{M} = \begin{cases} \epsilon_j, \text{ if user } i \text{ has not cached file } f_j \\ 0, \text{ if user } i \text{ has cached file } f_j \text{ and } i \neq j \\ 0, \forall j = 1, \ldots, K, \text{ if user } i \text{ has cached file } f_i. \end{cases} \tag{71}$$

where ϵ_j is the SINR threshold of the requested file of user j (file f_j). If user i has cached file f_j, the jth element on row i will be 0 implying that user i does not need to decode file f_j in the superposed signal. If user i has cached his requested file (file f_i), then all elements on row i are 0 implying that user i

does not need to decode any file in the superposed signal. Once users generate their requests, the vector $\mathbf{S}$ is built at the BS, then based on the central controller decision, the BS gives back an action as a K-dimensional vector

$$\alpha = [\alpha_1 \quad \alpha_2 \quad \dots \quad \alpha_K] \tag{72}$$

which are the portions of power allocated to users' signal. Once power is allocated to signals sent to users, the number of users who successfully decode their desired signals is gathered by the BS as a reward which we aim to maximize in the long run.

2) Deep-Reinforcement-Learning-Based Approach: In the proposed learning algorithm hereafter, there are generally three phases called *exploration*, *training* and *exploitation* phases. The description is given below.

Exploration Phase: In this phase, we try to discover which action returns the best reward for each encountered state. This is done by trying and observing. When an action (power allocation vector) is applied to the environment, the reward (number of success users) can be obtained correspondingly. Since the channel gains are random, the reward associated with a specific action in a specific state will also be random. Therefore, the average values of rewards will be considered instead of instantaneous values. This requires to apply a specific action to a specific state several times to obtain the average reward associated with that state-action pair. In this context, we are dealing with a continuous action space, hence, by generating actions randomly, there is no chance to pick an action twice. Even with quantization, it will take a significantly longer time. To address this problem, we denote $A_{\max}$ to be the maximum number of actions that we will apply to a given state before concluding about the best action associated with that state. The larger $A_{\max}$ is, the more reliable our conclusion about the best action is.

Before executing the exploration phase, the users' channels are probed to construct the distribution of channels. Then, this phase is completed in four main steps: randomly generating users' requests and cached items, randomly generating a power allocation vector, drawing channel coefficients from the constructed distribution and computing the reward, and finally, computing the average reward and storing the best found action with its associated state. This process can be applied to a general context when the channel distribution is unknown. However, in this work we assume that the channel follows a Rayleigh distribution, hence, channel probing can be omitted. The mentioned four steps will be looped sequentially (T_{Trial} loops). Because the channels vary over time, the reward associated with a state-action pair is not consistent, hence, the third step (drawing channel coefficients and computing rewards) needs to be done several times to compute the average reward. Thus, it is put into an internal loop (T_{Eval} loops). The complexity of this process is, therefore, $O\left(T_{\text{Trial}} \times T_{\text{Eval}}\right)$. Note that we aim to deal with all possible cases, hence, we do not need to consider users' preferences or caching strategies, which is the reason why the first step in the loop can be done in a random manner. Algorithms 1 and 2 summarize the described process in which $C_i^{\max}$ denotes the maximum cache capacity of user i and $\alpha\left(\mathbf{S}\right)$ and $r\left(\mathbf{S}\right)$ are the best action and reward associated with state $\mathbf{S}$, respectively, that we have explored so far.

Algorithm 1 Exploration Phase

```
Input: $C_i^{\max}\ \forall i = 1, \dots, K$ and $T_{\text{Trial}}$.
Output: A stored list of state-action pairs where the action
    in a pair is the best one found for the state in the same
    pair.
for $t$ from 1 to $T_{\text{Trial}}$ do
    Caching: $K$ groups of files are chosen randomly. Group
        $i$ consists of an arbitrary number of different files up
        to $C_i^{\max}$ representing files cached by a user $i$.
    Requesting: $K$ files are chosen randomly representing
        files requested by users.
    State Forming: Based on the set of cached and
        requested files, the matrix of (71) is constructed, and
        the state vector $\mathbf{S}$ is formed by concatenating row
        vectors.
    $\alpha^*, r^* \leftarrow$ find_best_action$(\mathbf{S})$
    if $\mathbf{S}$ has not been stored and memory is not full then
        $\mathbf{S}$ is stored; $\alpha\left(\mathbf{S}\right) \leftarrow \alpha^*$; $r\left(\mathbf{S}\right) \leftarrow r^*$
    if $\mathbf{S}$ has been stored and $\alpha^* > \alpha\left(\mathbf{S}\right)$ then
        $\alpha\left(\mathbf{S}\right) \leftarrow \alpha^*$; $r\left(\mathbf{S}\right) \leftarrow r^*$
```

Training Phase: After the exploration phase, we obtain a list of states and the corresponding best actions. Subsequently, a parametric function is needed to be designed to learn from this data set the general rule of allocating power. After training, the function will be used as a predictor to predict the best action for an input state. To support the design of the predictor, understanding properties of the training data is necessary. Particularly, because the list of good actions in the training data is constructed from the random acting process, it bears a certain randomness. The randomness of training data comes from the fact that there are many different action vector α resulting in the same reward for each state. Therefore, two very similar states can be associated with totally different action vectors, which causes inconsistency and obstructs the learning process.

For ease of understanding, an example is given as follows. Considering a three-user scenario, for a state $\mathbf{S}_1$, an action $\alpha_1 = [0.7 \quad 0.2 \quad 0.1]$ results in a failure of only user 3, which brings the best reward of $r\left(\mathbf{S}_1\right) = 2$. However, another action $\alpha_2 = [0.0 \quad 0.3 \quad 0.7]$ which results in a failure of only user 1 can also give the best reward of $r\left(\mathbf{S}_1\right) = 2$. Then, either α_1 or α_2 will be associated with $\mathbf{S}_1$ depending on which one is encountered first. In addition, there is a situation that a state $\mathbf{S}_2$ which is very similar to $\mathbf{S}_1$ goes with α_2, while $\mathbf{S}_1$ goes with α_1. Hence, for two very similar states, the actions going with them can be randomly different. Moreover, if channel conditions of users are good, they can decode the desired signals without being allocated much power. This means that action vectors such as $\alpha_3 = [0.1 \quad 0.15 \quad 0.75]$, $\alpha_4 = [0.4 \quad 0.51 \quad 0.09]$, and $\alpha_5 = [0.9 \quad 0.05 \quad 0.05]$ can all give the same best reward, although these vectors are very different and distributed randomly in the action space.

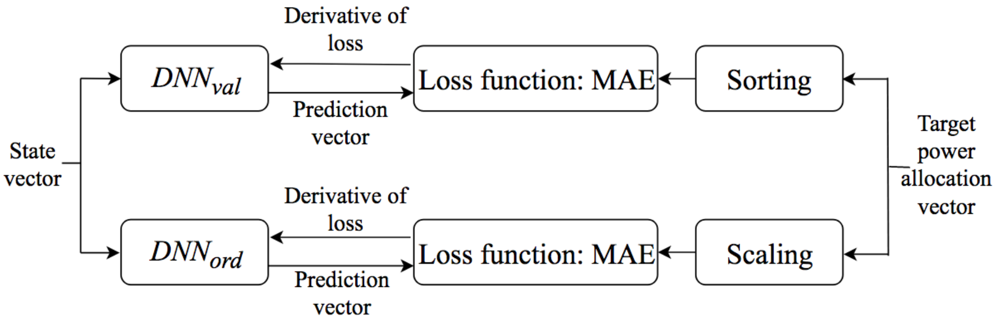


Fig. 3. Illustration for the training phase of the dual-network prediction model. The network weights are optimized based on forward and backpropagation mechanism with MAE as a loss function.

Algorithm 2 find_best_action($\mathbf{S}$)

```
Input: A state $\mathbf{S}$, $A_{\max}$ and $T_{\text{Eval}}$.
Output: The best action for the input state and the corresponding reward.
for $j$ from 1 to $A_{\max}$ do
    Action Generating: Generate $\alpha_j = (\alpha_{j1}, \ldots, \alpha_{jK})$ randomly where $\alpha_{jk} \sim Unif\,(0,1), \forall k = 1, \ldots, K$. If the $[K\,(k-1)+k]$-th element of $\mathbf{S}$ is greater than 0 then $\alpha_{jk} \leftarrow 0$. Finally, $\alpha_{jk} \leftarrow \alpha_{jk} / \sum_{i=1}^{K} \alpha_{ji}$.
    for $t$ from 1 to $T_{\text{Eval}}$ do
        Reward Obtaining: The channel coefficients are drawn from the obtained distribution (Rayleigh, in this case). From vectors $\mathbf{S}$ and $\alpha_j$, $r_{jt}$ is computed as the total number of success users following SIC process.
    Estimate the average reward by: $\bar{r}_j = \frac{1}{T_{\text{Eval}}} \sum_{t=1}^{T_{\text{Eval}}} r_{jt}$.
$j^* \leftarrow \underset{j}{\text{argmax}}\ \bar{r}_j$
return $\alpha_{j^*}, \bar{r}_{j^*}$
```

Due to the complexity and noisiness of the training data in our work, and to generalize the relationship between states and actions, deep neural network models are built for the goal of anticipation. Our networks are trained based on the forward and backpropagation scheme [24]. The error is computed in the forward propagation stage, and the backpropagation stage evaluates the gradient of the error function with respect to the weight set via the derivative chain rule. By executing the forward and backpropagation alternatively, the weights can be updated in each training epoch. In the experiments in Section V, our networks are trained with two different types of loss. Also, the ADAM optimization algorithm [25] is employed to minimize the loss. The first loss function type is simply the mean absolute error (MAE) between the network outputs and the target output vectors. The second type consists of two terms. The first term is simply the MAE between network outputs and target outputs. The second term is the average SINR values computed from the network outputs and target outputs. The average-SINR part consists of $K + (K-1) + \ldots + 1$ terms corresponding to SINR values associated with signals that users need to decode. For example, if $K = 2$ with $\alpha_1 > \alpha_2$, the average SINR in this case will consist of three terms. The first one is the average SINR regarding signal 1 received by user 1, i.e., $\mathbb{E}\left[\frac{|h_1|^2\alpha_1 P_{\max}}{|h_1|^2\alpha_2 P_{\max}+\beta_1}\right]$. Since user 2 will need to decode signal 1 and then signal 2, the last two terms are $\mathbb{E}\left[\frac{|h_2|^2\alpha_1 P_{\max}}{|h_2|^2\alpha_2 P_{\max}+\beta_2}\right]$ and $\mathbb{E}\left[\frac{|h_2|^2\alpha_2 P_{\max}}{\beta_2}\right]$.

Besides, we observe that the randomness takes place in two aspects of the training data (action vectors), the first one is the values of elements and the second one is the order of those elements. In order to mitigate the randomness and simplify the learning process, one of our solution is separating those two aspects and learning them separately. To be more specific, we design a dual-network model consisting of two deep networks with the same input and output dimensions, called DNN_{val} and DNN_{ord}. We denote $\alpha_{val}^{\mathbf{S}}$ and $\alpha_{ord}^{\mathbf{S}}$ the output from DNN_{val} and DNN_{ord}, respectively, with input state $\mathbf{S}$. The elements of $\alpha_{val}^{\mathbf{S}}$ is supposed to indicate the power allocation for all users in a descending (or ascending) order. The elements of $\alpha_{ord}^{\mathbf{S}}$ is supposed to indicate the corrected order of elements in $\alpha_{val}^{\mathbf{S}}$. Note that elements of $\alpha_{ord}^{\mathbf{S}}$ have continuous values in $[0,1]$, and their order (not the values) are used to arrange elements in $\alpha_{val}^{\mathbf{S}}$ before applying to the environment.

For training DNN_{val}, the set of target vectors are sorted to eliminate the randomness in element order. In our experiments in Section V, descending order is used. Then, DNN_{val} is trained with the original input set and the sorted target output set. In terms of DNN_{ord}, we want to learn only the order of elements (to remove the randomness in their values). Hence, for the ease of learning and, concurrently, for preserving the correlation of data, we scale elements in the target output vectors by multiplying with $\xi_{scale} > 1$. The training process for the dual-network model is illustrated in Fig. 3.

Exploitation Phase: Finally, the trained model can be used to perform the power allocation for every encountered state. The use of the single-network model is straightforwardly passing the input through layers of the network. Meanwhile, exploiting the dual-network model is a bit more complicated, thus, the process is summarized in Fig. 4. Note that the order used in "Sorting" and "Extracting element order" must be the same as that used in the training phase.

In conclusion, from the timing perspective, solving (63)-(70) includes searching for the set of ψ_{ij}, whose delay is unstable with a high worst-case delay of $2^{K^2}T_\alpha$ where T_α is the delay of finding α_i for a given ψ_{ij} set. For example, given the

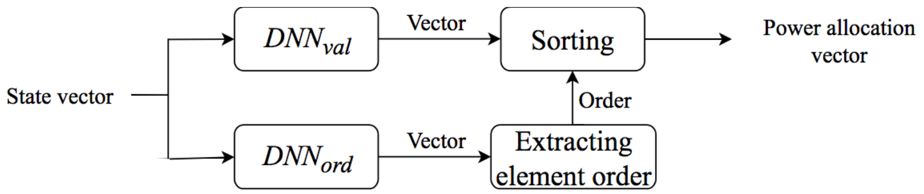


Fig. 4. Exploiting dual-network model for predicting power allocation vector.

set ψ_{ij}, minimizing the Lagrangian function using a standard Newton's method will result in a complexity of $O\left(\varepsilon^{-2}\right)$ with ε is the error between the obtained result and the nearest local optimum. In addition, the delay will take place infinitely many times whenever users make their requests. On the other hand, the proposed learning approach has a stable and finite one-time delay, because the delay comes mainly from the exploration and training phases. To be more specific, the complexity of the exploration phase is $O\left(T_{\text{Trial}} \times T_{\text{Eval}}\right)$ as mentioned, and the complexity of the training phase, with forward and backpropagation algorithm, is $O\left(cS_{train}T_{train}\right)$. Here, c is a constant which depends on the number of layers and units of DNN model, S_{train} is the training set size and T_{train} is the number of training epochs. Finally, the complexity of the exploitation phase is $O\left(1\right)$.

V. Numerical Results

Numerical results illustrating the performance of proposed methods are presented in this section. We consider a library of 38 files. Their SINR thresholds relative to the full bandwidth usage take values from 0.016 to 0.608 with a step of 0.016, respectively. The default values of other parameters are as follows: $P_{\max} = 1, \sigma^2 = 1$ (when using the entire bandwidth), $d_i = 1, \forall i = 1, \ldots, K$ and every file has the same chance to be requested. Because the proposed methods are targeted to work for all cases of cache placement at the user side, in our experiments, the cache placement will be done in a random manner.

For the neural network architecture, we use a 5-layer network having the dimension of 9×20 (input layer), 20×30, 30×20, 20×10 (three hidden layers) and 10×3 (output layer), respectively. The activation functions between hidden layers are **relu** functions, and that at the output is a **softmax** function. For the case of the dual-network model, each network also has the same mentioned architecture. Those hyper parameters in our networks are obtained by tuning. The scaling factor when using the dual-network model is set to be $\xi_{scale} = 2$ by default. At the beginning of the training session, the weights of our networks are initialized randomly.

Baseline methods: For performance evaluation, we include the following methods in our experiments:

- Orthogonal multiple access (OMA): all users are allocated orthogonal subchannels and the power can be assigned following our Method 1 with only stage 1 (as in Fig. 2) where a user pair is replaced by an individual. Therefore, the expression (60) can be applied with the following straightforward modifications: $\Psi_i^* = \lambda_i \beta_i \epsilon_i$ is associated with each user i and the denominator is now $\sum_{j=1}^{K} \sqrt{\Psi_j^*}$.
- Equal power allocation: all users are allowed to share the whole bandwidth and allocated the same power of $P_{\max}/K$.
- Maximin-fairness power allocation (MMF) [14]: MMF is chosen for comparison since both the network scenario and objective considered in this work are similar to ours. The differences are that there is no caching enabled, the objective is to maximize the lowest communication rate among users, and this method allocates power based on channel-to-noise ratios (CNRs) defined by $\frac{|h_i|^2}{d_i^\gamma \sigma^2}$. In other words, it requires the knowledge of instantaneous channel information which is not assumed to be available in our work. Therefore, in the experiments, the average CNRs will be used when applying this method.

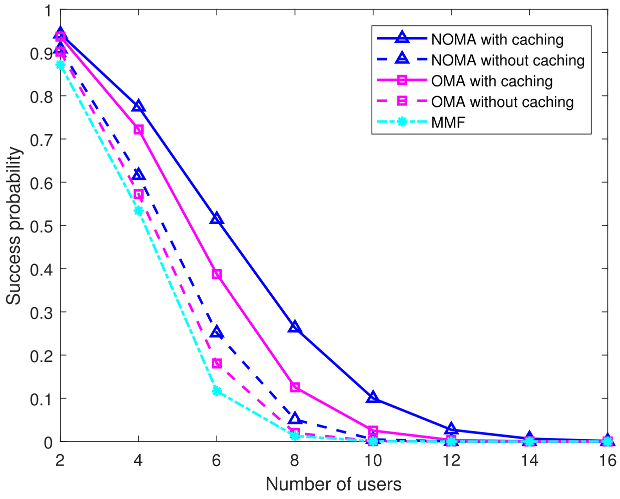
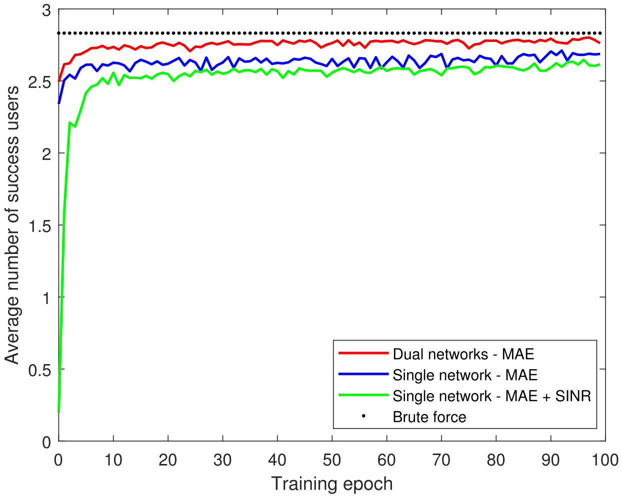
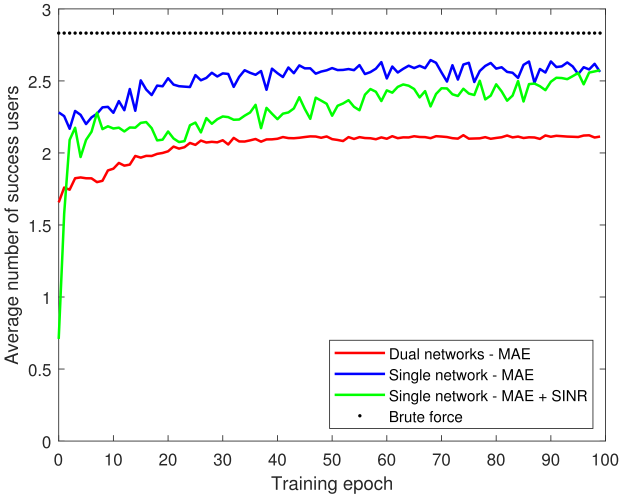


Fig. 5. The success probability as a function of user number who joins the system under the deployment of Method 1. The comparison is conducted between NOMA and OMA with caching enabled and disabled.

In Fig. 5, we investigate the impact of the user density on the communication quality under the deployment of Method 1. There are $K = 4$ users involved in this experiment. Users are paired with $|h_1|^2$ and $|h_2|^2$ follow the exponential distribution with mean ~~1~~ and 2, respectively. This setup is repeated for every pair. First of all, with caching enabled, the effectiveness of both NOMA and OMA is increased, it also enlarges the gap between these two schemes. This, once again, emphasizes the important role of caching in our communication systems. Secondly, as in the previous figure, NOMA still outperforms its counterpart. This is because each user can use as twice



(a) 10^5-sample training set



(b) 10^4-sample training set

Fig. 6. Performance of prediction models in terms of the average number of success users during the training process.

of the bandwidth as that of the OMA case, hence deal with lower SINR requirement. Although OMA offers a lower noise power for each user, it is not sufficient to compensate for the rise in SINR thresholds. In addition, the gap between NOMA and OMA is enlarged when caching in introduced, telling that NOMA scheme can make use better of the advantage of cache-enabled networks. Finally, the MMF method is a bit below the OMA scheme due to the unavailability of instantaneous channel information.

The performance of Method 2 during the training session is recorded in Fig. 6. There are $K = 3$ users in this context with $|h_1|^2, |h_2|^2$ and $|h_3|^2$ follow exponential distribution with mean 1, 2 and 3, respectively. Each user can cache up to 2 files. As mentioned previously, two kinds of predictor architectures are presented with different loss functions for training. "MAE + SINR" implies the loss function with two terms of output's MAE and SINR's MAE. Note that in the system with 3 users, the SINR itself will consist of 6 terms. The average number of users who can successfully decode their desired signals is used as a metric to evaluate the prediction accuracy of neural networks. As can be seen from the figure, on one hand with a single network, we achieve slightly better results when using the simple MAE as loss function. This is because SINR depends on the order of power allocated amounts (stronger signals are decoded first and decoded by more users), hence, is a complicatedly non-differentiable function with respect to the network's output vector. On the other hand, using the dual networks gives the best results closing to brute force level (Fig. 6(a)). Since this model can reduce the uncertainty in the training data. However, the error from this kind of model is affected by that of two networks, thus, it is required to be trained well with sufficient samples. This makes the dual-network model appears to be the most sensitive one to dataset size. When the dataset size is restricted to one-tenth of the original one (Fig. 6(b)), the rise in the error of both networks suppresses the performance of this model. Generally, all the presented candidates have their accuracy grown when being trained with more samples.

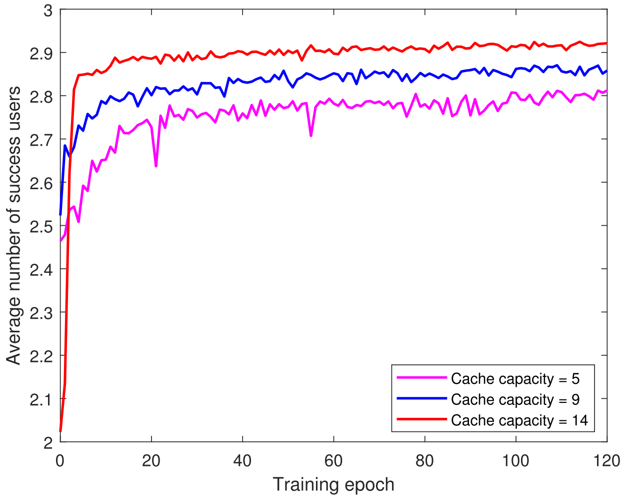


Fig. 7. The performance of dual networks during the training process under different users' cache capacity conditions.

Fig. 7 illustrates the performance of the dual-network models during its training process with different users' cache capacity conditions. The advantage of deploying caching at users' devices is not only associated with better results in terms of the average success user number, but also with the stability in the system performance as shown in the figure.

In our experiments, the training stage of neural networks in Method 2 is done with respect to a specific communication condition of users. We would like to investigate how the trained model behaves when it is applied to a different context such as a different number of users and channel conditions. Regarding this, Fig. 8 and 9 show the difference between the second method and the first method in a different environment than the one where neural networks are trained. Particularly, in these experiments, the number of users is 2 with $|h_1|^2$ and $|h_2|^2$ having their mean to be 1 and 2, respectively, with a variance of 1 and 4, respectively. Since the output from neural networks are fixed to have a dimension of 3, in order to work with 2-user case, we treat the third user as a virtual one who always caches his requests. This issue is a

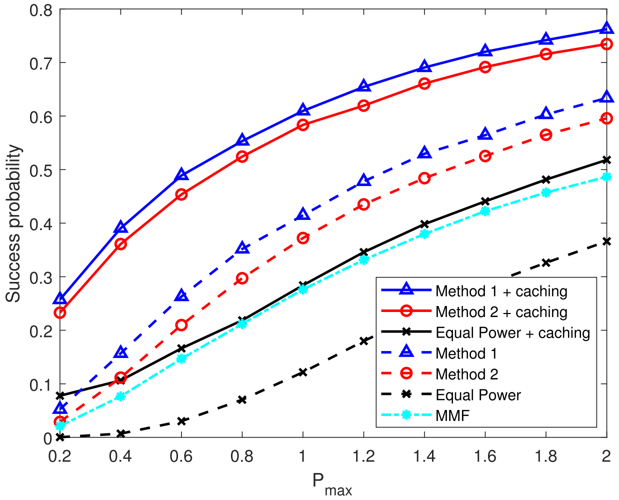


Fig. 8. The adaptability of dual-network prediction model in a comparison to the optimal policy under the variation of communication SNR adjusted by the maximum transmission power $P_{\max}$.

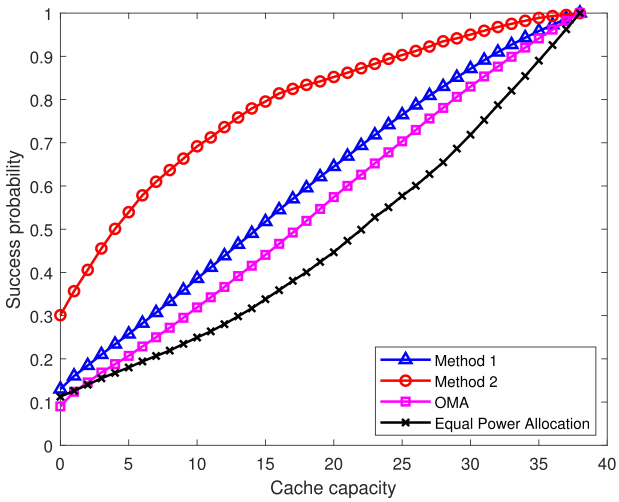


Fig. 10. The performance comparison between Method 1, Method 2, and OMA scheme in a 4-user scenario. $|h_i|^2, \forall i = 1, 2, 3, 4$ follow the exponential distribution with mean 3 and variance 9.

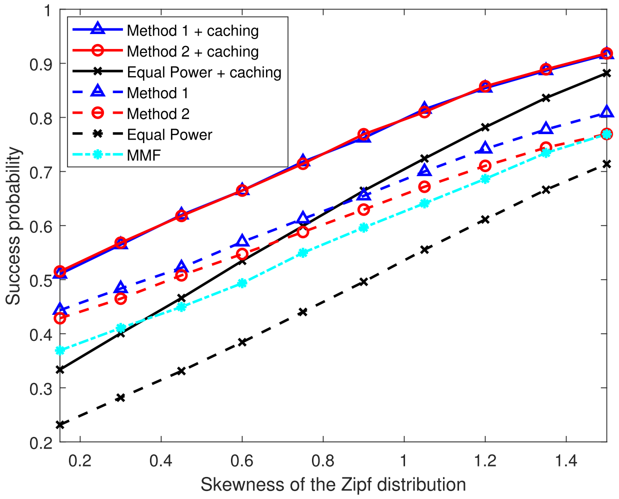


Fig. 9. Under the assumption that the request probability follows Zipf distribution, this figure shows the adaptability of dual-network prediction model in comparison to the optimal policy with respect to the variation of the Zipf's skewness. When the skewness is large, users requests are focused on files with low indexes.

challenge for our prediction model resulted from the learning nature of parametric functions. When learning a general policy to maximize the reward associated with 3 users, the reward associated with the first two users will need to be sacrificed sometimes to achieve the best overall reward. Thus, training a new model to work directly with 2 users will give a better result. However, as mentioned we would like to examine the adaptability of our trained model in a new context. Another important point is that in this situation, the results from the first method is theoretically optimal, hence, is an upper bound for Method 2.

Fig. 8 presents the improvement in the success probability following the enhancement of SNR adjusted by rising $P_{\max}$. In Fig. 9, the request probability of users is assumed to follow Zipf distribution with a certain skewness. This figure presents the effect from the skewness factor. As this factor increases, users' requests concentrate on the files with low indexes. Therefore, the optimal caching policy is to cache file from index 0 to 38 until the maximum capacity is reached. Although being brought to a different and unexpected situation, the prediction from our model is closed to that of the global optimal results, which verifies the effectiveness of our learning model. Furthermore, both of our methods are superior to all the baseline schemes. This is because the simple equal power allocation scheme does not exploit any of system information and the MMF method cannot adapt well to our context which has no instantaneous channel information.

For a clearer comparison between methods, we train another dual-network prediction model for the case of 4 users. Each network of the predictor has 5 layers with dimensions of 16×35, 35×50, 50×35, 35×12 and 12×4. Other setups remain unchanged. The model is trained in a context where $|h_i|^2, \forall i = 1, 2, 3, 4$ follow the exponential distribution with mean 3 and variance 0.1, and the variance is 9 in a context of comparison. The corresponding results are in Fig. 10 which shows that regardless of the change in channel variance, Method 2 still outperforms the others with significant gaps. This is because this method exploits better the available bandwidth. However, Method 2 is more dependent on the system context than Method 1. Making use of orthogonal frequencies to isolate user pairs makes Method 1 simple and easy to be deployed in different contexts.

VI. Conclusion

In this work, we have combined caching and NOMA which are two prominent techniques in future wireless networks. Our analysis has shown that the combination creates another type of cache hit which takes place when users cache the requests of others even without their collaboration. This enables interference cancellation, enhancing the effectiveness of both caching and NOMA. Such an interference-cancellation mechanism in a non-collaborative manner also helps simplify the system implementation, emphasizing that caching and NOMA can be deployed concurrently in future networks in order to boost the system performance further. Although deriving a joint cache placement and transmission strategy is not considered in this work, it is a topic of interest for further study.

In order to optimize the communication quality and concurrently guarantee the fairness among users, our target has been to maximize the probability that all users can successfully decode their desired signals. To achieve this, we have proposed divide-and-conquer-based and deep-learning-based methods. To compare the two methods, the former is simpler and more flexible to the system context with closed-form solutions derived. Although the second method requires system training, it is able to reach higher performance due to better bandwidth usage. In this method, a dual DNN model has been proposed to overcome the noisiness/randomness problem in training data.

Appendix
Proof of Theorem 1

■ 1^{st} case: $\alpha \geq 0.5$.

The conditions (15)-(17) can be respectively rewritten as

$$|h_1|^2 \geq \frac{\epsilon_1\beta_1}{\alpha} \tag{73}$$

$$|h_2|^2 \geq \max\left(\frac{\epsilon_1\beta_2}{(1+\epsilon_1)\alpha-\epsilon_1}, \frac{\epsilon_2\beta_2}{1-\alpha}\right) \tag{74}$$

$$\alpha > 1-\frac{1}{1+\epsilon_1} \tag{75}$$

- 1^{st} subcase: $\frac{\epsilon_1\beta_2}{(1+\epsilon_1)\alpha-\epsilon_1} \geq \frac{\epsilon_2\beta_2}{1-\alpha}$.

The condition for this subcase is equivalent to

$$\alpha \leq 1-\frac{1}{1+\epsilon_1+\frac{\epsilon_1}{\epsilon_2}}. \tag{76}$$

Then, the system success probability is expressed as

$$p_{11}^{C1} = \exp\left(-\frac{\lambda_1\epsilon_1\beta_1}{\alpha} - \frac{\lambda_2\epsilon_1\beta_2}{(1+\epsilon_1)\alpha-\epsilon_1}\right). \tag{77}$$

p_{11}^{C1} is a monotonically increasing function of α, hence, is maximized when

$$\alpha = 1-\frac{1}{1+\epsilon_1+\frac{\epsilon_1}{\epsilon_2}}. \tag{78}$$

- 2^{nd} subcase: $\frac{\epsilon_1\beta_2}{(1+\epsilon_1)\alpha-\epsilon_1} \leq \frac{\epsilon_2\beta_2}{1-\alpha}$.

The condition for this subcase is equivalent to

$$\alpha \geq 1-\frac{1}{1+\epsilon_1+\frac{\epsilon_1}{\epsilon_2}}. \tag{79}$$

With the function g_1^{C1} defined in (23), the success probability is

$$p_{12}^{C1} = \exp\left(-g_1^{C1}(\alpha)\right). \tag{80}$$

Maximizing p_{12}^{C1} is finding α such that g_1^{C1} is minimized. Two stationary points can be obtained by solving $\frac{\partial g_2^{C1}(\alpha)}{\partial\alpha} = 0$, and it is straightforward to verify that the minimum is attained at the first point, $\alpha = 1-\frac{1}{\sqrt{\zeta}+1}$. Combining the obtained result with (79), we have

$$\alpha = z_1^{C1} = \max\left(1-\frac{1}{\sqrt{\zeta}+1}, 1-\frac{1}{1+\epsilon_1+\frac{\epsilon_1}{\epsilon_2}}\right). \tag{81}$$

Combining (81) with the result (78) from previous subcase yields the same result as (81), and this result satisfies $\alpha \geq 0.5$, since $\zeta \geq 1$.

■ 2^{nd} case: $\alpha \leq 0.5$.

The condition (15) and (20) can be rewritten as

$$|h_1|^2 \geq \frac{\epsilon_1\beta_1}{\alpha} \tag{82}$$

$$|h_2|^2 \geq \frac{\epsilon_2\beta_2}{1-(1+\epsilon_2)\alpha} \tag{83}$$

$$\alpha < \frac{1}{1+\epsilon_2} \tag{84}$$

Then, with the function g_2^{C1} defined in (24), the success probability is expressed as

$$p_2^{C1} = \exp\left(-g_2^{C1}(\alpha)\right). \tag{85}$$

Maximizing p_2 is equivalent to minimizing g_2^{C1}. Similarly to the previous case, solving $\frac{\partial g_2^{C1}(\alpha)}{\partial\alpha} = 0$ and combing with $\alpha \leq 0.5$, we have

$$\alpha = z_2^{C1} = \min\left(\frac{1}{1+\epsilon_2}\left(1-\frac{1}{\sqrt{\zeta(1+\epsilon_2)}+1}\right), 0.5\right). \tag{86}$$

To this end, to choose the better result between (81) and (86), we plug them into the corresponding success probability expression to compare. Note that in the case $\alpha \geq 0.5$ there are two subcases. However, at the optimal point (78) of the first subcase, we have $\frac{\epsilon_1\beta_2}{(1+\epsilon_1)\alpha-\epsilon_1} = \frac{\epsilon_2\beta_2}{1-\alpha}$. Therefore, it is sufficient to compare $g_1^{C1}\left(z_1^{C1}\right)$ and $g_2^{C1}\left(z_2^{C1}\right)$ as presented in Theorem 1.

References

[1] M. Vaezi, Z. Ding, and H. V. Poor, *Multiple Access Techniques for 5G Wireless Networks and Beyond*. Cham, Switzerland: Springer, 2018.

[2] V. Chandrasekhar, J. G. Andrews, and A. Gatherer, "Femtocell networks: A survey," *IEEE Commun. Mag.*, vol. 46, no. 9, pp. 59–67, Sep. 2008.

[3] K. N. Doan, T. Van Nguyen, T. Q. S. Quek, and H. Shin, "Content-aware proactive caching for backhaul offloading in cellular network," *IEEE Trans. Wireless Commun.*, vol. 17, no. 5, pp. 3128–3140, May 2018.

[4] Y. Shen, C. Jiang, T. Q. S. Quek, and Y. Ren, "Device-to-device-assisted communications in cellular networks: An energy efficient approach in downlink video sharing scenario," *IEEE Trans. Wireless Commun.*, vol. 15, no. 2, pp. 1575–1587, Feb. 2016.

[5] M. Ji, G. Caire, and A. F. Molisch, "Fundamental limits of caching in wireless D2D networks," *IEEE Trans. Inf. Theory*, vol. 62, no. 2, pp. 849–869, Feb. 2016.

[6] K. N. Doan, T. V. Nguyen, H. Shin, and T. Q. S. Quek, "Socially-aware caching in wireless networks with random D2D communications," *IEEE Access*, vol. 7, pp. 58394–58406, May 2019.

[7] W. Shin, M. Vaezi, B. Lee, D. J. Love, J. Lee, and H. V. Poor, "Non-orthogonal multiple access in multi-cell networks: Theory, performance, and practical challenges," *IEEE Commun. Mag.*, vol. 55, no. 10, pp. 176–183, Aug. 2017.

[8] Z. Ding, Z. Wei, J. Yuan, D. W. K. Ng, and M. Elkashlan, "A survey of downlink non-orthogonal multiple access for 5G wireless communication networks," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 10, pp. 2181–2195, Oct. 2017.

[9] Z. Ding, M. Peng, and H. V. Poor, "Cooperative non-orthogonal multiple access in 5G systems," *IEEE Commun. Lett.*, vol. 19, no. 8, pp. 1462–1465, Aug. 2015.

[10] Y. Saito, Y. Kishiyama, A. Benjebbour, T. Nakamura, A. Li, and K. Higuchi, "Non-orthogonal multiple access (NOMA) for cellular future radio access," in *Proc. IEEE Veh. Technol. Conf.*, Dresden, Germany, Jun. 2013, pp. 1–5.

[11] F. Fang, H. Zhang, J. Cheng, and V. C. M. Leung, "Energy-efficient resource allocation for downlink non-orthogonal multiple access network," *IEEE Trans. Commun.*, vol. 64, no. 9, pp. 3722–3732, Sep. 2016.

[12] Z. Xiao, L. Zhu, J. Choi, P. Xia, and X.-G. Xia, "Joint power allocation and beamforming for non-orthogonal multiple access (NOMA) in 5G millimeter wave communications," *IEEE Trans. Wireless Commun.*, vol. 17, no. 5, pp. 2961–2974, May 2018.

[13] K. N. Doan, W. Shin, M. Vaezi, H. V. Poor, and T. Q. S. Quek, "Optimal power allocation in cache-aided non-orthogonal multiple access systems," in *Proc. IEEE Int. Conf. Commun.*, Kansas City, MO, USA, May 2018, pp. 1–6.

[14] J. Zhu, J. Wang, Y. Huang, S. He, X. You, and L. Yang, "On optimal power allocation for downlink non-orthogonal multiple access systems," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 12, pp. 2744–2757, Dec. 2017.

[15] Z. Ding, P. Fan, G. K. Karagiannidis, R. Schober, and H. V. Poor, "NOMA assisted wireless caching: Strategies and performance analysis," *IEEE Trans. Commun.*, vol. 66, no. 10, pp. 4854–4876, May 2018.

[16] Y. Fu, H. Wang, and C. W. Sung, "Optimal power allocation for the downlink of cache-aided NOMA systems," in *Proc. Int. Conf. Wireless Commun. Signal Process.*, Oct. 2018, pp. 1–6.

[17] Y. Fu, Y. Liu, H. Wang, Z. Shi, and Y. Liu, "Mode selection between index coding and superposition coding in cache-based NOMA networks," *IEEE Commun. Lett.*, vol. 23, no. 3, pp. 478–481, Mar. 2019.

[18] Z. Zhao, M. Xu, Y. Li, and M. Peng, "A non-orthogonal multiple access-based multicast scheme in wireless content caching networks," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 12, pp. 2723–2735, Dec. 2017.

[19] H. Zhang, Y. Qiu, K. Long, G. K. Karagiannidis, X. Wang, and A. Nallanathan, "Resource allocation in NOMA-based fog radio access networks," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 110–115, Jun. 2018.

[20] M. Vaezi, R. Schober, Z. Ding, and H. V. Poor, "Non-orthogonal multiple access: Common myths and critical questions," *IEEE Wireless Commun.*, to be published.

[21] S. Burer and A. N. Letchford, "Non-convex mixed-integer nonlinear programming: A survey," *Surv. Oper. Res. Manage. Sci.*, vol. 17, no. 2, pp. 97–106, 2012.

[22] P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, and A. Mahajan, "Mixed-integer nonlinear optimization," *Acta Numer.*, vol. 22, pp. 1–131, 2013.

[23] J. Lee and S. Leyffer, *Mixed Integer Nonlinear Programming*, vol. 154. New York, NY, USA: Springer, 2012.

[24] Y. Chauvin and D. E. Rumelhart, Eds., *Backpropagation: Theory, Architectures, and Applications*. Hillsdale, NJ, USA: Lawrence Erlbaum Associates, 1995.

[25] D. P. Kingma and L. J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, 2015, pp. 1–13.

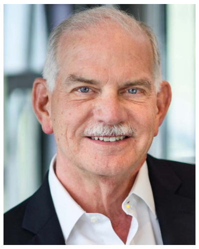
Khai Nguyen Doan (S'18) received the B.E. degree (Hons.) in electronics and telecommunications engineering from the Ho Chi Minh City University of Technology, Vietnam, in 2014. He is currently pursuing the Ph.D. degree with the Singapore University of Technology and Design (SUTD). He was a Research Assistant with SUTD in 2015. His major research interests include network optimization, the application of probability and statistical theories, resource allocation, machine learning, game theory, and signal processing.

Mojtaba Vaezi (S'09–M'14–SM'18) received the B.Sc. and M.Sc. degrees from Amirkabir University of Technology (Tehran Polytechnic) and the Ph.D. degree from McGill University, all in Electrical Engineering. From 2015 to 2018, he was a Post-Doctoral Research Fellow and an Associate Research Scholar with Princeton University, Princeton, NJ, USA. He is currently an Assistant Professor with the ECE Department, Villanova University, Villanova, PA, USA. His research interests include the broad areas of signal processing and machine learning for wireless communications with an emphasis on physical layer security and fifth generation (5G) and beyond radio access technologies. Among his publications in these areas is the book *Multiple Access Techniques for 5G Wireless Networks and Beyond* (Springer, 2019).

Dr. Vaezi was a recipient of several academic, leadership, and research awards, including the McGill Engineering Doctoral Award, the IEEE Larry K. Wilson Regional Student Activities Award in 2013, the Natural Sciences and Engineering Research Council of Canada (NSERC) Postdoctoral Fellowship in 2014, the Ministry of Science and ICT of Korea's best paper award in 2017, and the IEEE Communications Letters Exemplary Editor Award in 2018. He has co-organized five NOMA workshops at IEEE VTC 2017-Spring, Globecom'2017 and 2018, and ICC'2018 and 2019. He is an Editor of IEEE TRANSACTIONS ON COMMUNICATIONS and IEEE COMMUNICATIONS LETTERS.

Wonjae Shin (S'14–M'18) received the B.S. degree (Hons.) and the M.S. degree from the Korea Advanced Institute of Science and Technology (KAIST) in 2005 and 2007, respectively, and the Ph.D. degree from the Department of Electrical and Computer Engineering, Seoul National University (SNU), South Korea, in 2017. From 2007 to 2014, he was a member of Technical Staff with the Samsung Advanced Institute of Technology (SAIT) and Samsung Electronics Co., Ltd., South Korea, where he contributed to next generation wireless communication networks, especially for 3GPP LTE/LTE-advanced standardizations. He was a Visiting Scholar and a Post-Doctoral Research Fellow with Princeton University, Princeton, NJ, USA, from 2016 to 2018. He is currently an Assistant Professor with the Department of Electronics Engineering, Pusan National University, Busan, South Korea. His research interests are in the design and analysis of future wireless communications, such as interference-limited networks.

Dr. Shin received the Best Ph.D. Dissertation Award from the ECE Department, SNU, 2017, Gold Prize from IEEE Student Paper Contest (Seoul Section) 2014, and the Award of the Ministry of Science and ICT of Korea in IDIS-Electronic News ICT Paper contest 2017. He was a co-recipient of the SAIT Patent Award in 2010, Samsung Journal of Innovative Technology in 2010, Samsung HumanTech Paper Contest in 2010, and Samsung CEO Award in 2013. He also received several fellowships, including the Samsung Fellowship Program in 2014 and the SNU Long Term Overseas Study Scholarship in 2016. He was recognized as an Exemplary Reviewer for IEEE WIRELESS COMMUNICATIONS LETTERS in 2014. He is serving as an Associate Editor for IEEE OPEN JOURNAL OF THE COMMUNICATIONS SOCIETY.



H. Vincent Poor (S'72–M'77–SM'82–F'87) received the Ph.D. degree in EECS from Princeton University, Princeton, NJ, USA, in 1977. From 1977 to 1990, he was on the faculty of the University of Illinois at Urbana-Champaign, Urbana, IL, USA. Since 1990, he has been on the faculty at Princeton University, where he is currently the Michael Henry Strater University Professor of Electrical Engineering. From 2006 to 2016, he was the Dean of the Princeton's School of Engineering and Applied Science. He has also held visiting appointments at several other universities, including most recently at Berkeley and Cambridge. His research interests are in the areas of information theory, signal processing, and their applications in wireless networks, energy systems and related fields. Among his publications in these areas is the recent book *Multiple Access Techniques for 5G Wireless Networks and Beyond* (Springer, 2019).

Dr. Poor is a member of the National Academy of Engineering and the National Academy of Sciences, and a Foreign Member of the Chinese Academy of Sciences, the Royal Society, and other national and international academies. He received the Marconi and Armstrong Awards of the IEEE Communications Society in 2007 and 2009, respectively. Recent recognition of his work includes the 2017 IEEE Alexander Graham Bell Medal, the 2019 ASEE Benjamin Garver Lamme Award, a D.Sc. *honoris causa* from Syracuse University awarded in 2017, and a D.Eng. *honoris causa* from the University of Waterloo awarded in 2019.

Hyundong Shin (S'01–M'04–SM'11) received the B.S. degree in electronics engineering from Kyung Hee University (KHU), Yongin-si, South Korea, in 1999, and the M.S. and Ph.D. degrees in electrical engineering from Seoul National University, Seoul, South Korea, in 2001 and 2004, respectively.

During his post-doctoral research at the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, from 2004 to 2006, he was with the Wireless Communication and Network Sciences Laboratory within the Laboratory for Information Decision Systems (LIDS). In 2006, he joined KHU, where he is currently a Professor with the Department of Electronic Engineering. His research interests include quantum information science, wireless communication, and nanonetworks.

Dr. Shin received the Knowledge Creation Award in the field of computer science from the Korean Ministry of Education, Science and Technology in 2010. He received the IEEE Communications Society's Guglielmo Marconi Prize Paper Award in 2008 and William R. Bennett Prize Paper Award in 2012. He served as a Publicity Co-Chair for the IEEE PIMRC in 2018 and a Technical Program Co-Chair for the IEEE WCNC (PHY Track 2009) and the IEEE Globecom (Communication Theory Symposium 2012 and Cognitive Radio and Networks Symposium 2016). He was an Editor for IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS from 2007 to 2012 and IEEE COMMUNICATIONS LETTERS from 2013 to 2015.

Tony Q. S. Quek (S'98–M'08–SM'12–F'18) received the B.E. and M.E. degrees in electrical and electronics engineering from the Tokyo Institute of Technology and the Ph.D. degree in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, MA, USA. He is currently a tenured Associate Professor with the Singapore University of Technology and Design (SUTD). He also serves as the Associate Head of ISTD Pillar and the Deputy Director of the SUTD-ZJU IDEA. He is a coauthor of the *Small Cell Networks: Deployment, PHY Techniques, and Resource Allocation* (Cambridge University Press, 2013) and the *Cloud Radio Access Networks: Principles, Technologies, and Applications* (Cambridge University Press, 2017). His main research interests include the application of mathematical, optimization, and statistical theories to wireless communication, networking, signal processing, and resource allocation problems. Specific current research topics include network intelligence, wireless security, the Internet-of-Things, and big data processing. Dr. Quek has served as a member of the Technical Program Committee and a Symposium Chair in a number of international conferences. He is currently an Elected Member of the IEEE Signal Processing Society and the SPCOM Technical Committee. He received the 2008 Philip Yeo Prize for Outstanding Achievement in Research, the IEEE GLOBECOM 2010 Best Paper Award, the 2012 IEEE William R. Bennett Prize, the 2015 SUTD Outstanding Education Awards–Excellence in Research, the 2016 IEEE Signal Processing Society Young Author Best Paper Award, the 2017 CTTC Early Achievement Award, and the 2017 IEEE ComSoc AP Outstanding Paper Award. He was the 2017 Clarivate Analytics Highly Cited Researcher. He has been actively involved in organizing and chairing sessions. He is a Distinguished Lecturer of the IEEE Communications Society. He was an Executive Editorial Committee Member of IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, an Editor of IEEE TRANSACTIONS ON COMMUNICATIONS, and an Editor of IEEE WIRELESS COMMUNICATIONS LETTERS.