

Attention-based SIC Ordering and Power Allocation for Non-orthogonal Multiple Access Networks

Liang Huang, *Senior Member, IEEE*, Bincheng Zhu, Runkai Nan, Kaikai Chi, *Senior Member, IEEE*, Yuan Wu, *Senior Member, IEEE*

Abstract—Non-orthogonal multiple access (NOMA) emerges as a superior technology for enhancing spectral efficiency, reducing latency, and improving connectivity compared to orthogonal multiple access. In NOMA networks, successive interference cancellation (SIC) plays a crucial role in decoding user signals sequentially. The challenge lies in the joint optimization of SIC ordering and power allocation, a task made complex by the factorial nature of ordering combinations. This study introduces an innovative solution, the Attention-based SIC Ordering and Power Allocation (ASOPA) framework, targeting an uplink NOMA network with dynamic SIC ordering. ASOPA aims to maximize weighted proportional fairness by employing deep reinforcement learning, strategically decomposing the problem into two manageable subproblems: SIC ordering optimization and optimal power allocation. Our approach utilizes an attention-based neural network, which processes instantaneous channel gains and user weights to determine the SIC decoding sequence for each user. A baseline network, serving as a mimic model, aids in the reinforcement learning process. Once the SIC ordering is established, the power allocation subproblem transforms into a convex optimization problem, enabling efficient calculation of optimal transmit power for all users. Extensive simulations validate ASOPA's efficacy, demonstrating a performance closely paralleling the exhaustive method, with over 97% confidence in normalized network utility. Compared to the current state-of-the-art implementation, i.e., Tabu search, ASOPA achieves over 97.5% network utility of Tabu search. Furthermore, ASOPA is two orders magnitude less execution latency than Tabu search when $N = 10$ and even three orders magnitude less execution latency less than Tabu search when $N = 20$. Notably, ASOPA maintains a low execution latency of approximately 50 milliseconds in a ten-user NOMA network, aligning with static SIC ordering algorithms. Furthermore, ASOPA demonstrates superior performance over baseline algorithms besides Tabu search in various NOMA network configurations, including scenarios with imperfect channel state information, multiple base stations, and multiple-antenna setups. Such results underscore ASOPA's robustness and effectiveness, highlighting its ability to excel across various NOMA network environments. The complete source code for ASOPA is accessible at <https://github.com/Jil-Menzerna/ASOPA>.

Index Terms—non-orthogonal multiple access (NOMA), successive interference cancellation (SIC), deep reinforcement learning (DRL), resource allocation.



1 INTRODUCTION

WITH the rapid development of online gaming, augmented and virtual reality, 3-dimensional media, and Internet of Things, wireless network traffic has increased significantly, which is a challenge for orthogonal multiple access (OMA) schemes. To meet the growing demand, the next generation of wireless networks exploits advanced multiple access technologies [1]–[3], including the non-orthogonal multiple access (NOMA) [4], [5] and the rate-splitting multiple access (RSMA) [6]. Combining NOMA with other technologies, such as cognitive radios, unmanned aerial vehicles, mobile edge computing, simultaneous wireless information and power transfer, etc., can bring considerable advantages [7], [8]. Specifically, NOMA gives improved

spectral efficiency, energy efficiency, higher data rates, massive connectivity, and diversity of wireless service [9], [10].

For the uplink power-domain NOMA network, users simultaneously transmit their data over the same frequency resource, so there is inter-user interference due to the broadcast and superposition nature of the wireless medium [11], [12]. At the base station (BS) of uplink NOMA-based networks, the successive interference cancellation (SIC) technique decodes different users' messages from the received signal. Using SIC, the BS sequentially decodes users' messages according to a particular order. When decoding a user's message, the remaining undecoded signal is treated as interference. After decoding, a user's message will be subtracted from the received signal. The procedure continues until all users' messages are decoded according to a specific SIC order. Although different SIC orderings generate the same sum throughput of the NOMA network with a single BS [13], they affect the throughput of each user [14]–[17]. The earlier a user's signal is decoded, the more substantial interference it suffers. When the quality of service of an individual user matters, it is necessary to optimize the SIC order for better performance metrics, e.g., outage probability [18]–[20], latency [21], and energy

- L. Huang, B. Zhu, R. Nan, and K. Chi are with the School of Computer Science and Technology, Zhejiang University of Technology, China (e-mail: {lianghuang, bczhu, rknan, kkchi}@zjut.edu.cn).
- Y. Wu is with the State Key Laboratory of Internet of Things for Smart City, University of Macau, Macau, China, and also with the Department of Computer and Information Science, University of Macau, Macau, China (e-mail: yuanyu@um.edu.mo).

consumption [22], [23].

For a NOMA network, the joint optimization of SIC ordering and resource allocation is a Non-deterministic Polynomial (NP) hard problem [21]. Many researchers decomposed the joint optimization into SIC ordering optimization and resource allocation. For SIC ordering, the total number of decoding orderings is the factorial of the number of users. The exhaustive method obtains the optimal SIC ordering by enumerating all possible SIC orderings and is limited to small-scale scenarios, i.e., with five users [24]. To tradeoff performance and computational complexity, there are two types of heuristic methods for SIC ordering in many works, i.e., static heuristic methods and dynamic heuristic methods. Some researchers [17]–[20], [25]–[32] adopted a static SIC ordering order with respect to a single metric and optimized the resource allocation when considering specific NOMA-based wireless networks. The execution latency of static SIC ordering methods is very low, which is close to the conventional SIC ordering algorithms, i.e., the descending order and ascending order of channel quality. However, these static SIC ordering methods can not cope with complex NOMA wireless scenarios, as finding the corresponding single metric is problematic. Some other works [3], [14], [21], [33], [34] tried to iteratively search for the SIC ordering, e.g., greedy insertion and linear relaxation. While the dynamic heuristic methods apply to most NOMA wireless scenarios and achieve better performance compared to static SIC ordering algorithms, they still have high complexity in the joint optimization problems due to repeatedly optimizing the resource optimization.

Recently, deep learning has emerged as a promising approach for making near-optimal decisions. With a large labelled dataset, it can achieve substantial performance through supervised learning. However, in the joint optimization of SIC ordering and resource allocation, obtaining the optimal solution labels is challenging, particularly in large-scale NOMA wireless networks. In this regard, deep reinforcement learning (DRL) is a suitable technique that can train a model using a dataset without labels. Some recent studies have utilized DRL techniques to efficiently solve computation-intensive resource allocation problems in wireless networks, such as Deep Q-Network [35], [36], actor-critic algorithm [37], [38], deep deterministic policy gradient [39], [40], and proximal policy optimization [41], [42]. Recently, a pioneering deep reinforcement learning-based framework named DROO is introduced to address the hybrid integer-continuous challenge in mobile edge computing [43]. DROO ingeniously splits the primary optimization issue into two subproblems: a zero–one binary off-loading decision and a continuous resource allocation task. These subproblems are then individually managed through a model-free learning module and a model-based optimization module, respectively [44]. Nevertheless, DROO and its subsequent iterations [45], [46], are constrained by their reliance on quantization modules that exclusively produce binary decisions. This limitation becomes particularly evident in their inability to permute SIC ordering for NOMA networks. The permutation complexity for SIC orderings is factorial, presenting a significantly more challenging scenario than binary decisions. This complexity forms the basis of our motivation to develop a solution that adeptly handles

the joint SIC ordering and power allocation problem, aiming to achieve near-optimal performance. This is particularly crucial in meeting the real-time requirements of NOMA networks, where efficiently managing the factorial complexity is essential for optimal system operation.

In this paper, we consider the uplink NOMA with different weighted users. Since transmit power of the NOMA wireless network is typically shared in a best-effort fashion, we aim to ensure fairness across multiple users. We optimize the SIC ordering and users' transmit power to maximize the weighted proportional fairness function. To tackle this challenging problem, a novel Attention-based SIC ordering and power allocation (ASOPA) framework is proposed, which leverages both DRL and optimization theory. To assess the effectiveness of ASOPA, we conduct comparative analyses against a range of baseline algorithms. These comparisons focus on two key metrics: the network utility achieved and the execution duration of ASOPA. Furthermore, to demonstrate the wide-ranging applicability and versatility of ASOPA, the ASOPA is applied with various NOMA network configurations. This extension effectively highlights ASOPA's adaptability across diverse network structures within the NOMA framework, underlining its robustness and practical utility in diverse network conditions.

Our main contributions can be summarized as follows:

- We formulate a joint optimization problem of SIC ordering and power allocation to maximize the weighted proportional fairness on an uplink NOMA wireless network. To solve this problem efficiently, it is decomposed into two subproblems: a SIC ordering subproblem and a power allocation subproblem under the given SIC ordering. This decomposition approach enables us to leverage DRL and convex optimization for optimal performance.
- We propose the ASOPA framework to solve the joint optimization problem. This framework comprises three components: an attention-based actor network, a convex optimization module, and a baseline network. The actor network generates the SIC ordering, while the optimization module allocates users' transmit power. The baseline network is used to train and reinforce the actor network. The ASOPA framework is designed to generate feasible solutions that meet all physical constraints, ensuring optimal performance.
- We conduct comprehensive numerical experiments to validate the effectiveness of the ASOPA framework. The findings reveal that ASOPA attains near-optimal performance, fulfilling the real-time demands of NOMA networks. Notably, the normalized network utility achieved by ASOPA has a confidence interval exceeding 99%, closely mirroring the performance of exhaustive methods. In terms of execution latency, ASOPA is on par with static SIC ordering algorithms, with an approximate duration of 50 ms in a ten-user NOMA network. To highlight ASOPA's broad applicability and versatility, we extend its application to include scenarios with imperfect channel state information (CSI), networks comprising multiple BSs, and systems with multiple-antenna setups.

In each of these diverse environments, ASOPA consistently outperforms the baseline algorithms, showcasing its robustness and effectiveness across different NOMA network configurations.

The rest of this paper is organized as follows. The related work is introduced in Section 2. Section 3 gives the system model and problem formulation. Section 4 introduces ASOPA. The numerical results are given in Section 6. Finally, Section 7 concludes the paper.

2 RELATED WORK

Most of the existing works on NOMA use fixed SIC ordering according to channel conditions. Specifically, the descending order of channel quality is usually used in uplink NOMA [25], [26], and the ascending order of channel quality is generally used in downlink NOMA [27], [28]. However, in many scenarios, using the above fixed ascending or descending order for SIC might not be optimal. To achieve optimal performance, [24] used the exhaustive search to find the optimal decoding order. However, the computational complexity of the exhaustive search is at least $O(N!)$, and its usage is limited to small-scale NOMA wireless scenarios, i.e., less than five users. To balance performance and computational complexity, the following works further optimized the SIC ordering in a NOMA network, which affects different performance metrics, e.g., outage probability [18]–[20], throughput [14]–[17], [32], latency [21], energy consumption [22], [23] and the data rate of a particular user [24]. After elaborately designing the SIC ordering algorithm, Qian *et al.* [21] showed that by optimizing the SIC ordering, the min-max execution latency could be reduced by ten times compared to the best comparison method. Also, [14] showed that optimizing the SIC ordering can get a 48% improvement in the sum rate over the fixed SIC ordering. The above work of SIC ordering optimization can be classified into two types: static SIC ordering and dynamic SIC ordering.

2.1 Static SIC Ordering

In addition to wireless channel quality, some works have proposed using a static decoding order based on other performance metrics specific to certain problems and scenarios. For example, the descending order of received user signal power [17], [19], [20], the descending order of predicted user throughput [29], the decreasing order of channel gain normalized by noise and interference power [30], [31], the ascending order of average channel gain from user to the base station [32], and the ascending order of a user's maximum secrecy throughput [18]. For the uplink NOMA scenario, [18] adopted the ascending order of user's maximum secrecy throughput as the SIC ordering to achieve secrecy transmission in eavesdropper scenarios. [19] used the descending order of received user signal power as the SIC ordering and derived the closed-form expression of the outage probability for a single base station and three-user system. Building on this work, [20] considered the single base station and multiple active user scenario in the case of imperfect channel state information, using the descending order of estimated instantaneous received signal power as

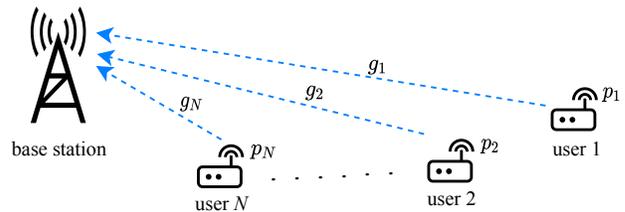


Fig. 1: An uplink NOMA network with one BS and N users.

the SIC ordering. Although these static SIC ordering methods can achieve excellent performance with small execution latency, they may suffer considerable performance degradation on complex NOMA networks due to the difficulty of finding a suitable metric for ordering.

2.2 Dynamic SIC Ordering

In addition to works that use a static SIC ordering based on specific problems and scenarios, a few studies have proposed iteratively searching heuristic algorithms to optimize the SIC ordering. For instance, [3] and [14] introduced binary variables to represent the SIC ordering, solving the problem via variable relaxation with compensation for performance degradation and as an integer linear programming problem, respectively. [33] adapt a permutation-based genetic algorithm to optimize the SIC ordering. [21] used the greedy meta-scheduling technique to develop a low-complexity and easy-to-implement SIC ordering algorithm. This algorithm sequentially inserts users into the existing ordering, tries every possible insertion position for each user, and chooses the position that provides the most significant benefit. [34] utilized a heuristic tabu search to optimize the SIC ordering in an iterative process. At each iteration, tabu search swapped the SIC ordering of any two users and selected the best one. To the best of our knowledge, [34] is the state-of-the-art algorithm for dynamic SIC ordering, it achieves the near-optimal performance, but suffers from high computational complexity due to the large number of iterations. While these dynamic SIC ordering methods apply to most NOMA wireless scenarios and achieve better performance than static SIC ordering, they all involve iterative updates and repeatedly solve resource allocation subproblems, resulting in high computational complexity.

3 SYSTEM MODEL AND PROBLEM FORMULATION

3.1 System Model

As shown in Fig. 1, we consider an uplink NOMA network with a central-located BS and N active users, denoted as a set $\mathcal{N} = \{1, 2, \dots, N\}$, where each user has a single antenna. Users have a stable power supply, and all N users simultaneously transmit their information to the BS by NOMA.

The system time is divided into consecutive slots of equal length, smaller than the channel coherence time. We assume that the wireless channel gain is constant in each time slot and may vary across different slots. Without loss of generality, the slot length is normalized for brevity.

The BS employs SIC in a successive order to decode users' signals. In our framework, we define a function $\xi(n) = i$ and its inverse $\pi(i) = n$, which establish a mapping between a user's index n and its corresponding decoding order i . For instance, $\xi(3) = 2$ indicates that the user 3 is decoded second in the sequence. Conversely, $\pi(2) = 3$ indicates that the second user to be decoded in the order is user 3. This bi-directional mapping functions serve to clearly delineate the relationship between the decoding sequence and the specific users in the network. Hence, the signal-to-interference-plus-noise ratio of the user n can be expressed as

$$\phi_n = \frac{p_n g_n}{\sum_{\xi(n') > \xi(n), \forall n' \in \mathcal{N}} p_{n'} g_{n'} + N_0}, \quad (1)$$

where N_0 is the power of the additive Gaussian noise at the BS, g_n denotes the wireless channel gain between user n and the BS at a tagged time slot, and p_n denotes the transmit power of user n sending its information. So the data rate of user n can be expressed as

$$R_n = B \log_2(1 + \phi_n), \quad (2)$$

where B denotes the communication bandwidth.

We summarize essential notations used throughout this paper in Table 1.

3.2 Problem Formulation

This paper aims to achieve weighted proportional fairness across multiple users. Proportional fairness can be achieved by maximizing individual rates with a logarithmic utility function [48]. In addition, considering users' different priorities, individual weights are assigned to each user to achieve weighted proportional fairness [49]–[51]. Thus, the aim is to maximize the weighted sum of the logarithmic throughput of different users by jointly optimizing the SIC ordering π and the power allocation \mathbf{p} , denoted as the network utility $R(\pi, \mathbf{p})$. This optimization problem can be expressed mathematically as:

$$\mathbf{P0} : R(\pi, \mathbf{p}) = \max_{\pi, \mathbf{p}} \sum_{n=1}^N w_n \ln R_n \quad (3a)$$

$$s.t. 0 < p_n \leq P_n^{max}, \forall n \in \mathcal{N}, \quad (3b)$$

$$\pi \in \Pi, \quad (3c)$$

where w_n is the weight of user n . $\pi = [\pi_1, \pi_2, \dots, \pi_N]$ indicates the SIC order, where we denote $\pi_i = \pi(i)$ for brevity. Π is the permutation set of all possible SIC orderings with size factorial N , represented as $N!$. $\mathbf{p} = [p_1, p_2, \dots, p_N]$ is the power allocation. (3b) is the power constraint for each user n , where P_n^{max} is the maximum power that user n can achieve. (3c) is the constraint for π .

The problem **P0** involves combinatorial optimization and continuous numerical optimization, which is NP-hard. To effectively solve the problem **P0**, we decompose it into the SIC ordering optimization and the optimization of power allocation under the given SIC ordering, as shown in Fig. 2:

- *SIC Ordering*: It is computationally expensive to iteratively search for the optimal SIC ordering from

TABLE 1: Notations

Notation	Definition
N	The number of users
\mathcal{N}	The set of users
g_n	The wireless channel gain between the user n and the BS
p_n	The transmit power of user n
\mathbf{p}	The vector representation of the power allocation
P_n^{max}	The maximum power that user n can achieve
π	The SIC ordering of all users
π^{BL}	The SIC ordering generated by the baseline network
Π	The set of all possible SIC orderings
$\xi(n)$	The order of user n to be decoded
N_0	The power of the additive Gaussian noise at the BS
ϕ_n	The signal-to-interference-plus-noise ratio of the user n
B	The communication bandwidth
R_n	The data rate of user n
w_n	The weight of user n
\mathbf{X}	The representation of all users' information
θ	The parameters of the actor network
θ'	The parameters of the baseline network
\mathbf{E}	The embedding of all users generated by the encoder
$\bar{\mathbf{e}}$	The global information embedding which is the mean of $\mathbf{e}_n, \forall n \in \mathcal{N}$
ℓ_t	The probability of users being selected at iteration t
$\mathbf{q}, \mathbf{K}, \mathbf{V}$	The query, key, and value
d_k	The dimension of \mathbf{q} and \mathbf{k}
d_e	The dimension of each user's embedding \mathbf{e}_n
$S(\theta \mathbf{X})$	The expected objective value for input \mathbf{X} under the network parameters θ
$z_\theta(\pi \mathbf{X})$	The probability of π generated by the actor network θ for the input \mathbf{X}
$R(\pi \mathbf{X})$	The network utility under the given SIC ordering π for \mathbf{X}
$ \tau $	The batch size
τ	The index of sample in a training batch
\mathcal{T}	The set of training batch

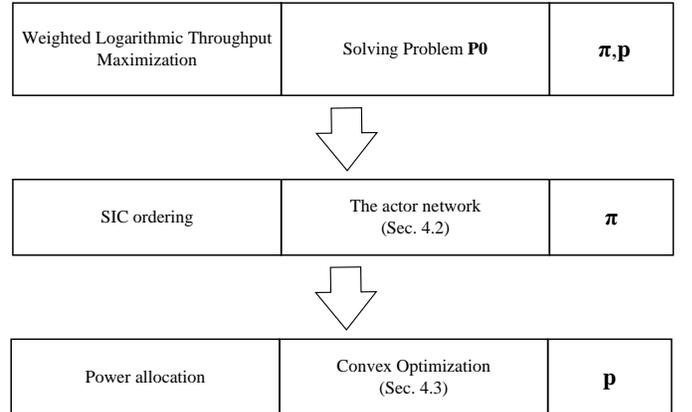


Fig. 2: The two-level optimization structure of solving problem **P0**.

Π at the $N!$ scale. We tell that one SIC ordering outperforms another one by solving the power allocation problems **P1** and comparing their utilities $R(\pi)$. However, classical comparison-based sorting algorithms cannot perform better than $O(n \log n)$ on average [52], which requires repeatedly solving **P1**, resulting in long execution latency. In this paper, the deep reinforcement learning method is adopted to generate the SIC ordering before the power allocation.

- *Power Allocation*: When the SIC ordering π is determined, we only need to solve the power allocation \mathbf{p} , as follows:

$$\mathbf{P1} : R(\pi) = \max_{\mathbf{p}} \sum_{n=1}^N w_n \ln R_n \quad (4a)$$

$$s.t. 0 < p_n \leq P_n^{max}, \forall n \in \mathcal{N}. \quad (4b)$$

We can solve this power allocation sub-problem **P1** by converting it to a convex problem and using the inter-point method.

In the next Section, these two subproblems are solved by taking advantage of DRL and convex optimization.

4 ALGORITHM DESIGN

4.1 Algorithm Overview

Fig. 3 shows the schematics of the proposed ASOPA framework. It uses an encoder-decoder-based actor network θ to generate the SIC ordering sequentially and uses optimization techniques to optimize all users' transmit power. The design of the actor network follows from the pointer network [53], [54] for routing problems. In Fig. 3, the black solid lines depict the inference process of ASOPA, which necessitates real-time network parameters. These parameters include each user's weight w_n , maximum power P_n^{max} , and channel gain g_n . The complete set of users' information, denoted by $\mathbf{X} = \{(w_i, P_i^{max}, g_i)\}_{i \in \mathcal{N}}$, is fed into the actor network to generate the SIC ordering π . Following the determination of the SIC ordering π , we solve the sub-problem **P1** for the optimal power allocation \mathbf{p} through convex optimization. Then, ASOPA outputs the network decision, represented as (π, \mathbf{p}) , based on the instantaneous user information \mathbf{X} .

The red dotted lines in Fig. 3 illustrate the policy update process in ASOPA. We employ a replica of the actor network, referred to as the baseline network θ' , and train the actor network using the REINFORCE algorithm. Each users' information \mathbf{X} is fed into both the actor and baseline networks. This process yields the output SIC orderings π and the baseline SIC orderings π^{BL} . Subsequently, $R(\pi|\mathbf{X})$ and $R(\pi^{BL}|\mathbf{X})$ are obtained by solving problem **P1**, which are used to compute the loss function. The backpropagation method is employed to update the parameters θ of the actor network. The procedures of ASOPA are detailed in the following subsections.

4.2 SIC Ordering

As shown in Fig. 3, the SIC ordering π is generated by the actor network composed of an encoder and a decoder as follows:

- 1) The encoder takes users' information $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$ as input and outputs users' embedding $\mathbf{E} = [\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_N]$ using self-attention layers. The global embedding $\bar{\mathbf{e}}$ is then calculated as the average of \mathbf{E} .
- 2) The decoder generates the SIC ordering in an iterative process. At each iteration t , by utilizing the cross-attention layers, the decoder generates the probability of all users according to $[\bar{\mathbf{e}}, \mathbf{e}_{\pi_{t-1}}]$ and \mathbf{E} . By masking the previously selected users, the probability of the remaining users being selected is calculated using the softmax function, allowing the decoder to determine the current user's index π_t . At the end of each iteration, the decoder updates users selected for masking and takes \mathbf{e}_{π_t} as input to next iteration. It iterates N times to obtain the complete SIC ordering $\pi = [\pi_1, \pi_2, \dots, \pi_N]$.

The integration of an attention scheme in the encoder and an iterative decoding scheme in the decoder empowers ASOPA to effectively manage a varying number of users in NOMA networks. This approach ensures adaptability and responsiveness to user dynamics, maintaining optimal performance across diverse network scenarios.

4.2.1 Encoder

In this section, we describe how the encoder maps user information \mathbf{X} through the neural network into users' embedding \mathbf{E} suitable for subsequent processing. Users' information $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N]$ is first expanded into d_e dimensions by a fully connected feed forward (FF₁) layer, and then it passes a multi-head attention (MHA) layer and a feed forward (FF₂) layer. Both MHA and FF₂ layer have a residual connection and are followed by batch normalization in [53]. Hence, we have:

$$\begin{aligned} \hat{\mathbf{E}} &= \text{BN} \left(\text{FF}_1(\mathbf{X}) + \text{MHA} \left(\text{FF}_1(\mathbf{X}) \right) \right) \\ \mathbf{E} &= \text{BN} \left(\hat{\mathbf{E}} + \text{FF}_2(\hat{\mathbf{E}}) \right). \end{aligned} \quad (5)$$

The details of the multi-head attention mechanism are shown in Appendix A. The length of users' embedding is adaptive to the variable number of users. Each term in the obtained users' embedding $\mathbf{E} = [\mathbf{e}_1, \dots, \mathbf{e}_N]$ takes into account all the users' information.

4.2.2 Decoder

The decoder iteratively generates the users' SIC ordering by utilizing the user embeddings obtained from the encoder. In the decoder, the user embedding \mathbf{E} is initially passed through a linear layer to derive \mathbf{K} , \mathbf{V} , and \mathbf{K}' as follows:

$$\mathbf{K} = \mathbf{W}^K \mathbf{E}, \quad \mathbf{V} = \mathbf{W}^V \mathbf{E}, \quad \mathbf{K}' = \mathbf{W}^{K'} \mathbf{E}, \quad (6)$$

where \mathbf{W}^K , \mathbf{W}^V , and $\mathbf{W}^{K'}$ are matrices of learnable parameters. The global embedding $\bar{\mathbf{e}} = \frac{1}{N} \sum_{n=1}^N \mathbf{e}_n$ is computed to effectively capture the overall network state. The derived values of \mathbf{K} , \mathbf{V} , and \mathbf{K}' , along with $\bar{\mathbf{e}}$, are then utilized in subsequent iterations of the decoder.

As shown in Fig. 3, the decoder performs in an iteration mode and generates an N users' SIC ordering

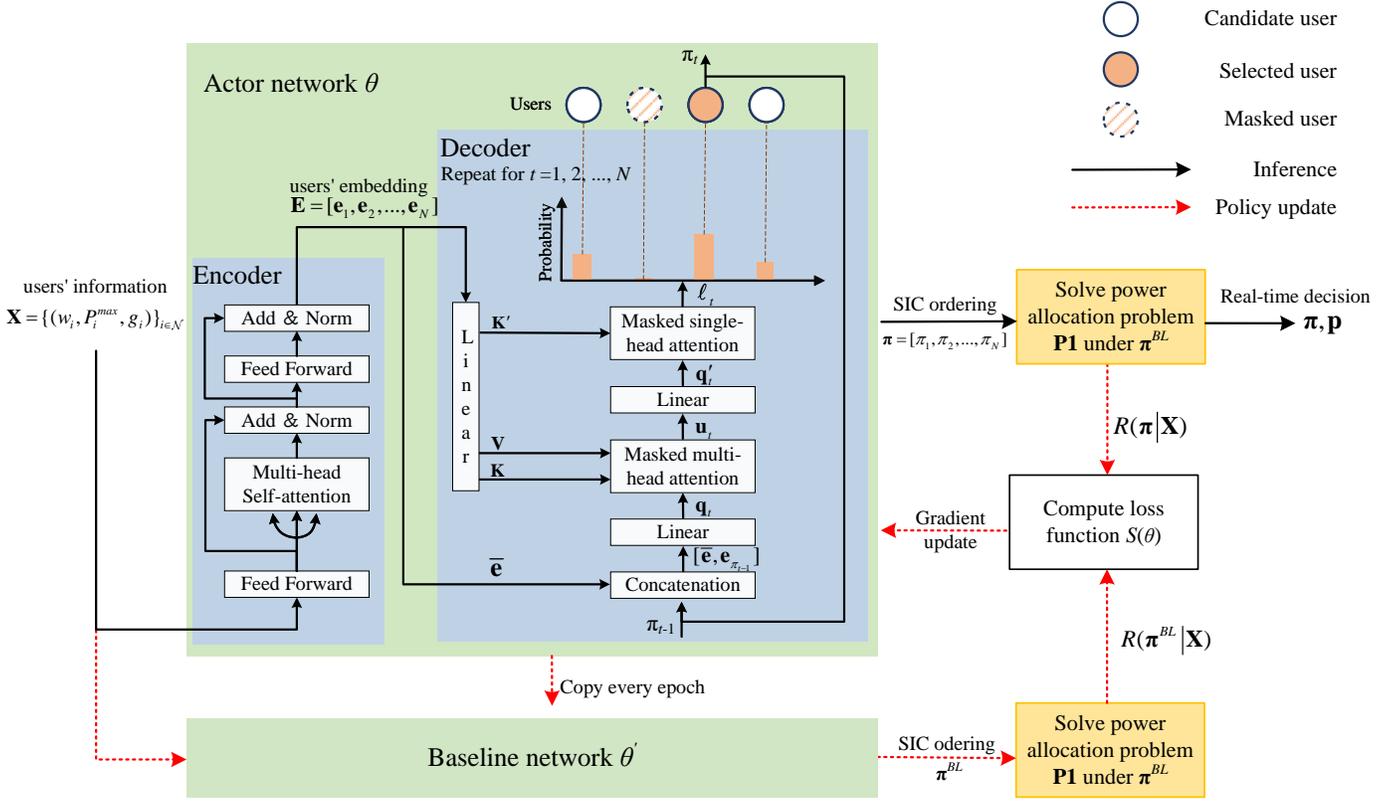


Fig. 3: The schematics of the proposed ASOPA framework.

$\boldsymbol{\pi} = [\pi_1, \pi_2, \dots, \pi_N]$. For each iteration $t \in [1, 2, \dots, N]$, the decoder takes the last decoded user's index π_{t-1} and decides the current decoded user's index π_t .

Firstly, the concatenation module concatenates an input vector $[\bar{\mathbf{e}}, \mathbf{e}_{\pi_{t-1}}]$ that contains the global information and the information of the previously decoded users. $\mathbf{e}_{\pi_{t-1}}$ denotes the previous decoded user's embedding. The embedding $\mathbf{e}_{\pi_{t-1}}$ of the input vector changes with iteration, which captures the user preferences on the SIC ordering. When decoding the first user with no previous decoded user, \mathbf{e}_0 is set as the learnable parameter vector.

Secondly, the masked multi-head attention layer generates a feature vector \mathbf{u}_t that involves the query, keys, and values initially introduced in [55]. The keys \mathbf{K} and values \mathbf{V} are obtained in (6), while the single query is variable input computed as follows:

$$\mathbf{q}_t = \mathbf{W}^Q [\bar{\mathbf{e}}, \mathbf{e}_{\pi_{t-1}}], \quad (7)$$

where \mathbf{W}^Q are learnable parameters matrices. The multi-head attention mechanism is described in Appendix A and omitted for brevity. Then the output of the masked multi-head attention layer can be calculated as

$$\mathbf{u}_t = \text{softmax} \left(\text{mask} \left(\frac{\mathbf{q}_t^T \mathbf{K}}{\sqrt{d_k}} \right) \right) \mathbf{V}, \quad (8)$$

where d_k is the dimension of \mathbf{q} , the softmax function can refer to equation (4) of Appendix A, and the mask function can be expressed as:

$$\text{mask} \left(\frac{\mathbf{q}_t^T \mathbf{k}_i}{\sqrt{d_k}} \right) = \begin{cases} -\infty & \text{if } i \in \{\pi_1, \pi_2, \dots, \pi_{t-1}\}, \\ \frac{\mathbf{q}_t^T \mathbf{k}_i}{\sqrt{d_k}} & \text{otherwise,} \end{cases} \quad (9)$$

where \mathbf{k}_i denotes the i -th element of the keys \mathbf{K} . At each iteration t , the users selected in previous iterations are masked to guard that each user's index appears precisely once in $\boldsymbol{\pi}$.

Thirdly, the single-head attention layer is used to generate the probability of users being selected at time t . ℓ_t can also be considered the similarity between the single query and the keys of the single-head attention layer. The keys \mathbf{K}' is obtained in (6), and the single query \mathbf{q}'_t is computed as follows:

$$\mathbf{q}'_t = \mathbf{W}^{Q'} \mathbf{u}_t, \quad (10)$$

where $\mathbf{W}^{Q'}$ are learnable parameters matrices. Then, ℓ_t can be derived as

$$\ell_t = \text{softmax} \left(\text{mask} \left(\text{clip} \left(\frac{\mathbf{q}'_t^T \mathbf{K}'}{\sqrt{d_k}} \right) \right) \right), \quad (11)$$

where the clip function can clip the result within $[-10, 10]$ by the tanh function to avoid the probability of each selected user being too large or too small [53].

Finally, the selection module selects the user to be decoded based on ℓ_t . In the inference phase, the selection module works in the greedy mode. Specifically, the selection module greedily selects the one with the greatest probability to be the t -th decoded user, as

$$\pi_t = \arg \max_n \ell_{t,n}. \quad (12)$$

Up to now, the decoder operation at iteration t is completed.

Upon repeating the aforementioned steps N times in the decoder, we can compile all π_t to form the SIC decoding order $\boldsymbol{\pi}$:

$$\boldsymbol{\pi} = [\pi_1, \dots, \pi_N]. \quad (13)$$

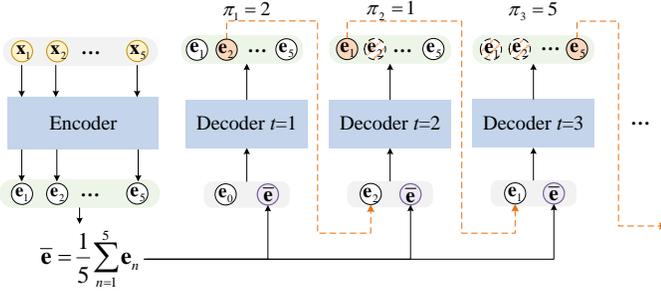


Fig. 4: The process of iteratively determining the SIC ordering in ASOPA.

4.2.3 Step-by-Step SIC Ordering Example

To illustrate the iterative generation of the SIC ordering, we provide an example involving 5 users as shown in Fig. 4.

The encoder takes the five users' information $\mathbf{X} = [x_1, x_2, \dots, x_5]$ as input, and correspondingly calculate the users' embedding $\mathbf{E} = [e_1, e_2, \dots, e_5]$ and the global embedding $\bar{e} = \frac{1}{5} \sum_{n=1}^5 e_n$.

The decoder then iteratively generates the first three SIC orderings of five users as follows:

- 1) In the first iteration ($t = 1$), the decoder inputs $[\bar{e}, e_0]$, computes the probabilities ℓ_t using Equation (11), and selects the user with the highest probability for the first decoding. In this example, user 2 has the highest probability ($\ell_{1,2} = 0.4187$) and is selected as $\pi_1 = 2$.
- 2) In the second iteration ($t = 2$), the decoder takes the global embedding \bar{e} and the embedding of the first decoded user (e_2) as inputs $[\bar{e}, e_2]$, and recalculates the probabilities ℓ_t . As user 2 has already been selected, its probability is masked and set to zero ($\ell_{2,2} = 0$). The highest probability in this iteration is $\ell_{2,1} = 0.4728$, leading to the selection of user 1 as $\pi_2 = 1$.
- 3) In the third iteration ($t = 3$), the decoder inputs $[\bar{e}, e_1]$, and recalculates the probabilities ℓ_t . With the probabilities of the previous selected users masked ($\ell_{3,2}, \ell_{3,1} = 0$), user 5 has the highest probability ($\ell_{3,5} = 0.7953$) in this iteration and thus user 5 is selected as $\pi_3 = 5$.

This procedure is repeated for two more iterations until the SIC decoding order for all five users is determined. The specific probabilities $\ell_{t,n}$ of all users over iterations are shown in Table 2.

4.3 Power Allocation Under Given SIC Ordering

In this subsection, we design a convex transformation algorithm for the power allocation problem under given SIC ordering. Referring to Fig. 3, the actor network of ASOPA only generates SIC ordering without considering transmit power allocation and the corresponding constraints. To obtain the corresponding power allocation and evaluate the SIC ordering, we solve the power allocation subproblem and obtain the achieved network utility as follows.

Upon determining the SIC ordering π , the sub-problem **P1** is addressed to identify the optimal power allocation

satisfying the constraints and subsequently calculate the network utility $R(\pi|\mathbf{X})$ for the specified SIC ordering. Given that the data rate R_n for each user is non-convex, **P1** is inherently a non-convex problem. To tackle this, we employ variable substitution to transform **P1** into a convex problem. The details of this transformation and the proof of its convexity are provided in Appendix B. We solve the transformed version of **P1** using the interior-point method [56] of the CVX solver. The solution of **P1** yields the optimal power allocation \mathbf{p} under the given SIC ordering π satisfying the constraints. Following this, ASOPA outputs the network decision (π, \mathbf{p}) based on the instantaneous user information \mathbf{X} .

Consequently, the network utility $R(\pi|\mathbf{X})$ can be calculated. This calculated utility provides essential feedback on the effectiveness of the SIC ordering under the current policy. As such, the resource allocation module acts as a critic in training the actor network, playing a crucial role in evaluating these generated SIC orderings.

Notice that ASOPA can be migrated to any other NOMA optimization problem with the required resource allocation problem. The extensibility of ASOPA is discussed in following Sec. 5.

4.4 Policy Update

The red dotted lines in Fig. 3 represent the policy update process. For an input instance \mathbf{X} , our goal is to maximize the expected sum of weighted users' logarithmic throughput, as

$$\mathbb{E}_{\pi \sim z_\theta(\cdot|\mathbf{X})} R(\pi|\mathbf{X}). \quad (14)$$

where $z_\theta(\pi|\mathbf{X}) = \prod_{t=1}^N \ell_{t,\pi_t}$ is a measure of the likelihood that the generated SIC decoding order π is the optimal sequence given the current network state \mathbf{X} .

To explore more SIC orderings in the training phase, at the selection module of the decoder network, the user based on the probability ℓ_t at the iteration t is sampled [53] as

$$\pi_t \sim \ell_t. \quad (15)$$

After iterating $t \in [1, 2, \dots, N]$, we obtain the SIC ordering π . Then, the gradient of the actor network θ can be formulated by the REINFORCE algorithm [57] as

$$\mathbb{E} \left(R(\pi|\mathbf{X}) \nabla_\theta \log z_\theta(\pi|\mathbf{X}) \right). \quad (16)$$

However, the REINFORCE algorithm may be of high variance and thus produce slow learning [59]. To improve the performance of DRL, the REINFORCE algorithm with baseline [59] is adopted in this paper. As illustrated in Fig. 3, the baseline network θ' , which uses the actor network parameters from the previous epoch, serves as a baseline to generate SIC orderings π^{BL} and subsequently calculates the network utility $R(\pi^{BL}|\mathbf{X})$. The difference between $R(\pi^{BL}|\mathbf{X})$ and $R(\pi|\mathbf{X})$ is utilized to train the current actor network θ .

Consequently, the gradient of the REINFORCE algorithm with baseline can be expressed and then approximated by Monte Carlo sampling as

$$\begin{aligned} & \mathbb{E} \left(\left(R(\pi|\mathbf{X}) - R(\pi^{BL}|\mathbf{X}) \right) \nabla_\theta \log z_\theta(\pi|\mathbf{X}) \right) \\ & \approx \frac{1}{|\tau|} \sum_{i=1}^{|\tau|} \left(\left(R(\pi_i|\mathbf{X}_i) - R(\pi_i^{BL}|\mathbf{X}_i) \right) \nabla_\theta \log z_\theta(\pi_i|\mathbf{X}_i) \right). \end{aligned} \quad (17)$$

TABLE 2: Case study - Inference of the decoder

	Input	Output					Decoded user π_t
		user 1	user 2	ℓ_t	user 4	user 5	
$t = 1$	$[\bar{\mathbf{e}}, \mathbf{e}_0]$	0.2727	0.4187	0.0223	0.0295	0.2568	$\pi_1 = 2$
$t = 2$	$[\bar{\mathbf{e}}, \mathbf{e}_2]$	0.4728	0 (masked)	0.0399	0.0528	0.4345	$\pi_2 = 1$
$t = 3$	$[\bar{\mathbf{e}}, \mathbf{e}_1]$	0 (masked)	0 (masked)	0.0893	0.1154	0.7953	$\pi_3 = 5$
$t = 4$	$[\bar{\mathbf{e}}, \mathbf{e}_5]$	0 (masked)	0 (masked)	0.4520	0.5480	0 (masked)	$\pi_4 = 4$
$t = 5$	$[\bar{\mathbf{e}}, \mathbf{e}_4]$	0 (masked)	0 (masked)	1	0 (masked)	0 (masked)	$\pi_5 = 3$

where $|\tau|$ is the batch size, \mathbf{X}_i is the i -th input, π_i is the SIC ordering produced by the actor network based on (15), and π_i^{BL} is the i -th SIC ordering produced by the baseline network. In practice, the expectation in Equation (17) is approximated by averaging over a batch of uniformly sampled input instances $\{\mathbf{X}_\tau\}_{\tau \in \tau'}$, where τ represents the index set of these sampled instances. After obtaining the gradients (17), Adam [58] is applied as the optimizer to update the actor network's parameters θ .

Our reinforcement learning algorithm for training the actor network is outlined in Algorithm 1. To facilitate this process, an empty memory with limited capacity is established to store past samples. As new samples are received in each time slot, policy updates are executed infrequently. For every policy update, a random batch of samples is selected from this memory to train the actor network. The baseline network, on the other hand, undergoes updates every epoch which consists of M times policy update. This update process involves copying the parameters from the actor network to the baseline network, as denoted by $\theta' = \theta$. This systematic approach ensures continuous adaptation and optimization of the actor network's performance based on the latest data.

4.5 Computation Complexity

ASOPA operates through two distinct processes: the inference process and the policy update process. During each time slot, ASOPA's inference process is activated to generate Successive Interference Cancellation (SIC) orderings and power allocations. Contrarily, the policy update process can be carried out less frequently, and can also be executed in parallel on different servers in practical applications. Given the crucial role of inference delay in determining the feasibility of field deployment, the inference complexity of ASOPA is a key area of interest.

The inference process is detailed in lines 4-5 of Algorithm 1. Line 4 involves generating the SIC ordering from the actor network, where the computational complexity is primarily driven by matrix multiplications in the attention mechanism. The complexity of interactions between the query, key, and value is $O(HDN^2)$, with H representing the number of heads in multiple attention mechanisms, D the dimension of these components, and N the number of users. Line 5 addresses the generation of power allocation by solving subproblem **P1**, which is reformulated into a convex problem **P2** and solved using the cvxopt solver with the Interior Point Method. The computational complexity of this method is $O(N^{3.5})$ [60]. Therefore, the overall inference complexity of ASOPA is $O(N^{3.5})$. In Section 6.3, we will numerically demonstrate that ASOPA meets the real-time requirements of NOMA networks.

Algorithm 1: Training ASOPA

input : Users' weights, maximum transmit power, and channel gains at each time slot s
 $\mathbf{X}_s = \{(w_i, P_i^{max}, g_i)\}_{i \in \mathcal{N}}$, the training interval δ_T of the actor network, the update epoch δ_E of the baseline network;
output: SIC order π and power allocation \mathbf{p} ;

- 1 Initialize the actor network's parameters θ ;
- 2 Initialize the baseline network's parameters $\theta' \leftarrow \theta$;
- 3 **for** $epoch = 1, \dots, E$ **do**
- 4 Generate the SIC ordering π for \mathbf{X} of the epoch from the actor network based on (15);
 // Inference of ASOPA for each sample
- 5 Obtain \mathbf{p} and $R(\pi|\mathbf{X})$ for \mathbf{X} of the epoch by solving problem **P1**;
- 6 **for** $batch = 1, \dots, M$ **do**
- 7 Uniformly sample a batch of samples $\{\mathbf{X}_\tau\}_{\tau \in \tau}$ from previous samples of the epoch; // Infrequently policy update of ASOPA can be executed in parallel on different servers in practical applications
- 8 Generate the SIC ordering $\{\pi_\tau\}_{\tau \in \tau}$ from the actor network based on (15);
- 9 Generate the baseline SIC ordering $\{\pi_\tau^{BL}\}_{\tau \in \tau}$ from the baseline network based on (12);
- 10 Obtain $\{(R(\pi_\tau|\mathbf{X}_\tau), R(\pi_\tau^{BL}|\mathbf{X}_\tau))\}_{\tau \in \tau}$ by solving problem **P1**;
- 11 Calculate the gradient $\nabla_\theta S(\theta)$ from (17) based on $\{(R(\pi_\tau|\mathbf{X}_\tau), R(\pi_\tau^{BL}|\mathbf{X}_\tau))\}_{\tau \in \tau}$;
- 12 Update the actor network's parameters θ using the Adam optimization algorithm based on the calculated gradients ∇_θ ;
- 13 **end**
- 14 Update the baseline network's parameters $\theta' \leftarrow \theta$;
- 15 **end**

5 EXTENSION SCENARIOS

ASOPA can be easily extended to various NOMA scenarios. To migrate to a new scenario, only the inputs of the actor network and baseline network need to be modified, while the structure of the actor network used to determine the SIC ordering remains unchanged. Correspondingly, the resource allocation subproblems are adjusted and re-solved for specific NOMA scenarios, ensuring optimal performance and efficiency under different network conditions.

In this section, we evaluate the extension scenarios of ASOPA, including NOMA networks with imperfect CSI, NOMA networks with multiple-antenna setups, and

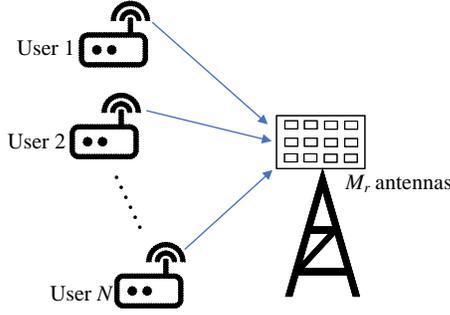


Fig. 5: A NOMA network with multiple-antenna setup at the BS.

NOMA networks with multiple-BS setups.

5.1 Multiple Antenna

In this subsection, we evaluate ASOPA in NOMA networks with multiple-antenna setups, since multiple-antenna technology has advantages in improving spectrum and energy efficiency [61], [62]. The uplink multiple-antenna system consists of a M_r antennas BS and N single-antenna users as shown in Fig. 5. The received signal $\mathbf{Y} \in \mathbb{C}^{M_r \times 1}$ of BS is

$$\mathbf{Y} = \mathbf{H}\mathbf{S} + \mathbf{n}, \quad (18)$$

where $\mathbf{H} \in \mathbb{C}^{M_r \times N}$ denotes the channel state from users to the receive antennas of BS, $\mathbf{S} = [s_1, s_2, \dots, s_N]^T$ denotes the transmit signal matrix, and $\mathbf{n} \sim \mathcal{CN}(\mathbf{0}, \sigma^2 \mathbf{I})$ represents the additive white Gaussian noise at the BS side. $\mathcal{CN}(0, \sigma^2)$ denotes the complex Gaussian distribution with mean zero and the variance σ^2 . The linear equalization, such as zero-forcing (ZF) or minimum-mean-square-error (MMSE), is used in multiple-antenna scenarios to symbol-by-symbol detection. According to the states of multiple-antenna scenario, ASOPA correspondingly generates the SIC ordering and power allocation. The specific steps are as follows.

The equalization matrix $\mathbf{V} \in \mathbb{C}^{N \times M_r}$ for ZF [61] or MMSE [62] can be expressed as [63]

$$\mathbf{V} = \begin{cases} \mathbf{P}\mathbf{H}^H(\mathbf{H}\mathbf{P}\mathbf{H}^H + \sigma\mathbf{I})^{-1} & \text{when MMSE,} \\ \mathbf{H}^H(\mathbf{H}\mathbf{H}^H)^{-1} & \text{when ZF,} \end{cases} \quad (19)$$

where \mathbf{P} denotes the diagonal matrix $\text{diag}(p_1, p_2, \dots, p_N)$.

Then the received estimated signal is

$$\hat{\mathbf{S}} = \mathbf{V}\mathbf{Y} = \mathbf{V}\mathbf{H}\mathbf{S} + \mathbf{V}\mathbf{n}, \quad (20)$$

and the estimated value of user n is

$$\hat{s}_n = \sqrt{p_n} \mathbf{v}_n \mathbf{h}_n s_n + \sum_{n' \neq n} \sqrt{p_{n'}} \mathbf{v}_n \mathbf{h}_{n'} s_{n'} + \mathbf{v}_n \mathbf{n}, \quad (21)$$

where $\mathbf{h}_n \in \mathbb{C}^{M_r \times 1}$ denotes the channel states from user n to the receive antenna of the BS, and $\mathbf{v}_n \in \mathbb{C}^{1 \times M_r}$ denotes the n -th row of \mathbf{V} , which can be given by

$$\mathbf{v}_n = \begin{cases} p_n \mathbf{h}_n^H \left(\sum_{\xi(n') \geq \xi(n), \forall n' \in \mathcal{N}} p_{n'} \mathbf{h}_{n'} \mathbf{h}_{n'}^H + \sigma \mathbf{I} \right)^{-1} & \text{when MMSE,} \\ \mathbf{h}_n^H (\mathbf{h}_n \mathbf{h}_n^H)^{-1} & \text{when ZF,} \end{cases} \quad (22)$$

According to the estimated value of user n , its achieved transmit rate can be calculated by

$$R_n = \log_2 \left(1 + \frac{|\mathbf{v}_n \mathbf{h}_n|^2 p_n}{\sum_{\xi(n') > \xi(n), \forall n' \in \mathcal{N}} |\mathbf{v}_n \mathbf{h}_{n'}|^2 p_{n'} + |\mathbf{v}_n|^2 \sigma^2} \right). \quad (23)$$

When using ZF method, the equalization matrix \mathbf{V} in (22) is independent of \mathbf{p} , so that the power allocation problem can be transformed into a convex as appendix E and solved by the CVX solver.

However, when using the MMSE method, the equalization matrix \mathbf{V} depends on the variable \mathbf{p} , making the power allocation problem of MMSE intractable and non-convex. To address this difficulty, alternative optimization is employed to decompose the non-convex problem into two subproblems: one for \mathbf{V} and one for \mathbf{p} . When \mathbf{V} is fixed, the power allocation problem can be transformed into a convex problem, as shown in Appendix E. Therefore, the alternative optimization starts with an initial power allocation to calculate the corresponding equalization matrix. Using this equalization matrix, it then applies the convex method to determine a new power allocation. With this updated power allocation, the equalization matrix is recalculated. This process repeats until the difference between successive power allocations is smaller than the set threshold.

Different from the single antenna scenario, the channel gain between the BS and a user n is a complex value, whose real and imaginal parts are denoted as $\mathbf{h}_n^r = \{h_{1,n}^r, \dots, h_{M_r,n}^r\}$ and $\mathbf{h}_n^c = \{h_{1,n}^c, \dots, h_{M_r,n}^c\}$, respectively. To tackle multiple-antenna scenarios, ASOPA modifies its input from $\mathbf{X} = \{(w_n, P_i^{max}, g_n)\}_{n \in \mathcal{N}}$ to $\mathbf{X} = \{(w_n, P_n^{max}, \mathbf{h}_n^r, \mathbf{h}_n^c)\}_{n \in \mathcal{N}}$. The rest of the ASOPA structure remains the same as one illustrated in Fig. 3.

5.2 Imperfect Channel

In this subsection, we assess the impact of estimation errors on the performance of ASOPA. For the perfect CSI scenario, the channel gain is expressed as $g_n = \bar{g}_n |\alpha_n|^2$, where $\bar{g}_n = A_d \left(\frac{3 \cdot 10^8}{4\pi f_c b_n} \right)^{b_e}$ and $\alpha_n \sim \mathcal{CN}(0, 1)$ account for path loss power gain and the Rayleigh fading channel coefficient between BS and n -th user, respectively. Since the path loss coefficient are large-scale fading factors and are slowly varying, we assume that the path loss coefficient \bar{g}_n between BS and each user can be estimated perfectly. However, in dynamic and complex wireless environments, accurately acquiring time-varying Rayleigh fading channel gains is challenging. Following the approaches in [64], [65], the Rayleigh fading channel gain is modeled as

$$\alpha_n = \hat{\alpha}_n + \epsilon_n \quad (24)$$

where α_n is the realistic Rayleigh fading channel coefficient between BS and n -th user, $\hat{\alpha}_n \sim \mathcal{CN}(0, 1 - \sigma_\epsilon^2)$ denotes the estimated channel coefficient, and $\epsilon_n \sim \mathcal{CN}(0, \sigma_\epsilon^2)$ is the estimated error. Note that the parameter σ_ϵ^2 indicates the quality of channel estimation, and keeps constant as [66], [67]. We assume that $\hat{\alpha}_n$ and ϵ_n are uncorrelated.

If the perfect CSI is known, the maximum achievable data rate between BS and n -th can be written as

$$c_n = W \log_2(1 + \phi_n) \quad (25)$$

where

$$\phi_n = \frac{p_n |\alpha_n|^2 \bar{g}_n}{\sum_{\xi(n') > \xi(n), \forall n' \in \mathcal{N}} p_{n'} |\alpha_n|^2 \bar{g}_{n'} + N_0}. \quad (26)$$

In (26), ϕ_n denotes the signal-to-interference-plus-noise ratio (SINR) of the user n . In practice, the BS can only obtain the estimated fading channel coefficient $\hat{\alpha}_n$. The scheduled data rate with imperfect CSI can be expressed as

$$r_n = W \log_2(1 + \hat{\phi}_n) \quad (27)$$

where

$$\hat{\phi}_n = \frac{p_n |\hat{\alpha}_n|^2 \bar{g}_n}{\sum_{\xi(n') > \xi(n), \forall n' \in \mathcal{N}} p_{n'} |\hat{\alpha}_n|^2 \bar{g}_{n'} + N_0}. \quad (28)$$

However, the scheduled data rate with imperfect CSI may easily exceed the maximum achievable data rate, i.e., $r_n > c_n$. To measure the performance of this case, we introduce outage probability as a metric [68], [69]. Therefore, the weighted proportional fairness function with outage probability can be expressed as $\sum_{n=1}^N w_n \ln r_n \Pr[r_n \leq c_n | \hat{\alpha}_n]$. $\Pr[r_n \leq c_n | \hat{\alpha}_n]$ denotes the probability of a case when the scheduled data rate r_n is less than or equal to the maximum data rate c_n under the estimated channel coefficient $\hat{\alpha}_n$. The optimization problem can be reformulated as

$$\max_{\boldsymbol{\pi}, \mathbf{p}} \sum_{n=1}^N w_n \ln r_n \Pr[r_n \leq c_n | \hat{\alpha}_n] \quad (29a)$$

$$s.t. \Pr[c_n < r_n | \hat{\alpha}_n] \leq \epsilon_{out}, \forall n \in \mathcal{N}, \quad (29b)$$

$$0 < p_n \leq P_n^{max}, \forall n \in \mathcal{N}, \quad (29c)$$

$$\boldsymbol{\pi} \in \Pi, \quad (29d)$$

where (29b) is introduced to satisfy the channel outage probability requirement ϵ_{out} for all users in the imperfect CSI scenario. Due to the probability constraints (29b), this problem (29) turns into a non-convex problem and cannot easily be optimally solved in polynomial time [68]. To tackle this problem efficiently, we transform the probabilistic mixed problem into a non-probability problem as

$$\max_{\boldsymbol{\pi}, \mathbf{p}} \sum_{n=1}^N w_n \ln(1 - \epsilon_{out}) \tilde{r}_n \quad (30a)$$

$$s.t. 0 < p_n \leq P_n^{max}, \forall n \in \mathcal{N}, \quad (30b)$$

$$\boldsymbol{\pi} \in \Pi, \quad (30c)$$

where $\tilde{r}_n = W \log_2(1 + \tilde{\phi}_n)$, and the transformed SINR $\tilde{\phi}_n$ can be expressed as

$$\tilde{\phi}_n = \frac{\epsilon_{out} F_{|\hat{g}_n|^2}^{-1}(\epsilon_{out}/2) p_n}{\epsilon_{out} \sigma_\epsilon^2 + \sum_{\xi(n') > \xi(n), \forall n' \in \mathcal{N}} 2(|\hat{g}_{n'}|^2 + \sigma_\epsilon^2) p_{n'}}, \quad (31)$$

where $F_{|\hat{g}_n|^2}^{-1}(\epsilon_{out}/2)$ denotes the inverse cumulative distribution function of a noncentral chi-square random variable with 2 degrees of freedom and non-centrality parameter $2|\hat{g}_n|^2/\sigma_\epsilon^2$. The details of the probabilistic mixed problem transformation are shown in Appendix D.

Notice that once the SIC ordering is determined, the power allocation problem of (61) can be transformed into

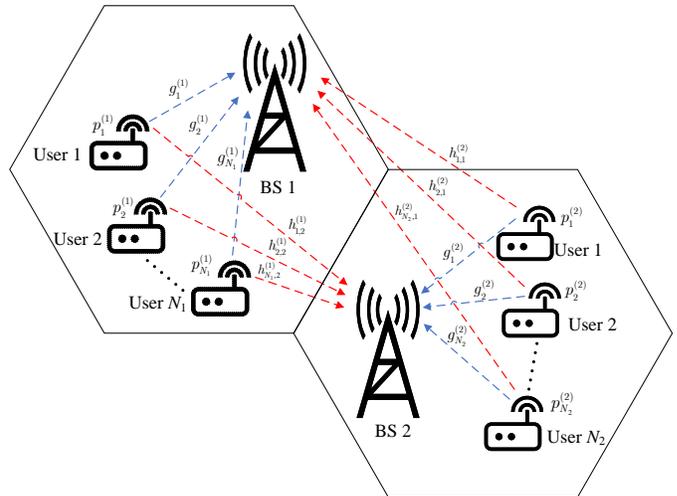


Fig. 6: The system model of the dual-BS NOMA networks.

a convex problem as well as Appendix B and solved by the CVX solver. Thus, ASOPA can be applied to solve it by simply modifying its input from $\mathbf{X} = \{(w_n, P_i^{max}, g_n)\}_{n \in \mathcal{N}}$ to $\mathbf{X} = \{(w_n, P_n^{max}, |\hat{\alpha}_n|^2 \bar{g}_n)\}_{n \in \mathcal{N}}$

5.3 Multiple BS

In this subsection, we assess the performance of ASOPA in NOMA networks with multiple BSs, represented by the set \mathbf{B} , each containing N_b users where $b \in \mathbf{B}$. In scenarios involving multiple BSs, each user experiences inter-cell interference from users linked to other BSs. To quantify this interference, the channel gain from a user n in BS b to another BS b' is defined as $h_{n,b'}^{(b)}$. Specifically, the superscript (b) indicates that user n is associated with BS b .

To accommodate the multiple-BS scenario, it is straightforward to adjust the input of ASOPA to $\mathbf{X} = \{(w_n^{(b)}, P_{n,max}^{(b)}, g_n^{(b)}, \mathbf{h}_n^{(b)})\}_{n \in \mathcal{N}_b, b \in \mathbf{B}}$, where $\mathbf{h}_n^{(b)} = \{h_{n,b'}^{(b)}\}_{b' \in \mathbf{B} \setminus \{b\}}$. The resource allocation problem is still convex and can be solved using the CVX solver. The detailed setup of the multiple-BS scenario is in Appendix E.

The addition of inter-cell interference, however, adds complexity to the SIC ordering problem. In ASOPA, the next decoding user is iteratively chosen based on the highest probability from Equation (12), under the assumption that all users are within the same BS. However, in scenarios involving multiple BSs, the SIC decoding order is specific to each BS. Comparing the probabilities of users from different BSs lacks physical insight, as such comparisons are not meaningful in this context. To overcome this issue, we introduce enhance the decoding process by introducing an additional masking mechanism in the decoder. This mechanism allows for the generation of appropriate SIC orderings for users across all BSs. Specifically, the decoder generates the SIC ordering for users associated with the current BS while effectively masking the users of other BSs. This approach ensures efficient decoding order determination in multi-BS NOMA networks without needing to modify Equation (12).

To demonstrate the effectiveness of the enhanced mask mechanism in ASOPA, let's consider a case study outlined in

TABLE 3: Case study - The mask mechanism in ASOPA in a dual-BS NOMA network

	Input	Output						
		$\ell_t^{(1)}$			$\ell_t^{(2)}$			Decoded user $\pi_t^{(b)}$
		user 1	user 2	user 3	user 1	user 2	user 3	
$t = 1$	$[\bar{e}, \mathbf{e}_0]$	0.3997	0.2890	0.3114	0(masked)	0 (masked)	0 (masked)	$\pi_1^{(1)} = 1$
$t = 2$	$[\bar{e}, \mathbf{e}_1^{(1)}]$	0 (masked)	0.4839	0.5161	0 (masked)	0 (masked)	0 (masked)	$\pi_2^{(1)} = 3$
$t = 3$	$[\bar{e}, \mathbf{e}_3^{(1)}]$	0 (masked)	1	0 (masked)	0 (masked)	0 (masked)	0 (masked)	$\pi_3^{(1)} = 2$
$t = 4$	$[\bar{e}, \mathbf{e}_2^{(1)}]$	0 (masked)	0 (masked)	0 (masked)	0.0731	0.4698	0.4571	$\pi_1^{(2)} = 2$
$t = 5$	$[\bar{e}, \mathbf{e}_2^{(2)}]$	0 (masked)	0 (masked)	0 (masked)	0.0407	0 (masked)	0.9593	$\pi_2^{(2)} = 3$
$t = 6$	$[\bar{e}, \mathbf{e}_3^{(2)}]$	0 (masked)	0 (masked)	0 (masked)	1	0 (masked)	0 (masked)	$\pi_3^{(2)} = 1$

Table 3. As depicted in Fig. 6, we consider a dual-BS NOMA network, where each BS contains three users, as $N_b = 3$ for all $b \in \mathbf{B} = \{1, 2\}$. Consequently, it takes six iterations for ASOPA to establish the SIC ordering for all users. Utilizing the mask mechanism, ASOPA initially decodes the users in BS 1 during iterations $t = 1, 2$, and 3, and then shifts to decoding users in BS 2 for iterations $t = 4, 5$, and 6. In the first iteration ($t = 1$), the algorithm calculates the probabilities $\ell_1^{(1)}$ for users in BS 1 from Equation (12) and simultaneously masks the probabilities of users in BS 2 by setting $\ell_1^{(2)} = 0$. Given that user 1 in BS 1 has the highest probability of 0.3997, it is selected as the first decoded user, $\pi_1^{(1)} = 1$. In the next two iterations, users in BS 2 remain masked, indicated by $\ell_2^{(2)} = 0$ and $\ell_3^{(2)} = 0$. Conversely, when decoding the SIC ordering for users in BS 2 during iterations $t = 4, 5$, and 6, the users in BS 1 are masked with $\ell_t^{(1)} = 0$. This mechanism enhances the efficiency and accuracy of ASOPA in multi-BS NOMA networks by systematically focusing on one BS at a time, thereby streamlining the decoding process.

6 NUMERICAL RESULTS

In this section, we evaluate the proposed ASOPA algorithm through simulations in uplink NOMA networks. In these simulations, users are uniformly deployed within a 100-meter radius circle, with a BS at the center. The average channel gain, \bar{g}_n , adheres to the free-space path loss model, following $\bar{g}_n = A_d \left(\frac{3 \cdot 10^8}{4\pi f_c b_n} \right)^{b_e}$ [43], where $A_d = 4.11$ represents the antenna gain, $f_c = 915$ MHz is the carrier frequency, b_n is the distance between each user and the BS, and $b_e = 2.8$ is the path loss exponent. Each user n 's wireless channel gain, g_n , is modeled as a Rayleigh fading channel, expressed as $g_n = \bar{g}_n |\alpha_n|^2$, with $|\alpha_n|^2$ being an independent random channel fading factor following an exponential distribution with unit mean. The system parameters include a bandwidth B of 1 MHz and a noise power spectral density of -174 dBm/Hz. Each user's maximum power is capped at $P_n^{max} = 1$ Watt, and the user weight w_n is chosen from the set $\{1, 2, 4, 8, 16, 32\}$.

For the neural network training, the samples \mathbf{X} arrive in each time slots and are stored in a replay memory of size 1280. The number of users N in each sample varies uniformly between 5 and 10. The batch size for once policy update is set to $|\tau| = 64$, and each training epoch consists of $M = 20$ times policy updates. After each training epoch, the baseline network updates its parameters θ' . The learning rate for the Adam optimizer is set at $1e-4$, and the embedding dimension for users in the actor network is $d_e = 128$.

The simulations are carried out on a desktop with an Intel Core i7-10700 2.9 GHz CPU, 32 GB memory, and an NVIDIA GeForce RTX 3060 Ti GPU, ensuring robust computational performance. The source code for ASOPA is accessible at <https://github.com/Jil-Menzerna/ASOPA>.

6.1 Convergence Performance

In Fig. 7, we evaluate the effect of different parameters on the convergence performance of ASOPA, including different learning rates, batch sizes, and embedding dimensions.

Fig. 7(a) shows the effect of different learning rates. We can see that a significant learning rate ($1e-2$ or $1e-3$) causes the algorithm to converge to a local optimum, but a small learning rate ($1e-5$) results in slow convergence. Hence, the learning rate is set as $1e-4$.

Fig. 7(b) shows the effect of different batch sizes. A small batch size (8 or 16) leads to high variance in the network utility. The larger the batch size, the more memory space the algorithm consumes. Also, a large batch size may reduce the randomness of gradient descent and lead to the local minimum value. Hence, the batch size is set to 64.

Fig. 7(c) shows the effect of different embedding dimensions d_e . A small embedding dimension (16) cannot adequately characterize features and thus degrades the performance and convergence speed. A large embedding dimension (256) may overfit the training set, resulting in unstable performance. Hence, the embedding dimension is set as $d_e = 128$.

Overall, the simulation results in Fig. 7 show that the proposed ASOPA can converge under the set parameters.

6.2 Network Utility Performance

To evaluate the SIC ordering generated by ASOPA, we compare it with five baseline algorithms:

- 1) Exhaustive search [24]: This scheme calculates the network utility for all $N!$ SIC orderings and obtains the optimal network utility.
- 2) Tabu search [34]: This scheme initiates a SIC ordering and swaps any two users' ordering to search. For each search iteration, it tries all possible swapping of two users for a SIC ordering and selects the best one for the next search iteration. To the best of our knowledge, the Tabu search algorithm presented in [34] is the state-of-the-art algorithm for dynamic SIC ordering, albeit at the cost of high computational complexity.

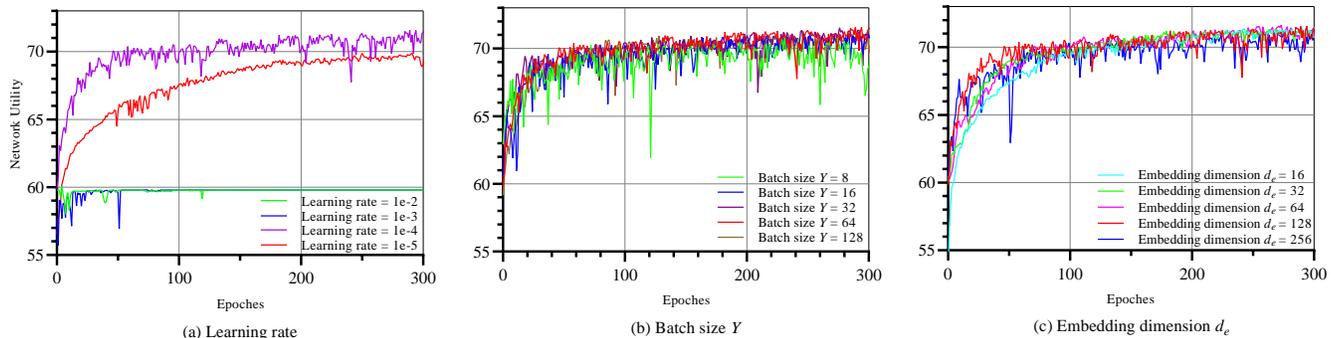


Fig. 7: Convergence performance of ASOPA under different algorithm parameters when $N = 10$.

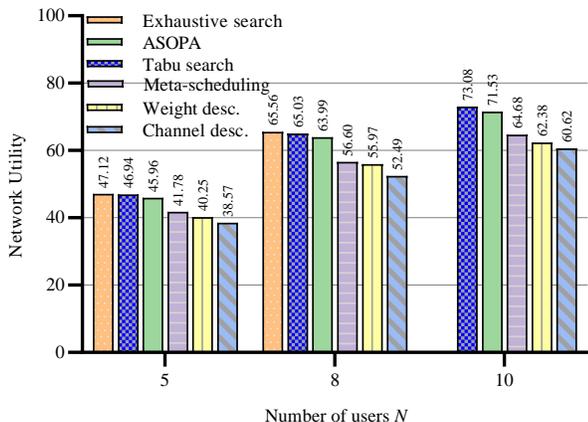
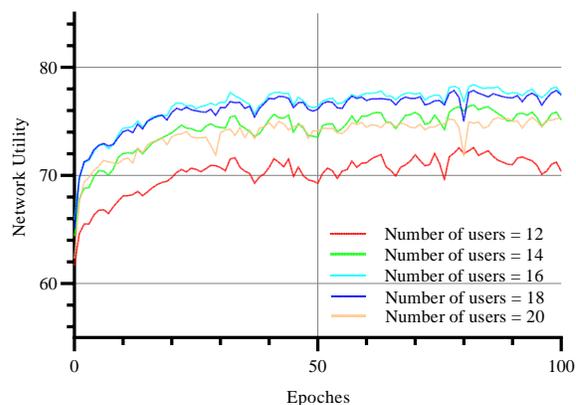


Fig. 8: The network utility under different numbers of users.

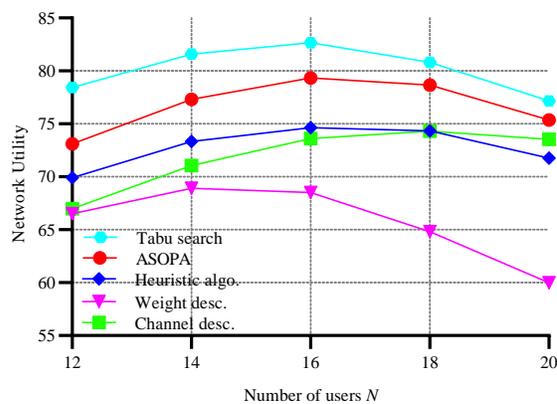
- 3) Meta-scheduling [21]: This scheme sequentially adds and inserts each user into an order. It tries every possible insertion position for each insertion and greedily chooses the one with the greatest utility gain.
- 4) Weight descending [23]: The static SIC ordering follows the descending order of users' weights.
- 5) Channel descending [25]: The static SIC ordering follows the descending order of users' channel gains.

After those baseline algorithms determine the SIC ordering, the optimal transmit power is determined by the power allocation method proposed in Section 4.3.

Fig. 8 presents the network utility achieved by different algorithms for varying numbers of users N . Exhaustive search achieves optimal performance with $N = 5$ and $N = 8$ but not $N = 10$ due to the unacceptable running time for enumerating $10!$ possible SIC ordering. When $N = 5$ and $N = 8$, through sufficient search iteration, Tabu search achieves 99.61% and 99.19% of the optimal performance obtained by exhaustive search. ASOPA achieves 97.54% and 97.60% of the optimal performance, which is close to the performance of Tabu search. When $N = 5$, $N = 8$, and $N = 10$, ASOPA is over 10% higher in network utility



(a) Convergence performance of ASOPA.



(b) The network utility of different algorithms.

Fig. 9: The performance of ASOPA in large-scale scenarios when N is between 10 and 20.

than the other three baseline algorithms besides Tabu search, respectively.

Fig. 9 provides further evaluation of ASOPA in large-scale scenarios, specifically where the number of users N varies from 10 to 20. In Fig. 9(a), ASOPA demonstrates a

consistent convergence rate of around 50 epochs, regardless of the specific values of N . Meanwhile, Fig. 9(b) illustrates that ASOPA consistently achieves average 95% performance of Tabu search, and outperforms the other three baseline algorithms in these large-scale scenarios, aligning with the observations from Fig. 8. An interesting observation is that for $N > 16$, the channel descending algorithm surpasses Meta-scheduling in terms of network utility. This comparison further underscores that network utility tends to decline when the number of users in a NOMA system exceeds a certain threshold, particularly when $N > 16$. The decline of network utility can be attributed to the concave logarithm throughput function $\ln R_n$ in network utility in (3). Intuitively, as the number of users increases, the sum rate $\sum_{n=1}^N R_n$ tends to saturate, leading to a decrease in the sum of logarithms $\sum_{n=1}^N \ln R_n$ due to Jensen's inequality. Overall, the results depicted in Fig. 9 confirm ASOPA's effectiveness in handling large-scale scenarios and its superiority over all baseline algorithms.

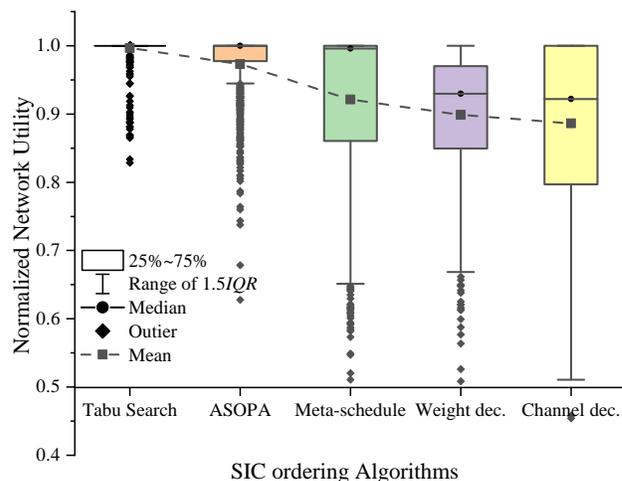
In Fig. 10, we further compare the performance of ASOPA and baseline algorithms over 1000 independent samples when $N = 5$. Fig. 10(a) displays the mean, median, confidence interval, and outliers of the normalized network utility for different algorithms. The normalized network utility is the ratio of the network utility achieved by an algorithm to the optimal network utility obtained by exhaustive search. We observe that the medians of Tabu search and ASOPA are close to 1, and the confidence intervals of Tabu search and ASOPA are over 99% and 97%, respectively. Although some outliers affect the mean of ASOPA, it still outperforms the baseline algorithms besides Tabu search. In Fig. 10(b), we present the hit rate of the top 10 maximum network utilities for ASOPA and baseline algorithms. The hit rate is defined as the percentage of times that an algorithm generates an SIC ordering that appears in the top 10 maximum network utilities obtained by exhaustive search. We observe that ASOPA achieves hit rates of over 55% and 70% for the top 5 and top 10 maximum network utilities, respectively. The results in Fig. 10 further confirm that ASOPA can achieve near-optimal network utility performance.

Fig. 11 shows the network utility under different maximum distances of users to the BS. The network utility achieved by all algorithms decreases slightly as the maximum distance increases. Within the distance range [100, 300] m, ASOPA achieves performance close to that of Tabu search algorithm and outperforms the other three baseline algorithms by an average of 10%.

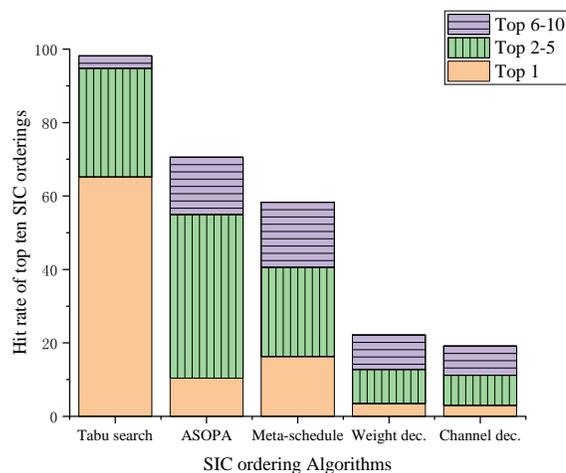
Fig. 12 demonstrates how the network utility varies with different levels of noise power spectral densities. As the noise power spectral density increases, the network utility of all algorithms decreases, and the difference between ASOPA and baseline algorithms diminishes. At a noise power spectral density of -144 dBm/Hz, ASOPA and the channel descending algorithm exhibit the slightest difference of 4.27%. This result indicates that when the users' channel quality is inferior, the users' channel state significantly impacts the SIC ordering.

6.3 Execution Latency

In order to meet the real-time requirement of NOMA networks, the execution time of the SIC ordering and power



(a) Boxplot of the normalized network utility for ASOPA and other baseline algorithms when $N = 5$. The central line indicates the median, while the bottom and top edges of the box indicate the 25th and 75th percentiles, respectively.



(b) Stacked plot of the top ten best SIC ordering hits when $N = 5$.

Fig. 10: The distribution of the network utility achieved by different algorithms.

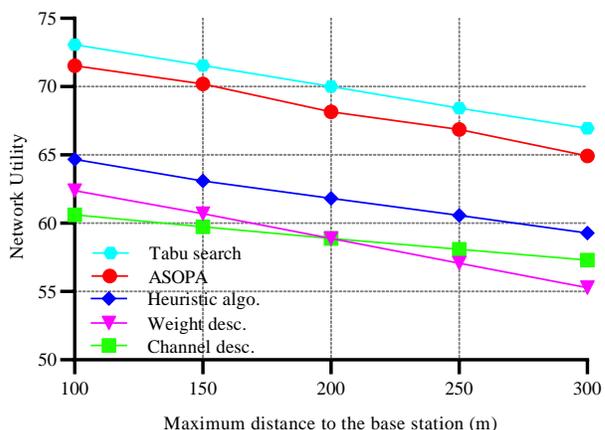


Fig. 11: The network utility under different maximum distances to the BS when $N = 10$.

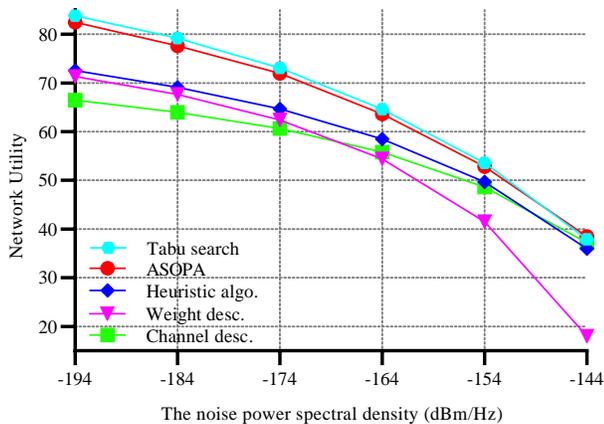


Fig. 12: The network utility under different noise power spectral densities when $N = 10$.

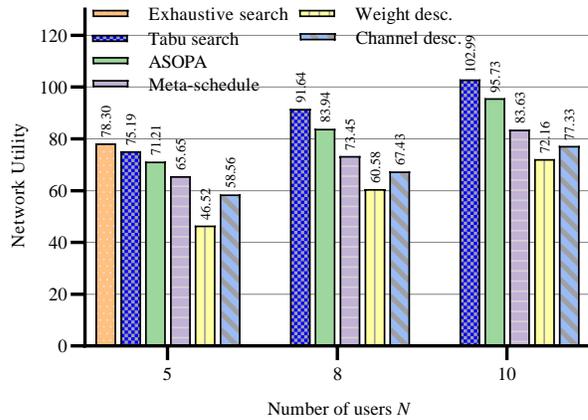


Fig. 14: The network utility under different numbers of users for multiple-antenna NOMA networks.

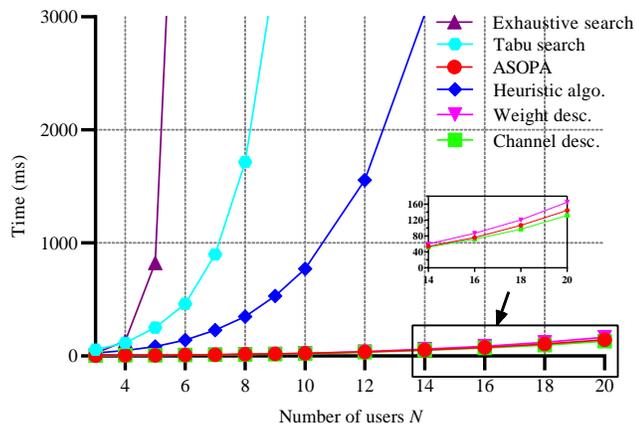


Fig. 13: The average execution time under different numbers of users.

allocation algorithm need be much smaller than the slot duration, i.e., two seconds [43]. To evaluate the efficiency of ASOPA and baseline algorithms, we test the average execution time under different numbers of users, and the results are shown in Fig. 13 and Table. 4.

The execution time of ASOPA is close to that of the weight descending algorithm and the channel descending algorithm, i.e., 24 ms, 25 ms, and 23 ms for a ten-user NOMA network. The execution delay is scalable with the network size N and is acceptable for field deployment. ASOPA

TABLE 4: The average execution latency of different algorithms (ms)

Algorithms	N=5	N=8	N=10	N=14	N=20
Exhaustive search	823	741 933	/	/	/
Tabu search	252	1 717	5 643	45 122	492 677
Meta-scheduling	84	349	771	3 025	14 737
Weight des.	7	15	25	60	165
Channel des.	7	14	23	52	132
ASOPA	8	16	24	53	152

only takes 152 ms even for a twenty-user NOMA network. However, the execution latency of Meta-scheduling and Tabu search significantly increases with the network size N , consuming 771 ms and 5643 ms for a ten-user NOMA network, respectively. The execution time of exhaustive search is exponentially increasing with N . It takes 823 ms and 6630 ms even for NOMA networks with five-user and six-user, respectively. According to Fig. 13 and Table. 4, Tabu search fails to cope with real-time execution when $N > 7$, while Meta-scheduling fails at $N > 10$. In contrast, ASOPA maintains the same latency as the static algorithm for all the number of users. In particular, is three orders of magnitude lower than Tabu search and two orders of magnitude lower than Meta-scheduling when $N = 20$.

ASOPA uses the actor network to generate the SIC ordering, whose time consumption is negligible. The primary time overhead of various algorithms comes from solving the power allocation problem by the interior-point method. Exhaustive search solves the power allocation problem $N!$ times. Meta-scheduling solves the power allocation $N(N+1)/2$ times. Tabu search solves the power allocation $IN(N+1)/2$ times, where I denotes the number of search iterations. ASOPA, the weight descending algorithm, and the channel descending algorithm solve the power allocation problem once. Therefore, the proposed ASOPA executes efficiently like the static SIC ordering algorithms, while performing as well as the exhaustive search algorithm.

Regarding the training latency, ASOPA's policy update is conducted infrequently and in parallel with the inference process, as detailed in Algorithm 1. Extensive evaluations have shown that the duration of a single policy update is approximately one second when the number of users N is 10 and training batch size is 64. On average, the duration of the policy update process is less than 20 ms for each sample. Therefore, the policy update process of ASOPA can feasibly be executed online for NOMA networks, ensuring that the system remains up-to-date and responsive to changing network conditions.

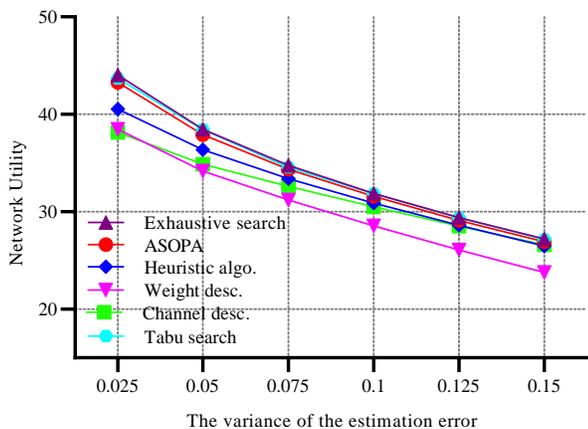


Fig. 15: The network utility under different estimated errors when $N = 5$.

6.4 Extension Scenario

Fig. 14 presents the network utility achieved by different algorithms for varying numbers of users N in multiple-antenna scenario. Specifically, we consider a NOMA network with two antennas at the BS. All algorithms use the minimum-mean-square-error (MMSE) [62] linear equalization to detect symbols. For $N = 5$, the optimal network utility was calculated using exhaustive search. Remarkably, ASOPA achieves 90.94% of the optimal network utility, with only a 5% performance degradation compared to the Tabu search algorithm. Furthermore, ASOPA consistently outperforms the other three baseline algorithms across all settings, which agrees the observation in Fig. 8. Due to the complexity of user state in multiple antenna scenarios, the performance of ASOPA can be further optimized in future work.

Fig. 15 shows how network utility varies with different variances of estimated error under imperfect CSI conditions with five users. ASOPA consistently achieves over 98% of the optimal performance obtained through exhaustive search and achieves 99% performance of Tabu search. Specifically, when $\sigma_{\epsilon_n}^2 = 0.025$, ASOPA's network utility is 6.75%, 12.49%, 13.44% higher than that of Meta-scheduling, channel descending and weight descending, respectively. These results demonstrate ASOPA's robustness and effectiveness in scenarios with imperfect CSI, highlighting its ability to adapt to varying degrees of channel estimation errors.

Fig. 16 presents the network utility achieved by various algorithms for different pairs of users N_1 - N_2 in dual-BS NOMA networks. From the figure, it's evident that ASOPA performs comparably to the exhaustive search method and surpasses other benchmark algorithms. Notably, Tabu search in [34] and Meta-scheduling proposed in [21] is not applicable in this scenario and thus is not included in the comparison. Particularly, when each BS has three or four users, ASOPA achieves 99.45% and 99.80% of the optimal performance determined by exhaustive search, respectively. The network utility reaches its peak when each BS is serving six users. This optimal network utilization can be attributed

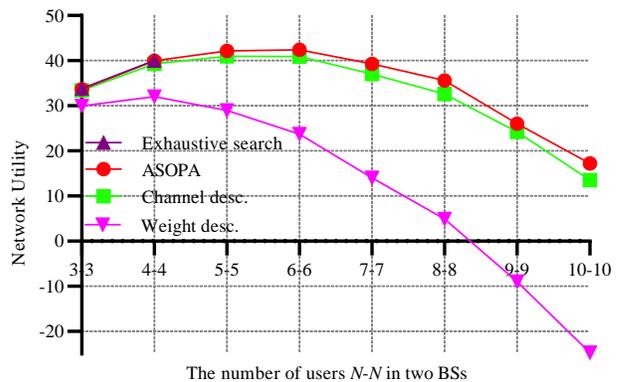


Fig. 16: The network utility under different numbers of users in a dual-BS NOMA network.

to the increasing inter-cell interference with the number of users and the logarithmic throughput function in network utility. As the number of users increases, although the sum throughput saturates, some users' weighted logarithmic throughput even becomes a small negative value (say -10), which leads to a decline in network utility. As the number of users per BS rises to ten, ASOPA's performance advantage becomes more pronounced, showing a 27.69% higher network utility compared to the channel descending algorithm. These results highlight ASOPA's effectiveness in adapting to varying user densities in multi-BS NOMA networks.

7 CONCLUSION

This paper focuses on optimizing the sum-weighted logarithmic throughput in uplink NOMA-based wireless networks by jointly optimizing the SIC ordering and users' transmit powers. To tackle this problem, we propose the ASOPA framework, which innovatively combines DRL with optimization theory. Key to ASOPA's success is an attention-based actor network, trained via reinforcement learning, which effectively derives a near-optimal SIC ordering. Subsequently, this is complemented by the application of optimization techniques to allocate the optimal transmit power for users. Simulation results show that ASOPA can achieve near-optimal performance in a low execution latency. A particularly noteworthy aspect of ASOPA is its extensibility; the framework is adept at solving a range of optimization challenges, particularly those that involve dynamic SIC orderings within the NOMA context.

Looking ahead, our aim is to evolve ASOPA for more complex scenarios, including developing a distributed framework for NOMA networks with multiple base stations, tackling the challenges of imperfect SIC decoding and integrating the QoS constraints in our framework.

APPENDIX A

DETAILS OF THE MULTI-HEAD ATTENTION MECHANISM

The input to the encoder is denoted as \mathbf{X} . It is first transformed into a d_e -dimensional space by a fully connected feed-forward (FF_1) layer. The output of the FF_1 layer is then

fed into a multi-head attention (MHA) layer and a feed-forward (FF₂) layer in sequence. Therefore, the encoding process can be expressed as

$$\begin{aligned}\hat{\mathbf{e}}_n &= \text{BN} \left(\text{FF}_1(\mathbf{x}_n) + \text{MHA} \left(\text{FF}_1(\mathbf{X}_1), \dots, \text{FF}_1(\mathbf{X}_N) \right) \right) \\ \mathbf{e}_n &= \text{BN} \left(\hat{\mathbf{e}}_n + \text{FF}_2(\hat{\mathbf{e}}_n) \right).\end{aligned}\quad (32)$$

For the MHA layer and the FF₂ layer, they have the residual connection (RC) and are followed by batch normalization (BN). The details of each layer are shown as follows.

A.1 Attention Mechanism

We utilize the attention mechanism proposed in [55]. The attention mechanism computes a weighted sum of values, where the weight is determined by a compatibility function based on a query and a set of keys. The query, keys, and values are all embeddings. Specifically, we compute the query \mathbf{q}_n , the key \mathbf{k}_n , and the value \mathbf{v}_n for each user n by multiplying their respective embedding \mathbf{e}_n with parameter matrices \mathbf{W}^Q , \mathbf{W}^K , and \mathbf{W}^V . These parameter matrices have sizes $d_k \times d_e$, $d_k \times d_e$, and $d_v \times d_e$, respectively, as

$$\mathbf{q}_n = \mathbf{W}^Q \mathbf{e}_n, \mathbf{k}_n = \mathbf{W}^K \mathbf{e}_n, \mathbf{v}_n = \mathbf{W}^V \mathbf{e}_n. \quad (33)$$

Then we compute the compatibility $u_{n,j}$ of user n 's query \mathbf{q}_n with user j 's key \mathbf{k}_j :

$$u_{n,j} = \frac{\mathbf{q}_n^T \mathbf{k}_j}{\sqrt{d_k}}. \quad (34)$$

From $u_{n,j}$, we can compute the attention weight $a_{n,j}$ by a softmax function:

$$a_{n,j} = \frac{\exp(u_{n,j})}{\sum_{j=1}^N \exp(u_{n,j})}. \quad (35)$$

Finally, we compute the sum of weighted keys to get the final message \mathbf{e}'_n :

$$\mathbf{e}'_n = \sum_{j=1}^N a_{n,j} \mathbf{v}_j. \quad (36)$$

A.2 Multi-head Attention

Multi-head attention uses M groups of different parameters $\mathbf{W}_{m'}^Q$, \mathbf{W}_m^K and \mathbf{W}_m^V . We set $M = 8$ and $d_k = d_v = \frac{d_e}{M} = 16$, to get the messages, which are denoted as $\mathbf{e}'_{n,m}$, $\forall m \in \{1, \dots, M\}$, and use $d_e \times d_v$ matrices \mathbf{W}_m^A to change their size and then sum them up as the final message:

$$\text{MHA}_n(\mathbf{e}_1, \dots, \mathbf{e}_N) = \sum_{m=1}^M \mathbf{W}_m^A \mathbf{e}'_{n,m}. \quad (37)$$

A.3 Feed Forward Layer

There are two feed-forward layers in the encoder. The first FF₁ is just a fully connected layer with learnable parameters \mathbf{W}_1 and \mathbf{b}_1 :

$$\text{FF}_1(\mathbf{x}_n) = \mathbf{W}_1 \mathbf{x}_n + \mathbf{b}_1. \quad (38)$$

And the second feed-forward layer FF₂ consists of two fully connected layer and use a Relu activation after the first connected layer:

$$\text{FF}_2(\hat{\mathbf{e}}_n) = \mathbf{W}_{2,2} \text{Relu}(\mathbf{W}_{2,1} \hat{\mathbf{e}}_n + \mathbf{b}_{2,1}) + \mathbf{b}_{2,2}, \quad (39)$$

where $\hat{\mathbf{e}}_n$ is the input for FF₂, $\mathbf{W}_{2,1}$ and $\mathbf{b}_{2,1}$ are the parameter matrix and bias of the first fully connected layer, respectively, and $\mathbf{W}_{2,2}$ and $\mathbf{b}_{2,2}$ are the one of the second layer, respectively.

A.4 Batch Normalization

We use batch normalization shown in [53]:

$$\text{BN}(\mathbf{e}_n) = \mathbf{w}^{\text{bn}} \odot \overline{\text{BN}}(\mathbf{e}_n) + \mathbf{b}^{\text{bn}}, \quad (40)$$

where \mathbf{w}^{bn} and \mathbf{b}^{bn} are learnable d_e -dimensional affine parameters, \odot denotes the element-wise product, and $\overline{\text{BN}}$ refers to batch normalization without affine transformation.

APPENDIX B

CONVEX TRANSFORMATION AND PROOF OF P1

For user's transmit power $p_n > 0, \forall n \in \mathcal{N}$, let $p_n = e^{y_n}, \forall n \in \mathcal{N}$. We introduce an auxiliary variable ν_n for user n and add a constraint to guarantee that the weighted logarithmic throughput of user n is not less than ν_n . Then the power allocation sub-problem **P1** can be transformed into the following convex problem

$$\mathbf{P2} : \max_{\boldsymbol{\nu}, \mathbf{y}} \sum_{n=1}^N \nu_n \quad (41a)$$

$$s.t. e^{y_n} \leq P_n^{\text{max}}, \quad (41b)$$

$$w_n \ln \log_2 \left(1 + \frac{e^{y_n} g_n}{\sum_{\xi(n') > \xi(n), \forall n' \in \mathcal{N}} e^{y_{n'}} g_{n'} + N_0} \right) \geq \nu_n, \quad (41c)$$

$$\forall n \in \mathcal{N},$$

where $\boldsymbol{\nu} = [\nu_1, \nu_2, \dots, \nu_N]$ and $\mathbf{y} = [y_1, y_2, \dots, y_N]$. It's easy to know that (41a) and (41b) are convex. Next, we will show that (41c) is convex. First we convert (41c) as

$$\frac{e^{y_n} g_n}{\sum_{\xi(n') > \xi(n), \forall n' \in \mathcal{N}} e^{y_{n'}} g_{n'} + N_0} \geq 2^{e^{\frac{\nu_n}{w_n}}} - 1. \quad (42)$$

Then we take the reciprocal of both sides and take the natural logarithm of both sides, so we can get

$$\ln \left(\frac{\sum_{\xi(n') > \xi(n), \forall n' \in \mathcal{N}} e^{y_{n'}} g_{n'} + N_0}{e^{y_n} g_n} \right) + \ln \left(2^{e^{\frac{\nu_n}{w_n}}} - 1 \right) \leq 0. \quad (43)$$

The first term in the left-hand-side (LHS) is a log-sum-exp function which is convex [52]. The second-order derivative of the second term in the LHS is

$$\frac{\ln 2}{w_n^2} e^{\frac{\nu_n}{w_n}} 2^{e^{\frac{\nu_n}{w_n}}} \left(2^{e^{\frac{\nu_n}{w_n}}} - \ln 2 e^{\frac{\nu_n}{w_n}} - 1 \right), \quad (44)$$

whose value is non-negative. Since the first order derivative of the term inside brackets in (44) is $\frac{\ln 2}{w_n} e^{\frac{\nu_n}{w_n}} \left(2^{e^{\frac{\nu_n}{w_n}}} - 1 \right)$, which is positive due to $\nu_n > 0, w_n > 0$. Thus, the minimum of (44) is $\ln \frac{e}{2}$ larger than zero, and the second term in the LHS is also convex. Therefore, (41c) is convex. For (41a)~(41c) are convex, the problem **P2** is convex. The proof is completed.

APPENDIX C

PROBABILISTIC PROBLEM TRANSFORMATION

In the imperfect CSI scenario, the outage probability requirement turns the problem into an intractable non-convex probability mixed problem. Following [69], [70], we transform this problem into a non-probability problem by approximations.

Firstly, we transform the outage probabilistic requirement into another form of probabilistic constraint as follows. The maximum achievable data rate is rewritten as

$$\begin{aligned} c_n &= W \log_2(1 + \phi_n) \\ &= W \log_2 \left(1 + \frac{c_n^S}{c_n^I} \right), \end{aligned} \quad (45)$$

where $c_n^S = p_n |\alpha_n|^2 \bar{g}_n$ and $c_n^I = \sum_{\xi(n') > \xi(n), \forall n' \in \mathcal{N}} p_{n'} |\alpha_{n'}|^2 \bar{g}_{n'} + N_0$.

The scheduled data rate can be rewritten as

$$\begin{aligned} r_n &= W \log_2(1 + \hat{\phi}_n) \\ &= W \log_2 \left(1 + \frac{b_n^S}{b_n^I} \right), \end{aligned} \quad (46)$$

and we have

$$\hat{\phi}_n = \frac{b_n^S}{b_n^I} = 2^{\frac{r_n}{W}} - 1. \quad (47)$$

According to the above transformation and the total probability theorem, the outage probability constraints can be transformed as

$$\begin{aligned} \Pr[c_n < r_n | \hat{\alpha}_n] &= \Pr[\phi_n < \hat{\phi}_n | \hat{\alpha}_n] \\ &= \Pr \left[\frac{c_n^S}{c_n^I} < 2^{\frac{r_n}{W}} - 1 | \hat{\alpha}_n \right] \\ &= \Pr[E1] \cdot \Pr[c_n^S \leq b_n^S | \hat{\alpha}_n] \\ &\quad + \Pr[E2] \cdot \Pr[c_n^S > b_n^S | \hat{\alpha}_n] \leq \epsilon_{out}, \end{aligned} \quad (48)$$

where $\Pr[E1] = \Pr \left[\frac{c_n^S}{c_n^I} < 2^{\frac{r_n}{W}} - 1 | c_n^S \leq b_n^S, \hat{\alpha}_n \right]$ and $\Pr[E2] = \Pr \left[\frac{c_n^S}{c_n^I} < 2^{\frac{r_n}{W}} - 1 | c_n^S > b_n^S, \hat{\alpha}_n \right]$. Then, we have the following theorem.

Theorem 1. Following [69], the outage probability constraint (48) can be approximated as

$$\Pr[c_n^I \geq b_n^I | \hat{\alpha}_n] \leq \epsilon_{out}/2, \quad (49)$$

and

$$\Pr[c_n^S \leq b_n^S | \hat{\alpha}_n] = \epsilon_{out}/2. \quad (50)$$

Proof 1. According to (23), we have

$$\begin{aligned} \Pr[c_n^I \geq b_n^I | \hat{\alpha}_n] &= \Pr \left[c_n^I \geq b_n^S / (2^{\frac{r_n}{W}} - 1) | \hat{\alpha}_n \right] \\ &= \Pr \left[\frac{b_n^S}{c_n^I} \leq 2^{\frac{r_n}{W}} - 1 | \hat{\alpha}_n \right] \leq \epsilon_{out}/2, \end{aligned} \quad (51)$$

and when $c_n^S > b_n^S$, we can always have

$$\Pr[E2] = \Pr \left[\frac{c_n^S}{c_n^I} < 2^{\frac{r_n}{W}} - 1 | \hat{\alpha}_n \right] \leq \epsilon_{out}/2. \quad (52)$$

According to (24), we have

$$\Pr[c_n^S > b_n^S | \hat{\alpha}_n] = 1 - \epsilon_{out}/2. \quad (53)$$

Based on (52) and (53), the probabilistic constraint (48) satisfies the following approximations

$$\begin{aligned} \Pr[c_n < r_n | \hat{\alpha}_n] &= \Pr[E1] \cdot \Pr[c_n^S \leq b_n^S | \hat{\alpha}_n] + \Pr[E2] \cdot \Pr[c_n^S > b_n^S | \hat{\alpha}_n] \\ &\leq \epsilon_{out}/2 + (\epsilon_{out}/2)(1 - \epsilon_{out}/2) = \epsilon_{out} - \epsilon_{out}^2/4. \end{aligned} \quad (54)$$

For $\epsilon_{out} \ll 1$, we have $\epsilon_{out} - \epsilon_{out}^2/4 \approx \epsilon_{out}$. Therefore, the probabilistic constraint (48) can be approximated as (49) and (50). This completes the proof.

Secondly, based on the transformed probabilistic constraints (49) and (50) of Theorem 1, the probabilistic mixed problem can be further transformed to a non-probabilistic problem as follows.

According to the Markov inequality, the LHS of (49) can derive as follows [70]

$$\begin{aligned} \Pr[c_n^I \geq b_n^I | \hat{\alpha}_n] &= \Pr \left[\sum_{\xi(n') > \xi(n), \forall n' \in \mathcal{N}} p_{n'} |\alpha_{n'}|^2 \bar{g}_{n'} + N_0 \geq b_n^I | \hat{\alpha}_n \right] \\ &\leq \frac{E \left[\sum_{\xi(n') > \xi(n), \forall n' \in \mathcal{N}} p_{n'} |\alpha_{n'}|^2 \bar{g}_{n'} \right]}{b_n^I - N_0} \\ &= \frac{\sum_{\xi(n') > \xi(n), \forall n' \in \mathcal{N}} p_{n'} |\alpha_{n'}|^2 \bar{g}_{n'}}{b_n^I - N_0} = \epsilon_{out}/2, \end{aligned} \quad (55)$$

where the right side of the Markov inequality is set to $\epsilon_{out}/2$ according to (49).

Since $|\alpha_n^2|$ is a non-central chi-squared distributed random variable with two degrees of freedom, the LHS of (50) can be rewritten as

$$\begin{aligned} \Pr[c_n^S \leq b_n^S | \hat{\alpha}_n] &= \Pr \left[p_n |\alpha_n|^2 \bar{g}_n \leq b_n^S | \hat{\alpha}_n \right] \\ &= \Pr \left[|\alpha_n|^2 \leq \frac{b_n^S}{p_n \bar{g}_n} | \hat{\alpha}_n \right] \\ &= F_{|\alpha_n|^2} \left(\frac{b_n^S}{p_n \bar{g}_n} \right) \\ &= 1 - Q_1 \left(\sqrt{\frac{2 |\hat{\alpha}_n|^2}{\sigma_\epsilon^2}}, \sqrt{\frac{2 b_n^S}{\sigma_\epsilon p_n \bar{g}_n}} \right) \end{aligned} \quad (56)$$

where $F(\cdot)$ denotes a cumulative distribution function (cdf) of a non-central chi-square random variable with non-centrality parameter $2 |\hat{\alpha}_n|^2 / \sigma_\epsilon^2$, and $Q_1(\cdot)$ is the first-order Marcum Q-function. Based on (50), (56) is equal to $\epsilon_{out}/2$, and b_n^S can be expressed as

$$b_n^S = F_{|\alpha_n|^2}^{-1}(\epsilon/2) \cdot p_n \bar{g}_n, \quad (57)$$

where $F^{-1}(\cdot)$ is the inverse non-central chi-square cdf of $F(\cdot)$. Based on (47), (57) and $|\alpha_n|^2 = |\hat{\alpha}_n|^2 + \sigma_\epsilon^2$, (55) can be further transformed into

$$\begin{aligned} & \frac{\sum_{\xi(n') > \xi(n), \forall n' \in \mathcal{N}} p_{n'} |\alpha_n|^2 \bar{g}_{n'}}{b_n^S / (2^{\frac{r_n}{W}} - 1) - N_0} \\ &= \frac{\sum_{\xi(n') > \xi(n), \forall n' \in \mathcal{N}} p_{n'} (|\hat{\alpha}_n|^2 + \sigma_\epsilon^2) \bar{g}_{n'}}{\frac{F^{-1}_{|\alpha_n|^2}(\epsilon_{out}/2) \cdot p_n \bar{g}_n}{2^{\frac{r_n}{W}} - 1} - N_0} = \frac{\epsilon_{out}}{2}. \end{aligned} \quad (58)$$

Therefore, considering the outage probability constraint, the approximated signal-to-interference-plus-noise ratio (SINR) $\tilde{\phi}_n$ for the n -th user can be derived as

$$\tilde{\phi}_n = \frac{\epsilon_{out} F^{-1}_{|\alpha_n|^2}(\epsilon_{out}/2) \cdot p_n \bar{g}_n}{\epsilon_{out} N_0 + 2 \sum_{\xi(n') > \xi(n), \forall n' \in \mathcal{N}} p_{n'} (|\hat{\alpha}_n|^2 + \sigma_\epsilon^2) \bar{g}_{n'}}, \quad (59)$$

and the corresponding data rate can be written as

$$\tilde{r}_n = W \log_2(1 + \tilde{\phi}_n). \quad (60)$$

Finally, the weighted proportional fairness function with outage probability is transformed into the following non-probability optimization problem

$$\begin{aligned} & \max_{\pi, \mathbf{p}} \sum_{n=1}^N w_n \ln(1 - \epsilon_{out}) \tilde{r}_n \\ & \text{s.t. } 0 < p_n \leq P_n^{max}, \forall n \in \mathcal{N}, \\ & \quad \pi \in \Pi. \end{aligned}$$

APPENDIX D

ALTERNATIVE ALGORITHM FOR MULTIPLE-ANTENNA WITH MMSE EQUALIZATION MATRICES

Under MMSE methods, the equalization matrices involving transmit power variable \mathbf{p} turn the power allocation subproblem into an intractable non-convex problem. To tackle this non-convex problem, we utilize the alternative algorithm to further transform it into the following subproblem: the calculation of \mathbf{V} under given \mathbf{p} and the optimization of \mathbf{p} under given \mathbf{V} . The specific process of the alternative algorithm is as follows.

Firstly, we initiate the transmit power \mathbf{p} .

Secondly, according to the definition, the equalization matrices \mathbf{V} under the MMSE method can be easily calculated by the given \mathbf{p} as

$$\mathbf{V} = \mathbf{P}\mathbf{H}^H(\mathbf{H}\mathbf{P}\mathbf{H}^H + \sigma\mathbf{I})^{-1} \quad (62)$$

Thirdly, obtained \mathbf{V} , the power allocation subproblem can be formulated as

$$\mathbf{P1} : R(\pi) = \max_{\mathbf{p}} \sum_{n=1}^N w_n \ln R_n \quad (63a)$$

$$\text{s.t. } 0 < p_n \leq P_n^{max}, \forall n \in \mathcal{N}. \quad (63b)$$

where R_n is

$$R_n = \log_2 \left(1 + \frac{|\mathbf{v}_n \mathbf{h}_n|^2 p_n}{\sum_{\xi(n') > \xi(n), \forall n' \in \mathcal{N}} |\mathbf{v}_n \mathbf{h}_{n'}|^2 p_{n'} + |\mathbf{v}_n|^2 \sigma^2} \right). \quad (64)$$

\mathbf{v}_n denotes the n -row of the obtained \mathbf{V} . Since \mathbf{v}_n is a constant under a given \mathbf{V} , (63) can be transformed into a convex problem as Appendix. B and then solved by CVX solver.

The alternative algorithm repeat the second and third steps above until the gap between the previous iteration's \mathbf{p} and the current iteration's \mathbf{p} is less than the threshold value.

APPENDIX E

MULTIPLE-BS SCENARIO

We consider a uplink NOMA network consisting of a set of BSs \mathbf{B} and each BS b is associated with N_b users. A BS simultaneously receives signal from its associated users and the other users, and iteratively decodes signal via a SIC ordering. In the SIC process, the remaining undecoded signal and the other users' signal are treated as interference. Therefore, the SINR between user n and BS b can be expressed as

$$\phi_n^{(b)} = \frac{p_n^{(b)} g_n^{(b)}}{\sum_{\xi(n') > \xi(n), \forall n' \in \mathcal{N}_b} p_{n'}^{(b)} g_{n'}^{(b)} + \sum_{b' \in \mathbf{B} \setminus b, \forall m \in \mathcal{N}_b} p_m^{(b')} h_{m,b}^{(b')} + N_0}.$$

Then, we have the data rate between user n associated with BS b ,

$$R_n^{(b)} = \log_2(1 + \phi_n^{(b)}). \quad (65)$$

Therefore, the joint SIC ordering and power allocation optimization problem for multiple-BS NOMA can be expressed as:

$$\max_{\pi, \mathbf{p}} \sum_{b \in \mathbf{B}} \sum_{n=1}^{N_b} w_n^{(b)} \ln R_n^{(b)}$$

$$\text{s.t. } 0 < p_n^{(b)} \leq P_{n,max}^{(b)}, \forall n \in N_b, \forall b \in \mathbf{B} \quad (66a)$$

$$\pi \in \Pi_{\mathbf{B}} \quad (66b)$$

where $w_n^{(b)}$ is the weight of user n associated with BS b and $\pi = [\{\pi_n^{(b)}\}_{n \in \mathcal{N}_b} | b \in \mathbf{B}]$ indicates the SIC order. Here $\pi_i^{(b)} = n$ means that the i -th decoded user in BS b is user n and $\Pi_{\mathbf{B}}$ is the permutation set of all possible SIC orderings.

REFERENCES

- [1] Z. Yang et al., "AI-Driven UAV-NOMA-MEC in Next Generation Wireless Networks," *IEEE Wireless Commun.*, vol. 28, no. 5, pp. 66-73, Oct. 2021.
- [2] Y. Liu, S. Zhang, X. Mu, Z. Ding, R. Schober, N. Al-Dhahir, E. Hossain, and X. Shen. "Evolution of NOMA toward next generation multiple access (NGMA) for 6G," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 4, pp. 1037-1071, Apr. 2022.
- [3] A. Zakeri, A. Khalili, M. R. Javan, N. Mokari, and E. Jorswieck, "Robust energy-efficient resource management, SIC ordering, and beamforming design for MC MISO-NOMA enabled 6G," *IEEE Trans. Signal Process.*, vol. 69, pp. 2481-2498, Mar. 2021.
- [4] Z. Lin, M. Lin, J.B. Wang, T. De Cola and J. Wang, "Joint beamforming and power allocation for satellite-terrestrial integrated networks with non-orthogonal multiple access," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 3, pp.657-670, 2019.
- [5] M. Diamanti, G. Fragkos, E.E. Tsiropoulou and S. Papavassiliou, "Unified user association and contract-theoretic resource orchestration in NOMA heterogeneous wireless networks," *IEEE Open J. Commun. Soc.*, 1, pp.1485-1502, 2020.

- [6] Z. Lin, M. Lin, T. De Cola, J.B. Wang, W.P. Zhu and J. Cheng, "Supporting IoT with rate-splitting multiple access in satellite and aerial-integrated networks," *IEEE Internet Things J.*, vol. 8, no. 14, pp. 11123-11134, 2021.
- [7] A. Akbar, S. Jangsher, and F. A. Bhatti, "NOMA and 5G emerging technologies: A survey on issues and solution techniques," *Comput. Netw.*, vol. 190, pp. 107950, 2021.
- [8] Y. Wu, Y. Song, T. Wang, L. Qian and T. Q. S. Quek, "Non-Orthogonal Multiple Access Assisted Federated Learning via Wireless Power Transfer: A Cost-Efficient Approach," *IEEE Trans. Commun.*, vol. 70, no. 4, pp. 2853-2869, Apr. 2022.
- [9] O. Maraqa, A. S. Rajasekaran, S. Al-Ahmadi, H. Yanikomeroğlu, and S. M. Sait, "A survey of rate-optimal power domain NOMA with enabling technologies of future wireless networks," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 4, pp. 2192-2235, 4th Quart., 2020.
- [10] M. vaezi, R. Schober, Z. Ding, and H. V. Poor, "Non-orthogonal multiple access: Common myths and critical questions," *IEEE Wireless Commun.*, vol. 26, no. 5, pp. 174-180, Oct. 2019.
- [11] S. M. R. Islam, N. Avazov, O. A. Dobre, and K. S. Kwak, "Power-domain non-orthogonal multiple access (NOMA) in 5G systems: Potentials and challenges," *IEEE Commun. Surveys Tuts.*, vol. 12, no. 2, pp. 721-742, 2nd Quart., 2016.
- [12] D. Zhai, R. Zhang, L. Cai, B. Li and Y. Jiang, "Energy-Efficient User Scheduling and Power Allocation for NOMA-Based Wireless Networks With Massive IoT Devices," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1857-1868, Jun. 2018.
- [13] K. Chi, Z. Chen, K. Zheng, Y.-H. Zhu, and J. Liu, "Energy provision minimization in wireless powered communication networks with network throughput demand: TDMA or NOMA?" *IEEE Trans. Commun.*, vol. 67, no. 9, pp. 6401-6414, Sep. 2019.
- [14] A. Zakeri, M. Moltafet, and N. Mokari, "Joint radio resource allocation and SIC ordering in NOMA-based networks using submodularity and matching theory," *IEEE Trans. Veh. Technol.*, vol. 68, no. 10, pp. 9761-9773, Oct. 2019.
- [15] L. Huang et al, "Throughput Guarantees for Multi-Cell Wireless Powered Communication Networks with Non-Orthogonal Multiple Access," *IEEE Trans. Veh. Technol.*, vol. 7, no. 11, pp. 12104-12116, Nov. 2022.
- [16] J. Cui, Z. Ding, and P. Fan, "The application of machine learning in mmWave-NOMA systems," in *Proc. IEEE 87th Veh. Technol. Conf.*, pp. 1-6., 2018.
- [17] L. You and D. Yuan, "A note on decoding order in user grouping and power optimization for multi-cell NOMA with load coupling," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 495-505, Jan. 2021.
- [18] K. Jiang, T. Jing, Y. Huo, F. Zhang, and Z. Li, "SIC-based secrecy performance in uplink NOMA multi-eavesdropper wiretap channels," *IEEE Access*, vol. 6, pp. 19664-19680, Apr. 2018.
- [19] Y. Gao, B. Xia, K. Xiao, Z. Chen, X. Li, and S. Zhang, "Theoretical analysis of the dynamic decode ordering SIC receiver for uplink NOMA systems," *IEEE Commun. Lett.*, vol. 21, no. 10, pp. 2246-2249, Oct. 2017.
- [20] Y. Gao, B. Xia, Y. Liu, Y. Yao, K. Xiao, and G. Lu, "Analysis of the dynamic ordered decoding for uplink NOMA systems with imperfect CSI," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 6647-6651, Jul. 2018.
- [21] L. Qian, A. Feng, Y. Huang, Y. Wu, B. Ji, and Z. Shi, "Optimal SIC ordering and computation resource allocation in MEC-aware NOMA NB-IoT networks," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2806-2816, Apr. 2019.
- [22] L. Zhang, F. Fang, G. Huang, Y. Chen, H. Zhang, Y. Jiang, and W. Ma, "Energy-Efficient Non-Orthogonal Multiple Access for Downlink Communication in Mobile Edge Computing Systems," *IEEE Trans. Mobile Comput.*, vol. 21, no. 12, pp. 4310-4322, Dec. 2022.
- [23] Z. Ding, R. Schober, and H. V. Poor, "Unveiling the importance of SIC in NOMA systems—Part II: New results and future directions," *IEEE Commun. Lett.*, vol. 24, no. 11, pp. 2378-2382, Nov. 2020.
- [24] D. Hu, Q. Zhang, Q. Li, and J. Qin, "Joint position, decoding order, and power allocation optimization in UAV-based NOMA downlink communications," *IEEE Syst. J.*, vol. 14, no. 2, pp. 2949-2960, Jun. 2020.
- [25] R. Duan, J. Wang, C. Jiang, H. Yao, Y. Ren, and Y. Qian, "Resource allocation for multi-UAV aided IoT NOMA uplink transmission systems," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 7025-7037, Aug. 2019.
- [26] W. Wang, N. Zhao, L. Chen, X. Liu, Y. Chen and D. Niyato, "UAV-Assisted Time-Efficient Data Collection via Uplink NOMA," *IEEE Trans. Commun.*, vol. 69, no. 11, pp. 7851-7863, Nov. 2021.
- [27] Y. Pan, C. Pan, Z. Yang, and M. Chen, "Resource allocation for D2D communication underlying a NOMA-based cellular network," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 130-133, Feb. 2018.
- [28] H. Sun, F. Zhou, R. Q. Hu, and L. Hanzo, "Robust beamforming design in a NOMA cognitive radio network relying on SWIPT," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 1, pp. 142-155, Jan. 2019.
- [29] X. Chen, A. Benjebbour, Y. Lan, A. Li, and H. Jiang, "Evaluation of downlink non-orthogonal multiple access (NOMA) combined with SU-MIMO," in *Proc. IEEE 25th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun.*, Washington, DC, USA, 2014, pp. 1887-1891.
- [30] P. Lai, Q. He, G. Cui, F. Chen, J. Grundy, M. Abdelrazek, J. Hosking, and Y. Yang, "Cost-Effective User Allocation in 5G NOMA-Based Mobile Edge Computing Systems," *IEEE Trans. Mobile Comput.*, vol. 21, no. 12, pp. 4263-4278, Dec. 2022.
- [31] G. Cui, Q. He, X. Xia, F. Chen, F. Dong, H. Jin, and Y. Yang, "OL-EUA: Online User Allocation for NOMA-Based Mobile Edge Computing," *IEEE Trans. Mobile Comput.*, vol. 22, no. 4, pp. 2295-2306, Apr. 2023.
- [32] K. Yakou and K. Higuchi, "Downlink NOMA with SIC using unified user grouping for non-orthogonal user multiplexing and decoding order," in *Proc. Int. Symp. Intell. Signal Process. Commun. Syst. (ISPACS)*, Indonesia, Nusa Dua, 2015, pp. 508-513.
- [33] Z. Liu, F. Yang, J. Song and Z. Han, "NOMA-Based MISO Visible Light Communication Systems With Optical Intelligent Reflecting Surface: Joint Active and Passive Beamforming Design," *IEEE Internet Things J.*, vol. 11, no. 10, pp. 18753-18767, 15 May 2024.
- [34] L. P. Qian, X. Dong, M. Wu, Y. Wu and L. Zhao. "Long-Term Energy Consumption Minimization in NOMA-Enabled Vehicular Edge Computing Networks." *IEEE Trans. Intell. Transp. Syst.*, doi: 10.1109/TITS.2024.3404991.
- [35] L. Huang, X. Feng, C. Zhang, L. P. Qian, and Y. Wu, "Deep Reinforcement Learning-based Joint Task Offloading and Bandwidth Allocation for Multi-User Mobile Edge Computing", *Digital Commun. Netw.*, vol. 5, no. 1, pp. 10-17, Feb. 2019.
- [36] Y.-C. Wu, T. Q. Dinh, Y. Fu, C. Lin, and T. Q. S. Quek, "A hybrid DQN and optimization approach for strategy and resource allocation in MEC Networks," *IEEE Trans. Wirel. Commun.*, vol. 20, no. 7, pp. 4282-4295, Jul. 2021.
- [37] X. Zhou, L. Huang, T. Ye and W. Sun, "Computation Bits Maximization in UAV-Assisted MEC Networks With Fairness Constraint," *IEEE Internet Things J.*, vol. 9, no. 21, pp. 20997-21009, Nov. 1, 2022.
- [38] S. Tuli, S. Ilager, K. Ramamohanarao, and R. Buyya, "Dynamic scheduling for stochastic edge-cloud computing environments using A3C learning and residual recurrent neural networks," *IEEE Trans. Mobile Comput.*, vol. 21, no. 3, pp. 940-954, Aug. 2020.
- [39] T. Zhao, F. Li and L. He, "DRL-based joint resource allocation and device orchestration for hierarchical federated learning in NOMA-enabled industrial IoT," *IEEE Trans. Ind. Informat.*, early access, Apr. 27, 2022, doi: 10.1109/TII.2022.3170900.
- [40] Y. Zou, Y. Liu, X. Mu, X. Zhang, Y. Liu and C. Yuen, "Machine Learning in RIS-assisted NOMA IoT Networks," *IEEE Internet Things J.*, early access, Feb. 16, 2022, doi: 10.1109/JIOT.2023.3245288.
- [41] M. Samir, M. Elhattab, C. Assi, S. Sharafeddine, and A. Ghayeb, "Optimizing age of information through aerial reconfigurable intelligent surfaces: A deep reinforcement learning approach," *IEEE Trans. Veh. Technol.*, vol. 70, no. 4, pp. 3978-3983, Apr. 2021.
- [42] X. Li, Q. Wang, J. Liu, and W. Zhang, "Trajectory design and generalization for UAV enabled networks: A deep reinforcement learning approach," *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Seoul, South Korea, pp. 1-6, 2020.
- [43] L. Huang, S. Bi, and Y.-J. A. Zhang, "Deep reinforcement learning for online computation offloading in wireless powered mobile edge computing networks," *IEEE Trans. Mobile Comput.*, vol. 19, no. 11, pp. 2581-2593, Jul. 2020.
- [44] Q. Zhang, "DROO: Integrated Learning and Optimization for Edge Computing Offloading," *Computer.*, vol. 54, no. 12, pp.4-6, Dec. 2021.
- [45] S. Bi, L. Huang, H. Wang and Y.-J. A. Zhang, "Lyapunov-Guided Deep Reinforcement Learning for Stable Online Computation Offloading in Mobile-Edge Computing Networks," in *IEEE Transac-*

- tions on Wireless Communications, vol. 20, no. 11, pp. 7519-7537, Nov. 2021.
- [46] X. Li, L. Huang, H. Wang, S. Bi, and Y. J. A. Zhang, "An integrated optimization-learning framework for online combinatorial computation offloading in MEC networks," *IEEE Wireless Communications*, vol. 29, no. 1, pp. 170-177, Jan. 2022.
- [47] B. Zhu, K. Chi, J. Liu, K. Yu, and S. Mumtaz, "Efficient offloading for minimizing task computation delay of NOMA-based multiaccess edge computing," *IEEE Trans. Commun.*, vol. 70, no. 5, pp. 3186-3203, 2022.
- [48] F. Kelly, A. Maulloo, and D. Tan, "Rate control for communication networks: Shadow prices, proportional fairness and stability," *J. Oper. Res. Soc.*, vol. 49, no. 3, pp. 237-252, 1998.
- [49] W. Li, S. Wang, Y. Cui, X. Cheng, R. Xin, M. A. Al-Rodhaan and A. Al-Dhelaan, "AP Association for Proportional Fairness in Multirate WLANs," *IEEE/ACM Trans. Netw.*, vol. 22, no. 1, pp. 191-202, Feb. 2014.
- [50] L. Chen, Y. Feng, B. Li, and B. Li, "Promenade: Proportionally Fair Multipath Rate Control in Datacenter Networks with Random Network Coding," *IEEE Trans. Parallel Distrib. Syst.*, vol. 30, no. 11, pp. 2536-2546, Nov. 2019.
- [51] I.-H. Hou and P. Gupta, "Proportionally Fair Distributed Resource Allocation in Multiband Wireless Systems," *IEEE/ACM Trans. Netw.*, vol. 22, no. 6, pp. 1819-1830, Dec. 2014.
- [52] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, *Introduction to algorithms*, MIT Press, 2022.
- [53] W. Kool, H. Van Hoof, and M. Welling, "Attention, learn to solve routing problems!" 2018. [Online]. Available: arXiv: 1803.08475.
- [54] O. Vinyals, M. Fortunato, and N. Jaitly, "Pointer networks," *Proc. 28th Int. Conf. Neural Inf. Process. Syst.*, pp. 2692-2700, 2015.
- [55] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser and I. Polosukhin, "Attention is all you need," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 6000-6010.
- [56] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
- [57] R. J. Williams, "Simple statistical gradient following algorithms for connectionist reinforcement learning," *Mach. Learn.*, vol. 8, pp. 229-256, 1992.
- [58] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014. [Online]. Available: arXiv: 1412.6980.
- [59] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, Cambridge, MA, USA: MIT Press, 2018.
- [60] S. Mehrotra, "On the implementation of a primal-dual interior point method," *SIAM J. Optim.*, vol. 2, no. 4, pp. 575-601, 1992.
- [61] X. Qi, M. Peng, and H. Zhang, "Joint mmWave Beamforming and Resource Allocation in NOMA-MEC Network for Internet of Things," *IEEE Trans. Veh. Technol.*, vol. 72, no. 4, pp. 4969-4980, Apr. 2023.
- [62] J. Zhang, J. Fan, J. Zhang, D. W.K.Ng, Q. Sun and B. Ai, "Performance Analysis and Optimization of NOMA-Based Cell-Free Massive MIMO for IoT," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 9625-9639, June. 2022.
- [63] W., Christoph. "Detection and precoding for multiple input multiple output channels." Ph.D. dissertation., Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Erlangen, Germany, 2004.
- [64] Taesang Yoo and A. Goldsmith, "Capacity and power allocation for fading MIMO channels with channel estimation error", *IEEE Transactions on Information Theory*, vol. 52, no. 5, pp. 2203-2214, May 2006.
- [65] S. Han, S. Ahn, E. Oh and D. Hong, "Effect of Channel-Estimation Error on BER Performance in Cooperative Transmission," *IEEE Trans. Veh. Technol.*, vol. 58, no. 4, pp. 2083-2088, May 2009.
- [66] Z. Yang, Z. Ding, P. Fan and G. K. Karagiannidis, "On the Performance of Non-orthogonal Multiple Access Systems With Partial Channel Information," *IEEE Trans. Commun.*, vol. 64, no. 2, pp. 654-667, Feb. 2016.
- [67] Y. Gao, B. Xia, Y. Liu, Y. Yao, K. Xiao and G. Lu, "Analysis of the Dynamic Ordered Decoding for Uplink NOMA Systems With Imperfect CSI," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 6647-6651, July 2018.
- [68] F. Fang, H. Zhang, J. Cheng, S. Roy, and V.C. Leung. "Joint user scheduling and power allocation optimization for energy-efficient NOMA systems with imperfect CSI," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 12, pp. 2874-2885, 2017.
- [69] Zhang, H., Zhang, J. and Long, K., "Energy efficiency optimization for NOMA UAV network with imperfect CSI," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 12, pp. 2798-2809, 2020.
- [70] D. W. K. Ng, E. S. Lo and R. Schober, "Energy-efficient resource allocation in OFDMA systems with large numbers of base station antennas," *IEEE Trans. Wireless Commun.*, vol. 11, no. 9, pp. 3292-3304, Sep. 2012.