Resource Allocation for Joint Interference Management and Security Enhancement in Cellular-Connected Internet-of-Drones Networks

This paper was downloaded from TechRxiv (<https://www.techrxiv.org>).

LICENSE

CC BY 4.0

SUBMISSION DATE / POSTED DATE

08-02-2022 / 22-02-2022

CITATION

HASSAN, MD ZOHEB; Kaddoum, Georges; Akhrif, Ouassima (2022): Resource Allocation for Joint Interference Management and Security Enhancement in Cellular-Connected Internet-of-Drones Networks. TechRxiv. Preprint. <https://doi.org/10.36227/techrxiv.19141907.v2>

DOI

[10.36227/techrxiv.19141907.v2](https://doi.org/10.36227/techrxiv.19141907.v2)

Resource Allocation for Joint Interference Management and Security Enhancement in Cellular-Connected Internet-of-Drones Networks

Md. Zoheb Hassan, Georges Kaddoum, *Senior Member, IEEE*, and Ouassima Akhrif, *Senior Member, IEEE*

Abstract—Internet-of-drones (IoD) systems require enhanced data transmission security and efficient interference management to accommodate the rapidly growing drone-based rate-intensive applications. This paper develops a novel resource allocation scheme to jointly manage interference and enhance the physical layer security of cellular-connected IoD networks in the presence of a multi-band eavesdropping drone. Our envisioned cellular-connected IoD network has multiple full-duplex cellular base stations (CBSs), where each CBS reserves an orthogonal cellular radio resource block (RRB) for the aerial communication links. To efficiently utilize the cellular RRBs, each CBS is connected to a cluster of data transmitting drones using uplink non-orthogonal multiple access (NOMA) scheme. In addition, all the CBSs simultaneously transmit artificial noise signals to weaken the eavesdropper links. A joint optimization problem, considering the transmit power allocation and clustering of the legitimate drones, and the jamming power allocation of the CBSs, is formulated to maximize the worst-case average sum-secrecy-rate of the network. To obtain an efficient solution, the joint optimization problem is decomposed into drone-clustering and power allocation sub-problems. A multi-agent reinforcement-learning framework is devised to solve the drone-clustering sub-problem. Meanwhile, the transmit and jamming power allocation sub-problem is solved by employing fractional programming, successive convex approximation, and alternating optimization techniques. By iteratively solving these two sub-problems, a convergent and online resource allocation algorithm, namely, security and interference management with reinforcement-learning and NOMA (SIREN), is proposed. The superiority of SIREN over several benchmark schemes is verified via extensive simulations.

I. INTRODUCTION

Cellular-connected Internet-of-drones (IoD) is a novel paradigm where drones are deployed as flying Internet-of-things devices, and the data collected by the drones are transmitted to the ground data centers via cellular radio resource blocks (RRBs) [1]. Thanks to the seamless coverage and increased bandwidth of mobile-networks, using cellular RRBs improves both range and capacity of the aerial data transmission links [2]. However, the performance of cellular-connected IoD networks suffers from both inter-drone interference and drone-to-ground interference experienced at neighboring cellular base stations (CBSs) [3]. Besides, due to the broadcasting nature of wireless transmissions and the presence of dominant line-of-sight (LOS) links, the confidential data transmitted in IoD systems can be readily eavesdropped by ground and aerial

adversaries. This work seeks to address these challenges by proposing a joint interference management and secure data transmission mechanism for cellular-connected IoD networks.

A. Related Works

1) *Interference Management in Drone Networks*: Transmit power allocation schemes that minimize the overall energy consumption and latency in IoD systems were investigated in [4], [5]. However, the studies in [4], [5] did not consider any interference in the system. Due to the presence of strong LOS air-to-ground (A2G) communication links, the drones in cellular-connected IoD networks can generate strong interference at the neighboring CBSs. The mitigation of the interference between aerial and cellular links over shared RRBs have been studied in several works. In [6], using a cognitive radio framework, the drone's transmit power control was exploited to maximize the data rate of the aerial communication links while keeping a low interference level at the cellular users. In [7], the joint optimization of the drones' trajectory, transmit power, and CBS association was investigated to manage interference in cellular-connected drone networks. However, the studies conducted in [6], [7] ignored the inter-drone interference. In [8], the joint transmit power control and sub-channel allocation problems were investigated for a cellular-connected multi-drone network using both licensed and unlicensed channels. However, this study considered orthogonal scheduling of the drones over the licensed channels, and treated the interference over the unlicensed channels as a noise. Essentially, the inter-drone interference management scheme of [8] is also sub-optimal.

2) *Physical Layer Security in Drone Networks*: Conventional cryptographic methods enable secured data transmission using encryption keys. However, due to the limited on-board energy and computation capability, and high-mobility of the drones, the generation and distribution of encryption keys is challenging in aerial communication systems. Leveraging wiretap channel coding, physical layer security (PLS) can safeguard the transmitted data without the need for exchanging complex encryption keys. In particular, PLS requires much less complexity than cryptographic methods, which makes PLS attractive for aerial communications [9]. In addition, a multi-layer security mechanism can be constructed by augmenting PLS with upper layer's existing security protocols. The key performance metric of PLS is the secrecy-rate (SR), which quantifies an effective transmission rate of the confidential

Authors are with the École de technologie supérieure (ETS), University of Quebec, Canada. (Corresponding author's e-mail: md-zoheb.hassan.1@ens.etsmtl.ca).

message without any information leakage at the eavesdropping nodes. A positive SR is guaranteed as long as the legitimate link has a higher channel capacity than the eavesdropping link. Hence, the SR can be enhanced by appropriately designing the resource allocations. The existing literature optimized several factors to enhance the SR of aerial communication systems, including the drones' transmit power, the trajectory followed by the drones, and the sub-channel assignment between the drones and the ground users [10]–[12].

The SR of aerial communications can be further enhanced using cooperative jamming, where artificial noise (AN) is simultaneously transmitted with the confidential messages. The purpose of AN is to artificially increase interference at the eavesdroppers without affecting the legitimate links. In [13], dual-drone based cooperative jamming, where one drone transmits confidential messages and another drone simultaneously transmits AN signals to reduce the eavesdropping capability of the ground adversaries, was developed. In [14], a power-splitting method was developed to split the transmit power of the legitimate drone between the confidential messages and AN signals. The studies in [10], [11], [13], [14] assumed accurate estimation of the eavesdroppers' locations. In [15]–[17], resource allocation schemes were developed to maximize the worst-case SR while considering bounded location error models for the eavesdroppers. In [18], a probabilistic location error model for the eavesdroppers was utilized to maximize the outage-constrained average SR via robust optimization of the drones' trajectory, and the transmit and jamming powers. Nonetheless, most of these studies ignored the interference by scheduling the available drone-to-ground links over orthogonal channels. Therefore, despite improving the PLS of drone networks, these studies fail to efficiently utilize the radio resources.

3) *Interference Management with PLS*: Non-orthogonal multiple access (NOMA) is a powerful technique that can enhance the spectral efficiency of multi-user networks [19]. In NOMA, signals from multiple users are multiplexed over the same RRB using different transmit power levels at the transmitter and decoded at the receiving node using successive interference cancellation (SIC). It is noted that the data transmission process in a cellular-connected IoD network is an uplink multiple access scheme. Hence, NOMA is promising to concurrently schedule multiple drones to transmit data over the same cellular RRB. In addition, NOMA improves the achievable SR, and thereby, enhances robustness against eavesdropping [20]. Recently, NOMA-enabled secured aerial communications has received considerable attention. In [21], NOMA was exploited to transmit the secured and public users' messages from an aerial BS such that the secured message cannot be eavesdropped by public users. In [22], NOMA was studied to efficiently allocate the transmit power of an aerial BS between a pair of security-demanding and rate-demanding ground users without leakage of secured information. In [23], both NOMA and cooperative jamming were integrated to enhance the PLS of an aerial communication system. The studies in [21]–[23] considered single drone-based communications, where the existing literature lacks works on the joint management of interference and PLS for multi-drone

cellular-connected IoD networks considering both inter-drone and drone-to-ground interference into account.

B. Contributions

The aim of this work is to develop a resource allocation scheme facilitating secured and high data rate communications between the drones and CBSs in cellular-connected IoD networks. In particular, we envision an IoD network, where multiple legitimate drones concurrently transmit confidential data to a set of full-duplex (FD) CBSs using cellular RRBs in the presence of a multi-band eavesdropping drone. A judicious resource allocation scheme is proposed in order to enhance the overall SR of the network, and efficiently control both co-channel inter-drone interference and drone-to-CBS interference. The proposed scheme has the following three attributes. First, a multi-agent reinforcement-learning (RL) enabled drone-clustering scheme is devised to associate the legitimate drones with the suitable CBSs. Second, uplink NOMA enabled transmit power allocation is developed to improve the system's resilience against the co-channel inter-drone interference and drone-to-CBS interference. Finally, cooperative jamming from the FD CBSs is exploited to counteract the eavesdropping attacks. To the best of our knowledge, this is the first work that investigates resource optimization to jointly manage interference and enhance the PLS in multi-drone cellular-connected IoD networks. Specific contributions of this work are summarized as follows.

- 1) The worst-case average sum-SR (WC-ASR) of the network is maximized subject to target SR constraints for the legitimate drones and maximum acceptable interference constraints at the CBSs. Towards this objective, an optimization problem is formulated by jointly considering the transmit power allocation and clustering of the legitimate drones, and the jamming power allocation of the CBSs. The presented joint optimization problem is NP-hard, making it inherently difficult to find a global optimal solution. To design a low-complexity resource allocation scheme, a two-level optimization approach is proposed. Specifically, the joint optimization problem is decomposed into two sub-problems, i.e., drone-clustering and power allocation sub-problems. A sub-optimal yet efficient solution is obtained by iteratively solving these two sub-problems.
- 2) The drone-clustering sub-problem is formulated as a mixed-strategy repeated game and solved using a multi-agent RL framework. Meanwhile, the transmit and jamming power allocation sub-problem is solved by employing fractional programming, successive convex approximation (SCA), and alternating optimization techniques. Near-optimality of the derived solutions to both sub-problems is proved.
- 3) A resource allocation algorithm of polynomial computational complexity, namely, security and interference management with reinforcement-learning and NOMA (SIREN), is proposed. SIREN converges to a local-optimal solution of the presented joint optimization problem in an online manner. Through extensive simula-

Fig. 1: Cellular-connected IoD network with two FD CBSs, two drone-clusters, two RRBs, and one eavesdropping drone (the drone-to-CBS interference links are not shown for clarity).

tions, the superiority of SIREN over several interference-aware power allocation, drone-clustering, and RL empowered methods is demonstrated.

The rest of this paper is organized as follows. In Section II, the system model and problem formulation are presented. The sub-problems' solutions are provided in Sections III and IV. Section V presents an overview of the properties of the SIREN algorithm. Finally, simulation results and concluding remarks are provided in Sections VI and VII, respectively.

II. SYSTEM MODEL AND PROBLEM FORMULATION

A. System Model

1) *System Overview*: We consider a cellular-connected IoD network, as shown in Fig. 1, with M legitimate drones, K FD CBSs, and a multi-band eavesdropping drone that passively overhears the data transmitted by the legitimate drones over different RRBs. The CBSs are connected with a centralized network controller, that coordinates the overall resource allocation procedure. The considered geographical region is divided into M different sensing zones, each assigned to a single drone. To facilitate the drones' exploration, each sensing zone is divided into N predefined stop-over points. Inspired by [4], we consider that the drones sequentially visit all the stop-over points of their assigned sensing zones, and hover at each stop-over point for a fixed duration of time. Hence, the given mission duration is divided into N time-slots (TSs) of equal duration. Specifically, in the n -th TS, where $n \in \{1, 2, \dots, N\}$, the drones move from the $(n-1)$ -th stop-over point to the n -th stop-over point of their assigned sensing zones, collect data, and send the collected data to the selected CBSs. To maximize the WC-ASR, in each TS, the drone-to-CBS associations, and the drones' transmit and CBSs' jamming powers are optimally determined. We assume that the position of the eavesdropping drone can also differ from a TS to another.

Let $\mathcal{M} = \{1, 2, \dots, M\}$ be the set of legitimate drones; $\mathcal{K} = \{1, 2, \dots, K\}$ be the set of CBSs; $\mathbf{q}_{m,n} = \{x_{m,n}, y_{m,n}\}$ be the 2D location of the n -th stop-over point of the m -th drone; $\nu_e^{(n)}$ be the 2D location of the eavesdropping drone at the n -th TS; $\bar{\mathbf{x}}_k = \{x_k, y_k\}$ be the 2D coordinate of the k -th CBS; H_b be the height of each CBS; H_d and H_e

be the fixed altitudes of the legitimate and eavesdropping drones, respectively. Both legitimate and eavesdropping drones are equipped with a single omnidirectional antenna, whereas each CBS is equipped with two antennas to support FD transmission. Each CBS employs an orthogonal cellular RRB for collecting data from the associated drones. Each cellular RRB is also reused in a set of predefined CBSs for uplink cellular communications, leading to an enhanced spectrum resource utilization [7].

2) *Interference Management*: A hybrid multiple access scheme is adopted for the drone-to-CBS communications. In particular, the data transmitting drones are partitioned into a maximum of K non-overlapping drone-clusters. Each drone-cluster is associated with only one CBS, and each CBS supports only one drone-cluster. Therefore, different drone-clusters utilize orthogonal RRBs for data transmission and as a result, the inter-cluster interference is eliminated. Meanwhile, to alleviate the intra-cluster interference, uplink NOMA with suitable transmit power allocation is employed in each drone-cluster. Finally, to exploit the diversity of the A2G channel, a given drone can be associated with different CBSs at different stop-over points. Consequently, the formation of drone-clusters can vary between different TSs.

Because the same RRB is reused at multiple CBSs for uplink aerial and cellular communications, the drones associated with a particular CBS can generate severe interference at other CBSs [7]. Let us consider that the RRB reserved for aerial communications in the k -th CBS is reused by a set of CBSs, denoted by $\mathcal{L}_k$, for uplink cellular communications. Essentially, the drones associated with the k -th CBS interfere with all the CBSs of the set $\mathcal{L}_k$. Hence, the transmit power of the drones associated with the k -th CBS is controlled to ensure that the resultant interference at each CBS in the set $\mathcal{L}_k$ is lower than an acceptable interference threshold.

3) *Cooperative Jamming*: To enhance the PLS, cooperative jamming is employed. In particular, leveraging its FD communication capability, each CBS simultaneously receives data from the associated drones and transmits AN signals over the RRB reserved for aerial communications. The transmitted AN signals reduce the channel capacity of the eavesdropping links at the cost of self-interference (SI) at the CBSs. We consider that the CBSs are equipped with a practical SI cancellation scheme, and thus are only affected by residual SI [16].

4) *Assumptions*: **A1**: The eavesdropping drone's position is imperfectly known at the network controller [15]–[17]. Let $\tilde{\nu}_e^{(n)}$ be the estimated 2D position of the eavesdropping drone at the n -th TS, and $\nu_e^{(n)} \in \Theta \triangleq \{\|\tilde{\nu}_e^{(n)} - \nu_e^{(n)}\| \leq \chi\}$, where χ is the maximum estimation error and $\|\cdot\|$ is the Euclidean norm. **A2**: The parameters of the A2G, ground-to-air (G2A), and air-to-air (A2A) channel models are accurately available at the network controller. **A3**: Due to shadowing and path loss, the uplink cellular users create negligible interference at the neighboring CBSs, and hence interference from the cellular users to the aerial links is not considered.

B. Channel Model and Secrecy-rate Expression

1) *Channel Model*: The A2G, G2A, and A2A communication channels are usually dominated by the LOS links [6],

[7]. The A2G channel gain between the m -th legitimate drone and k -th CBS in the n -th TS follows the free space path loss model, given by, $h_{m,k}^{(n)} = \frac{\beta_1}{\|q_{m,n} - \bar{x}_k\|^2 + (H_d - H_b)^2}$, where β_1 denotes the channel gain at the reference distance of 1 meter. Similarly, the G2A channel gain between the k -th CBS and the eavesdropping drone in the n -th TS is expressed as $f_{e,k}^{(n)} = \frac{\beta_1}{\|\nu_e^{(n)} - \bar{x}_k\|^2 + (H_b - H_e)^2}$. The A2A channel gain between the m -th legitimate and the eavesdropping drones in the n -th TS is expressed as $g_{m,e}^{(n)} = \frac{\beta_2}{\|q_{m,n} - \nu_e^{(n)}\|^2 + (H_d - H_e)^2}$, where β_2 denotes the channel gain at the reference distance of 1 meter.

2) *Secrecy-rate Expression*: In what follows, we provide the SR expression of the m -th legitimate drone in the n -th TS. Without loss of generality, we assume that in the n -th TS, the m -th legitimate drone is associated with the k -th CBS. Moreover, the set of drones associated with the k -th CBS in the n -th TS is denoted as $\mathcal{S}_k^{(n)}$. Let $P_{m,k}^{(n)}$ be the transmit power of the m -th drone in the n -th TS. The received signals at the k -th CBS and eavesdropping drone are expressed as

$$y_k^{(n)} = \sum_{m=1}^{|\mathcal{S}_k^{(n)}|} \sqrt{P_{m,k}^{(n)} h_{m,k}^{(n)}} s_m + f_d \sqrt{\epsilon Q_k^{(n)}} z_{AN} + n_a, \quad (1)$$

and

$$y_e^{(n)} = \sum_{m=1}^{|\mathcal{S}_k^{(n)}|} \sqrt{P_{m,k}^{(n)} g_{m,e}^{(n)}} s_m + \sqrt{Q_k^{(n)} f_{e,k}^{(n)}} z_{AN} + n_a, \quad (2)$$

respectively, where $|\cdot|$ denotes the cardinality of a set. Here, f_d is the SI channel gain, $Q_k^{(n)}$ is the jamming power of the k -th CBS in the n -th TS, $\epsilon \in (0, 1)$ is SI cancellation coefficient, s_m is the transmitted symbol from the m -th drone with $|s_m|^2 = 1$, z_{AN} is the unit-variance Gaussian AN signal, and n_a is the additive white Gaussian noise (AWGN) with variance σ^2 .

Without loss of generality, the channel gains of the drones associated with the k -th CBS are sorted as $h_{1,k}^{(n)} \geq h_{2,k}^{(n)} \geq \dots \geq h_{m,k}^{(n)} \geq \dots \geq h_{|\mathcal{S}_k^{(n)}|,k}^{(n)}$. The CBS applies SIC to decode for the received signals following the decreasing order of the drones' channel power gains. However, our presented optimization framework can be applied following other decoding orders as well. Moreover, we consider a practical SIC where each drone's signal is subjected to residual interference from the previously decoded signals. The channel capacity of the data transmission link between the m -th legitimate drone and k -th CBS in the n -th TS is given by

$$R_{m,k}^{(n)} = W \log_2 \left(1 + \frac{P_{m,k}^{(n)} h_{m,k}^{(n)}}{I_{SIC} + I_{Sig} + \epsilon Q_k^{(n)} |f_d|^2 + \sigma^2} \right), \quad (3)$$

where W is the bandwidth of the RRB reserved for aerial communications; $I_{SIC} = \epsilon_c \sum_{l=1}^{m-1} P_{l,k}^{(n)} h_{l,k}^{(n)}$ and $I_{Sig} = \sum_{l=m+1}^{|\mathcal{S}_k^{(n)}|} P_{l,k}^{(n)} h_{l,k}^{(n)}$; $\epsilon_c \in [0, 1]$ is the SIC error coefficient, where $\epsilon_c = 0$ and $\epsilon_c = 1$ denote the perfect SIC and treat-interference-as-noise scenarios, respectively.

In practice, the eavesdropping drone does not know the exact codebook used by the legitimate drones, and thereby

requires high complexity to apply SIC-based decoding. Similar to [21], we consider that the eavesdropping drone attempts to decode the transmitted signals of each drone-cluster by treating inter-drone interference as a noise. The channel capacity of the eavesdropping link between the m -th legitimate and eavesdropping drones in the n -th TS is expressed as

$$\tilde{R}_m^{(n)} = W \log_2 \left(1 + \frac{P_{m,k}^{(n)} g_{m,e}^{(n)}}{\sum_{l=1, l \neq m}^{|\mathcal{S}_k^{(n)}|} P_{l,k}^{(n)} g_{l,e}^{(n)} + Q_k^{(n)} f_{e,k}^{(n)} + \sigma^2} \right). \quad (4)$$

The achievable SR of the m -th legitimate drone in the n -th TS is obtained as

$$C_{m,k}^{(n)} = \left[R_{m,k}^{(n)} - \max_{\nu_e^{(n)} \in \Theta} \tilde{R}_m^{(n)} \right]^+, \quad (5)$$

where $[x]^+ = \max(x, 0)$. Due to the uncertainty about the eavesdropping drone's location, it is challenging to compute (5). To address this issue, we derive a tractable lower bound for (5). In particular, we consider the eavesdropping drone's location that results in the worst-case lower-bound of the achievable SR. From [15, eq. (5)] and [17, eq. (20)], we obtain an upper bound of the eavesdropping link's channel capacity as $\max_{\nu_e^{(n)} \in \Theta} \tilde{R}_m^{(n)} \leq \tilde{R}_m^{(up,n)}$, where

$$\tilde{R}_m^{(up,n)} = W \log_2 \left(1 + \frac{P_{m,k}^{(n)} \tilde{g}_m^{(n)}}{\sum_{l=1, l \neq m}^{|\mathcal{S}_k^{(n)}|} P_{l,k}^{(n)} \hat{g}_l^{(n)} + Q_k^{(n)} \hat{f}_k^{(n)} + \sigma^2} \right), \quad (6)$$

and where $\tilde{g}_m^{(n)} = \max_{\nu_e^{(n)} \in \Theta} g_{m,e}^{(n)} = \frac{\beta_2}{(\|q_{m,n} - \hat{\nu}_e^{(n)}\| - \chi)^2 + (H_d - H_e)^2}$, $\hat{g}_l^{(n)} = \min_{\nu_e^{(n)} \in \Theta} g_{l,e}^{(n)} = \frac{\beta_2}{(\|q_{l,n} - \hat{\nu}_e^{(n)}\| + \chi)^2 + (H_d - H_e)^2}$, and $\hat{f}_k^{(n)} = \min_{\nu_e^{(n)} \in \Theta} f_{e,k}^{(n)} = \frac{\beta_1}{(\|\hat{\nu}_e^{(n)} - \bar{x}_k\| + \chi)^2 + (H_b - H_e)^2}$. Therefore, the worst-case SR of the m -th legitimate drone in the n -th TS is expressed as

$$\tilde{C}_{m,k}^{(n)} = \left[R_{m,k}^{(n)} - \tilde{R}_m^{(up,n)} \right]^+. \quad (7)$$

Since (7) is analytically tractable, the ensuing problem formulation and analysis are developed using (7) instead of (5).

C. Problem Formulation

A drone-cluster is defined as a set of the drones associated with the same CBS in a given TS. We denote $\mathcal{S}^{(n)} = \{\mathcal{S}_1^{(n)}, \mathcal{S}_2^{(n)}, \dots, \mathcal{S}_K^{(n)}\}$ as the set of all drone-clusters in the n -th TS. Let $\alpha_{m,k}^{(n)} \in \{0, 1\}$ be a decision variable where $\alpha_{m,k}^{(n)} = 1$ if the m -th legitimate drone is associated with the k -th CBS in the n -th TS, otherwise $\alpha_{m,k}^{(n)} = 0$. Therefore, we obtain $\mathcal{S}_k^{(n)} = \{m \in \mathcal{M} | \alpha_{m,k}^{(n)} = 1\}$. The optimization problem for maximizing the WC-ASR of the network is formulated as P0 at the top of the next page. In P0, P_{max} denotes the maximum instantaneous transmit power of each legitimate drone; constraint C1 ensures that the average SR of each legitimate drone is greater than or equal to a target SR, R_s ; constraint C2 provides a maximum average transmit

power limit, P_{avg} , for each legitimate drone; constraint C3 provides a maximum jamming power limit, Q_{max} , for each CBS; constraint C4 mandates that the maximum interference caused by the drone-clusters to the neighboring CBS(s) is not more than an acceptable interference level, I_{th} ; constraint C5 ensures that the drone-clusters are non-overlapping.

We can readily prove that P0 is NP-hard, and therefore a global optimal solution to P0 is computationally intractable. To obtain a tractable solution to P0, we propose a two-level iterative optimization approach. To this end, P0 is decomposed into the upper-level and lower-level sub-problems as follows.

Upper-level sub-problem: The upper-level sub-problem optimizes the drone-clusters in each TS. The essence of drone-clustering is to associate each data transmitting drone with a suitable CBS so that the sum-SR of the network is maximized. Inspired by such a fact, the upper-level sub-problem for the n -th TS, $\forall n = 1, 2, \dots, N$, is formulated as

$$\begin{aligned} \text{P1: } & \{\alpha_{m,1}^{(n)*}, \alpha_{m,2}^{(n)*}, \dots, \alpha_{m,K}^{(n)*}\} \\ & = \arg \max_{\{\alpha_{m,k}^{(n)}\} \in \{0,1\}} C_{sys}^{(n)} \triangleq \sum_{m=1}^M \sum_{k=1}^K \alpha_{m,k}^{(n)} \tilde{C}_{m,k}^{(n)}, \text{ s.t. C5.} \end{aligned} \quad (9)$$

Lower-level sub-problem: For a given set of drone-clusters, the lower-level sub-problem jointly optimizes the transmit power of the drones and the jamming power of the CBSs using the following optimization problem.

$$\begin{aligned} \text{P2: } & \max_{\substack{0 \leq \{P_{m,k}^{(n)}\} \leq P_{max}, \\ Q_k^{(n)} \geq 0}} \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N \sum_{m=1}^M \sum_{k=1}^K \alpha_{m,k}^{(n)} \tilde{C}_{m,k}^{(n)} \\ & \text{s.t. C1, C2, C3, C4.} \end{aligned} \quad (10)$$

A converged sub-optimal solution to P0 is obtained by iteratively solving P1 and P2. We first solve P2 in Section III for any given set of drone-clusters. Thereafter, by leveraging the solution of P2, P1 is solved in Section IV. Finally, the overall solution to P0 is presented in Section V.

III. POWER ALLOCATION SUB-PROBLEM'S SOLUTION

A. Proposed Solution Approach

Due to the non-convexity of P2, a certain duality-gap exists between P2 and its dual problem. Similar to [24], for a known $\{\alpha_{m,k}^{(n)}\}$, the upper bound of the duality gap is estimated as $\frac{2M+1}{N} \mathcal{E}$, where $\mathcal{E} = \max_{n \in \{1,2,\dots,N\}} \sum_{m=1}^M \sum_{k=1}^K \alpha_{m,k}^{(n)} \tilde{C}_{m,k}^{(n)}$. Intuitively, as long as $N \gg M$, i.e., the total number of available TSs is sufficiently large, P2 can be equivalently solved in dual-domain without a notable loss of optimality. Inspired by this fact, in what follows, a dual-domain solution to P2 is developed. We first express the (partial) Lagrangian function of P2 as (11) at the top of the next page. In (11), λ , ρ , and μ are the non-negative Lagrangian multipliers for the constraints C1 C2, and C4, respectively; $\mathbf{p} = [P_{m,k}^{(n)}]_{m \in \mathcal{M}, k \in \mathcal{K}, 1 \leq n \leq N}$, $\mathbf{q} = [Q_k^{(n)}]_{k \in \mathcal{K}, 1 \leq n \leq N}$, $\boldsymbol{\lambda} = [\lambda_m]_{m \in \mathcal{M}}$, $\boldsymbol{\rho} = [\rho_m]_{m \in \mathcal{M}}$, and $\boldsymbol{\mu} = [\mu_{n,k}]_{k \in \mathcal{K}, 1 \leq n \leq N}$. The dual problem of P2 is expressed as

$$\text{P3: } \min_{\lambda \geq 0, \rho \geq 0, \mu \geq 0} G(\boldsymbol{\lambda}, \boldsymbol{\rho}, \boldsymbol{\mu}), \quad (12)$$

where $G(\boldsymbol{\lambda}, \boldsymbol{\rho}, \boldsymbol{\mu})$ is the Lagrangian dual function, defined as

$$\begin{aligned} G(\boldsymbol{\lambda}, \boldsymbol{\rho}, \boldsymbol{\mu}) &= \max_{\mathbf{p}, \mathbf{q}} \mathcal{L}(\mathbf{p}, \mathbf{q}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \boldsymbol{\mu}) \\ \text{s.t. } & 0 \leq P_{m,k}^{(n)} \leq P_{max}, 0 \leq Q_k^{(n)} \leq Q_{max}, \forall m, k, n. \end{aligned} \quad (13)$$

To solve the dual problem of P2, an optimal solution to (13) is required. Evidently, (13) is separable per TS and decomposed into N optimization problems. The optimization problem for the n -th TS, where $n \in \{1, 2, \dots, N\}$, is given by

$$\begin{aligned} \text{P4: } & \max_{\mathbf{p}, \mathbf{q}} \sum_{k=1}^K \sum_{m=1}^M (1 + \lambda_m) \alpha_{m,k}^{(n)} \tilde{C}_{m,k}^{(n)} \\ & - \sum_{k=1}^K \sum_{m=1}^M \left(\rho_m + \mu_{n,k} \times \max_{l \in \mathcal{L}_k} h_{m,l}^{(n)} \right) \alpha_{m,k}^{(n)} P_{m,k}^{(n)} \\ & \text{s.t. } 0 \leq P_{m,k}^{(n)} \leq P_{max}, 0 \leq Q_k^{(n)} \leq Q_{max}, \forall m, k. \end{aligned} \quad (14)$$

The optimal solution to P4 only depends on the drone-clusters in the current TS. Therefore, P2 can be optimally solved by repeating the following two steps in each TS: (i) solving P4 optimally for the given Lagrangian multipliers and (ii) updating the Lagrangian multipliers by minimizing the function, $G(\boldsymbol{\lambda}, \boldsymbol{\rho}, \boldsymbol{\mu})$ for the updated solution to P4.

B. Solution to P4 for the given Lagrangian Multipliers

Using [11, Lemma 1], we can justify that the operator $[\cdot]^+$ can be omitted from the objective function of P4 without loss of optimality. Hence, P4 can be equivalently expressed as

$$\begin{aligned} \text{P5: } & \max_{\mathbf{p}, \mathbf{q}} \mathcal{F}_0(\mathbf{p}, \mathbf{q}) \triangleq \sum_{k=1}^K \sum_{m \in \mathcal{S}_k^{(n)}} (1 + \lambda_m) \left(R_{m,k}^{(n)} - \tilde{R}_m^{(up,n)} \right) \\ & - \sum_{k=1}^K \sum_{m \in \mathcal{S}_k^{(n)}} \left(\rho_m + \mu_{n,k} \times \max_{l \in \mathcal{L}_k} h_{m,l}^{(n)} \right) P_{m,k}^{(n)} \\ & \text{s.t. } 0 \leq P_{m,k}^{(n)} \leq P_{max}, 0 \leq Q_k^{(n)} \leq Q_{max}, \forall m, k. \end{aligned} \quad (15)$$

Here, $\{\mathcal{S}_1^{(n)}, \mathcal{S}_2^{(n)}, \dots, \mathcal{S}_K^{(n)}\}$ can be any feasible sets of drone-clusters in the n -th TS. Since the CBSs are associated with non-overlapping drone-clusters, P5 is separable per CBS. The optimization problem for the k -th CBS, $\forall k \in \mathcal{K}$, is written as

$$\begin{aligned} \max_{\mathbf{p}, \mathbf{q}} \mathcal{F}_k(\mathbf{p}, \mathbf{q}) & \triangleq \sum_{m \in \mathcal{S}_k^{(n)}} (1 + \lambda_m) \left(R_{m,k}^{(n)} - \tilde{R}_m^{(up,n)} \right) \\ & - \sum_{m \in \mathcal{S}_k^{(n)}} \left(\rho_m + \mu_{n,k} \times \max_{l \in \mathcal{L}_k} h_{m,l}^{(n)} \right) P_{m,k}^{(n)} \\ \text{s.t. } & 0 \leq P_{m,k}^{(n)} \leq P_{max}, 0 \leq Q_k^{(n)} \leq Q_{max}, \forall m \in \mathcal{S}_k^{(n)}. \end{aligned} \quad (16)$$

Eq. (16) is a non-convex optimization problem, and thus solving (16) is non-trivial. To this end, we develop a low-complexity algorithm to near-optimally solve (16) by alternately updating the transmit and jamming power allocations. The detailed solution is provided as follows.

1) *Jamming Power Allocation:* Let us assume that the transmit power allocation of the legitimate drones is known.

$$\begin{aligned}
\text{P0: } & \max_{0 \leq \{P_{m,k}^{(n)}\} \leq P_{max}, Q_k^{(n)} \geq 0, \{\alpha_{m,k}^{(n)}\} \in \{0,1\}} \frac{1}{N} \sum_{n=1}^N \sum_{m=1}^M \sum_{k=1}^K \alpha_{m,k}^{(n)} \tilde{C}_{m,k}^{(n)} \\
\text{s.t. } & \begin{cases} \text{C1: } \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \alpha_{m,k}^{(n)} \tilde{C}_{m,k}^{(n)} \geq R_s, \forall m \in \mathcal{M} \\ \text{C2: } \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \alpha_{m,k}^{(n)} P_{m,k}^{(n)} \leq P_{avg}, \forall m \in \mathcal{M} \\ \text{C3: } Q_k^{(n)} \leq Q_{max}, \forall k \in \mathcal{K}, n \in \{1, 2, \dots, N\} \\ \text{C4: } \max_{l \in \mathcal{L}_k} \sum_{m=1}^M \alpha_{m,k}^{(n)} P_{m,k}^{(n)} h_{m,l}^{(n)} \leq I_{th}, \forall k \in \mathcal{K}, n \in \{1, 2, \dots, N\} \\ \text{C5: } \mathcal{S}_k^{(n)} = \{m \in \mathcal{M} | \alpha_{m,k}^{(n)} = 1\}, \mathcal{S}_k^{(n)} \cap \mathcal{S}_{k' \neq k}^{(n)} = \emptyset, \cup_{k=1}^K \mathcal{S}_k^{(n)} = \mathcal{M}, \forall k, k' \in \mathcal{K}. \end{cases}
\end{aligned} \tag{8}$$

$$\begin{aligned}
\mathcal{L}(\mathbf{p}, \mathbf{q}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \boldsymbol{\mu}) = & \frac{1}{N} \sum_{n=1}^N \sum_{m=1}^M \sum_{k=1}^K \alpha_{m,k}^{(n)} \tilde{C}_{m,k}^{(n)} + \sum_{m=1}^M \lambda_m \left(\frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \alpha_{m,k}^{(n)} \tilde{C}_{m,k}^{(n)} - R_s \right) \\
& - \sum_{m=1}^M \rho_m \left(\frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K \alpha_{m,k}^{(n)} P_{m,k}^{(n)} - P_{avg} \right) - \sum_{n=1}^N \sum_{k=1}^K \mu_{n,k} \left(\sum_{m=1}^M \alpha_{m,k}^{(n)} P_{m,k}^{(n)} \times \max_{l \in \mathcal{L}_k} h_{m,l}^{(n)} - I_{th} \right).
\end{aligned} \tag{11}$$

We first express the objective function of (16) as a difference-of-concave (DC) functions. To this end, using a change of variable $Q_k^{(n)} = \frac{1}{u_k}$, $\forall k$, we obtain $\mathcal{F}_k(\mathbf{p}, \mathbf{q}) = \mathcal{F}_k^{(1)}(\mathbf{p}, \mathbf{u})$, where $\mathbf{u} = [u_k]_{k \in \mathcal{K}}$, with

$$\begin{aligned}
\mathcal{F}_k^{(1)}(\mathbf{p}, \mathbf{u}) = & \sum_{m \in \mathcal{S}_k} \omega_m \left(\ln \left(1 + \gamma_{m,k,n}^{(1)} \right) - \ln \left(1 + \gamma_{m,k,n}^{(2)} \right) \right) \\
& - \sum_{m \in \mathcal{S}_k^{(n)}} \left(\rho_m + \mu_{n,k} \times \max_{l \in \mathcal{L}_k} h_{m,l}^{(n)} \right) P_{m,k}^{(n)}.
\end{aligned} \tag{17}$$

In (17), $\omega_m = \frac{W(1+\lambda_m)}{\ln 2}$, $\gamma_{m,k,n}^{(1)} = \frac{u_k P_{m,k}^{(n)} h_{m,k}^{(n)}}{(I_{SIC} + I_{Sig} + \sigma^2) u_k + \epsilon |f_d|^2}$, and $\gamma_{m,k,n}^{(2)} = \frac{u_k P_{m,k}^{(n)} \hat{g}_m^{(n)}}{u_k \left(\sum_{l=1, l \neq m}^{|\mathcal{S}_k^{(n)}|} P_{l,k}^{(n)} \hat{g}_l^{(n)} + \sigma^2 \right) + \hat{f}_k^{(n)}}$. Therefore, for a given $\mathbf{p}$, we can equivalently express (16) as

$$\max_{\mathbf{u} \geq 0} \mathcal{F}_k^{(1)}(\mathbf{p}, \mathbf{u}), \text{ s.t. } u_k \geq \frac{1}{Q_{max}}, \forall k. \tag{18}$$

Eq. (18) is a standard DC programming problem. We employ the SCA technique to solve (18). In particular, the first-order Taylor approximation is a global over-estimator of the concave function. Using such a fact, a lower bound to the objective function of (18) is obtained as $\mathcal{F}_k^{(1)}(\mathbf{p}, \mathbf{u}) \geq \mathcal{F}_k^{(2)}(\mathbf{p}, \mathbf{u}, \bar{\mathbf{u}})$, where $\bar{\mathbf{u}} = [\bar{u}_k]_{k \in \mathcal{K}}$ is a vector of the given feasible solution to (18) and where

$$\begin{aligned}
\mathcal{F}_k^{(2)}(\mathbf{p}, \mathbf{u}, \bar{\mathbf{u}}) = & \sum_{m \in \mathcal{S}_k} \omega_m \left(\ln \left(1 + \gamma_{m,k,n}^{(1)}(\mathbf{p}, \mathbf{u}) \right) \right. \\
& \left. - \ln \left(1 + \gamma_{m,k,n}^{(2)}(\mathbf{p}, \bar{\mathbf{u}}) \right) \right) - \sum_{m \in \mathcal{S}_k} \omega_m (u_k - \bar{u}_k) \times \Delta_m^{(1)}(\bar{u}_k) \\
& - \sum_{m \in \mathcal{S}_k^{(n)}} \left(\rho_m + \mu_{n,k} \times \max_{l \in \mathcal{L}_k} h_{m,l}^{(n)} \right) P_{m,k}^{(n)}.
\end{aligned} \tag{19}$$

In (19), $\Delta_m^{(1)}(\bar{u}_k) = \frac{(\gamma_{m,k,n}^{(1)}(\mathbf{p}, \bar{\mathbf{u}}))^2}{1 + \gamma_{m,k,n}^{(1)}(\mathbf{p}, \bar{\mathbf{u}})} \times \frac{\hat{f}_k^{(n)}}{P_{m,k}^{(n)} \hat{g}_m^{(n)}} \times \frac{1}{\bar{u}_k^2}$. An efficient solution to (18) is obtained by iteratively solving the

following sequence of convex optimization problems.

$$\mathbf{u}^{(t+1)} = \arg \max_{\mathbf{u} \geq 0} \mathcal{F}_k^{(2)}(\mathbf{p}, \mathbf{u}, \mathbf{u}^{(t)}), \text{ s.t. } u_k \geq \frac{1}{Q_{max}}, \forall k, \tag{20}$$

where $\mathbf{u}^{(t)}$ denotes the optimal solution for the t -th instance of (20), $\forall t \geq 1$. The optimal solution to (20) is obtained in the following proposition.

Proposition 1: An optimal solution to (20) is obtained at the limit point of the sequence $\{u_{k,j}^{(t+1)}\}$, where $1 \leq j \leq J_{max}$, $\forall t \geq 1$, and $\forall k \in \mathcal{K}$. Considering $u_{k,0}^{(t+1)}$ as a given initial point, each point of this sequence is iteratively computed as $u_{k,j+1}^{(t+1)} = f(u_{k,j}^{(t+1)})$, where

$$f(u_{k,j}^{(t+1)}) = \left[\sqrt{\frac{\sum_{m \in \mathcal{S}_k} \frac{\omega_m (\gamma_{m,k,n}^{(1)}(\mathbf{p}, u_{k,j}^{(t+1)}))^2}{1 + \gamma_{m,k,n}^{(1)}(\mathbf{p}, u_{k,j}^{(t+1)})} \frac{\epsilon |f_d|^2}{P_{m,k}^{(n)} h_{m,k}^{(n)}}}}{\sum_{m \in \mathcal{S}_k} \omega_m \Delta_m^{(1)}(u_{k,j}^{(t)})}} \right]^{\infty} \frac{1}{Q_{max}}, \tag{21}$$

with $[\cdot]_{\frac{1}{Q_{max}}}$ denoting the projection in the interval $[\frac{1}{Q_{max}}, \infty)$.

Proof: The proof is provided in Appendix A.

The jamming power allocation of the CBSs is optimized using the following iterative procedure. At first, the jamming power of the k -th CBS is initialized to some feasible value $\tilde{Q}_k$, and we set $u_k^{(0)} = (\tilde{Q}_k)^{-1}$, $\forall k \in \mathcal{K}$. Thereafter, at the t -th iteration, where $t \in \{1, 2, \dots, T_{max}\}$, $u_k^{(t)}$ is updated according to the limit point of the sequence presented in (21), i.e., $u_k^{(t)} \triangleq u_{k,J_{max}}^{(t-1)}$. The converged jamming power allocation of the k -th CBS is obtained as $Q_k^{(n)} = \frac{1}{u_k^{(T_{max})}}$, $\forall k \in \mathcal{K}$.

2) *Transmit Power Allocation:* For a known jamming power allocation, using [25, Proposition 2], we can express (16) as

$$\max_{\mathbf{p}, \mathbf{a} \geq 0} \mathcal{W}_1 + \mathcal{W}_2 - \psi(\mathbf{p}), \text{ s.t. } P_{m,k}^{(n)} \in [0, P_{max}], \forall m \in \mathcal{S}_k^{(n)}, \tag{22}$$

where $\mathbf{a} = [a_m]_{m \in \mathcal{S}_k^{(n)}}$ is a vector of auxiliary variables; $\mathcal{W}_1$, $\mathcal{W}_2$, and $\psi(\mathbf{p})$ are defined as

$$\mathcal{W}_1 = \sum_{m \in \mathcal{S}_k^{(n)}} \omega_m (\ln(1 + a_m) - a_m), \quad (23)$$

$$\mathcal{W}_2 = \sum_{m \in \mathcal{S}_k^{(n)}} \frac{\omega_m (1 + a_m) \hat{\gamma}_{m,k,n}^{(1)}}{1 + \hat{\gamma}_{m,k,n}^{(1)}} - \omega_m \ln(1 + \hat{\gamma}_{m,k,n}^{(2)}), \quad (24)$$

and

$$\psi(\mathbf{p}) = \sum_{m \in \mathcal{S}_k^{(n)}} \left(\rho_m + \mu_{n,k} \times \max_{l \in \mathcal{L}_k} h_{m,l}^{(n)} \right) P_{m,k}^{(n)}, \quad (25)$$

respectively. In (24), $\hat{\gamma}_{m,k,n}^{(1)} = \frac{P_{m,k}^{(n)} h_{m,k}^{(n)}}{I_{SIC} + I_{Sig} + \epsilon Q_k^{(n)} |f_d|^2 + \sigma^2}$ and $\hat{\gamma}_{m,k,n}^{(2)} = \frac{P_{m,k}^{(n)} \bar{g}_m^{(n)}}{\sum_{l=1, l \neq m}^{|\mathcal{S}_k^{(n)}|} P_{l,k}^{(n)} \bar{g}_l^{(n)} + Q_k^{(n)} \bar{f}_k^{(n)} + \sigma^2}$.

Eq. (22) is decomposed into outer and inner optimization problems as

$$\max_{\mathbf{a} \geq 0} \mathcal{W}_1 + \mathcal{W}_2, \quad (26)$$

and

$$\max_{\mathbf{p}} \mathcal{W}_2 - \psi(\mathbf{p}), \text{ s.t. } P_{m,k}^{(n)} \in [0, P_{max}], \forall m \in \mathcal{S}_k^{(n)}, \quad (27)$$

respectively. Applying the quadratic transformation to the fractions [25, Thm. 1], we can rewrite (27) as

$$\max_{\mathbf{p}, \boldsymbol{\eta} \geq 0} \mathcal{W}_3 - \psi(\mathbf{p}), \text{ s.t. } P_{m,k}^{(n)} \in [0, P_{max}], \forall m \in \mathcal{S}_k^{(n)}, \quad (28)$$

where $\mathcal{W}_3$ is defined in (29) at the top of the next page. In (29), $\boldsymbol{\eta} = [\eta_m]_{m \in \mathcal{S}_k^{(n)}}$ is a vector of auxiliary variables for transforming the fractions into quadratic functions. For a fixed $\boldsymbol{\eta}$, (29) is a non-convex function of the transmit power $\mathbf{p}$. The SCA technique is utilized to solve (29) for a fixed $\boldsymbol{\eta}$. A concave lower bound of $\mathcal{W}_3$, with respect to $\mathbf{p}$, is presented in (30) at the top of the next page. In (30), $\bar{\mathbf{p}}$ is a given feasible transmit power allocation vector. For a fixed $\boldsymbol{\eta}$, an approximate solution to (28) is obtained by iteratively solving the following optimization problem,

$$\max_{\mathbf{p}} \mathcal{W}_3'(\mathbf{p}, \bar{\mathbf{p}}) - \psi(\mathbf{p}), \text{ s.t. } P_{m,k}^{(n)} \in [0, P_{max}], \forall m \in \mathcal{S}_k^{(n)}, \quad (31)$$

and updating $\bar{\mathbf{p}}$ according to the solution obtained from (31).

It is noted that (26), (28), and (31) are the (unconstrained) convex optimization problems with respect to the variables $\mathbf{a}$, $\boldsymbol{\eta}$, and $\mathbf{p}$, respectively. Therefore, the optimal solutions to these optimization problems can be directly obtained at the stationary points of the objective functions. As a result, the optimal values of $\{a_m\}$, $\{\eta_m\}$, and $\{p_m\}$, $\forall m \in \mathcal{S}_k^{(n)}$, are obtained as

$$a_m = \hat{\gamma}_{m,k,n}^{(1)}(\mathbf{p}, \mathbf{q}), \quad (32)$$

$$\eta_m = \frac{\sqrt{\omega_m (1 + a_m) P_{m,k}^{(n)} h_{m,k}^{(n)}}}{P_{m,k}^{(n)} h_{m,k}^{(n)} + I_{SIC} + I_{Sig} + \epsilon Q_k^{(n)} |f_d|^2 + \sigma^2}, \quad (33)$$

Algorithm 1 Transmit and Jamming Power Allocation Algorithm for Solving P4 at the n -th TS

- 1: **Input:** Drone-cluster sets $\{\mathcal{S}_1^{(n)}, \mathcal{S}_2^{(n)}, \dots, \mathcal{S}_K^{(n)}\}$; Lagrangian multipliers $\boldsymbol{\lambda}$, $\boldsymbol{\rho}$, and $\boldsymbol{\mu}$; maximum loop iterations I_{max} , J_{max} , and T_{max} .
 - 2: **Initialize:** Transmit power vector $\mathbf{p}$ and jamming power vector $\mathbf{q}$; outer loop iteration index $i = 1$;
 - 3: **repeat** (Start of outer loop)
 - 4: **for** Each $k \in \{1, 2, \dots, K\}$ **do**
 - 5: Initialize $u_k^{(0)} = \frac{1}{Q_k^{(n)}}$, and iteration index $t = 0$;
 - 6: (Start of inner loop-I)
 - 7: **while** $t < T_{max}$ **do**
 - 8: Update $u_{k,j+1}^{(t+1)}$ by iteratively executing (21), $\forall j = 1, 2, \dots, J_{max}$. Assign $u_k^{(t+1)} = u_{k,J_{max}}^{(t+1)}$.
 - 9: **end while**
 - 10: Update $Q_k^{(n)} = \frac{1}{u_k^{(T_{max})}}$ (End of inner loop-I).
 - 11: Initialize iteration index $t = 0$. (Start of inner loop-II)
 - 12: **while** $t < T_{max}$ **do**
 - 13: Update $\{a_m\}$ and $\{\eta_m\}$, $\forall m \in \mathcal{S}_k^{(n)}$, by plugging current $\mathbf{p}$ and $\mathbf{q}$ to (32) and (33), respectively.
 - 14: Initialize $\bar{P}_{m,k}^{(n)}$, $\forall m \in \mathcal{S}_k^{(n)}$.
 - 15: **for** $J = 1 : J_{max}$ **do**
 - 16: Update $P_{m,k}^{(n)}$ using (34), $\forall m \in \mathcal{S}_k^{(n)}$.
 - 17: Assign $\bar{P}_{m,k}^{(n)} = P_{m,k}^{(n)}$, $\forall m \in \mathcal{S}_k^{(n)}$.
 - 18: **end for**
 - 19: **end while** (End of inner loop-II)
 - 20: **end for**
 - 21: Increase outer loop iteration index $i = i + 1$.
 - 22: **until** $i > I_{max}$ (End of outer loop)
 - 23: **Output:** Transmit power allocation $\{P_{m,k}^{(n)}\}$ and jamming power allocation $\{Q_k^{(n)}\}$, $\forall m \in \mathcal{S}_k^{(n)}$ and $\forall k \in \mathcal{K}$.
-

and

$$P_{m,k}^{(n)} = \left[\frac{\omega_m (1 + a_m) h_{m,k}^{(n)} \eta_m^2}{\left(h_{m,k}^{(n)} \bar{\eta}_m + \omega_m \Delta_m^{(2)}(\bar{\mathbf{p}}, \mathbf{q}) + D_{m,k}^{(n)} \right)^2} \right]_0^{P_{max}}, \quad (34)$$

respectively. In (34), $\bar{\eta}_m = \eta_m^2 + \epsilon_c \sum_{l=1}^{m-1} \eta_l^2 + \sum_{l=m+1}^{|\mathcal{S}_k^{(n)}|} \eta_l^2$, and $D_{m,k}^{(n)} = \rho_m + \mu_{n,k} \times \max_{l \in \mathcal{L}_k} h_{m,l}^{(n)}$. For a known jamming power at the CBSs, a converged transmit power allocation is obtained by alternatively updating the variables $\mathbf{a}$, $\boldsymbol{\eta}$, and $\mathbf{p}$ using (32), (33), and (34), respectively. Such a fact is also confirmed by Proposition 2 of Section V.

3) *Algorithm Development:* The overall steps to solve P4 are summarized in Algorithm 1. Next, we analyze the computational complexity of Algorithm 1. In particular, Algorithm 1 iteratively executes two inner loops, where inner loop-I and inner loop-II update the jamming and transmit power allocations, respectively. The total number of iterations of both inner loops is $T_{max} J_{max}$, and the total number of iterations of the outer loop is I_{max} . The computational complexity associated to inner loop-I is

$$\mathcal{W}_3 = \sum_{m \in \mathcal{S}_k^{(n)}} 2\eta_m \sqrt{\omega_m(1+a_m)P_{m,n}^{(k)}h_{m,n}^{(k)}} - \eta_m^2 \left(P_{m,k}^{(n)}h_{m,n}^{(k)} + I_{SIC} + I_{Sig} + \epsilon|f_d|^2Q_k^{(n)} + \sigma^2 \right) - \omega_m \ln \left(1 + \hat{\gamma}_{m,k,n}^{(2)} \right). \quad (29)$$

$$\begin{aligned} \mathcal{W}'_3(\bar{\mathbf{p}}, \bar{\mathbf{q}}) = & \sum_{m \in \mathcal{S}_k^{(n)}} 2\eta_m \sqrt{\omega_m(1+a_m)P_{m,n}^{(k)}h_{m,n}^{(k)}} - \eta_m^2 \left(P_{m,k}^{(n)}h_{m,n}^{(k)} + I_{SIC} + I_{Sig} + \epsilon|f_d|^2Q_k^{(n)} + \sigma^2 \right) - \omega_m \ln \left(1 + \hat{\gamma}_{m,k,n}^{(2)}(\bar{\mathbf{p}}, \bar{\mathbf{q}}) \right) \\ & - \sum_{m \in \mathcal{S}_k^{(n)}} \omega_m \left(P_{m,k}^{(n)} - \bar{P}_{m,k}^{(n)} \right) \times \underbrace{\left(\frac{\hat{\gamma}_{m,k,n}^{(2)}(\bar{\mathbf{p}}, \bar{\mathbf{q}})}{\bar{P}_{m,k}^{(n)} \left(1 + \hat{\gamma}_{m,k,n}^{(2)}(\bar{\mathbf{p}}, \bar{\mathbf{q}}) \right)} - \sum_{l=1, l \neq m}^{|\mathcal{S}_k^{(n)}|} \frac{\left(\hat{\gamma}_{l,k,n}^{(2)}(\bar{\mathbf{p}}, \bar{\mathbf{q}}) \right)^2}{\bar{P}_{l,k}^{(n)} \left(1 + \hat{\gamma}_{l,k,n}^{(2)}(\bar{\mathbf{p}}, \bar{\mathbf{q}}) \right)} \times \frac{\hat{g}_m^{(n)}}{\hat{g}_l^{(n)}} \right)}_{\Delta_m^{(2)}(\bar{\mathbf{p}}, \bar{\mathbf{q}})}. \end{aligned} \quad (30)$$

$\mathcal{O}(T_{max}J_{max}K)$. Meanwhile, the computational complexity associated to inner loop-II is $\mathcal{O}\left(T_{max}J_{max} \sum_{k=1}^K |\mathcal{S}_k^{(n)}|\right) = \mathcal{O}(T_{max}J_{max}M)$. Therefore, the required computational complexity for executing a single iteration of the outer loop is $\mathcal{O}(T_{max}J_{max}(K+M))$. The overall computational complexity of Algorithm 1 is obtained as $\mathcal{O}(\Delta_{max}(K+M))$ where $\Delta_{max} = T_{max}J_{max}I_{max}$.

C. Optimal Lagrangian Multipliers

We employ the well-known sub-gradient method to minimize $G(\lambda, \rho, \mu)$ and find the optimal Lagrangian multipliers. The update equations of λ , ρ , and μ are expressed as

$$\lambda_m \leftarrow \left[\lambda_m - \xi^{(1)} \left(\frac{1}{N'} \sum_{n=1}^{N'} \sum_{k=1}^K \alpha_{m,k}^{(n)} \tilde{C}_{m,k}^{(n)} - R_s \right) \right]^+, \forall m, \quad (35)$$

$$\rho_m \leftarrow \left[\rho_m + \xi^{(2)} \left(\frac{1}{N'} \sum_{n=1}^{N'} \sum_{k=1}^K \alpha_{m,k}^{(n)} P_{m,k}^{(n)} - P_{avg} \right) \right]^+, \forall m, \quad (36)$$

and

$$\mu_{n,k} \leftarrow \left[\mu_{n,k} + \xi^{(3)} \left(\max_{l \in \mathcal{L}_k} \sum_{m=1}^M \alpha_{m,k}^{(n)} P_{m,k}^{(n)} h_{m,l}^{(n)} - I_{th} \right) \right]^+, \forall n, k, \quad (37)$$

respectively. Here, $\xi^{(1)}$, $\xi^{(2)}$, and $\xi^{(3)}$ are the positive and square-summable step-sizes. It is noted that C1 and C2 are the constraints on the average SR and average transmit power over the TSs, respectively, and C4 is an instantaneous interference threshold constraint. Hence, both λ and ρ are updated only at the end of each TS based on the average SR and average transmit power, respectively. Conversely, μ is iteratively updated along with the instantaneous transmit power of the drones.

IV. DRONE-CLUSTERING SUB-PROBLEM'S SOLUTION

In this section, the (near) optimal drone-clusters are obtained by solving P1 while taking the power allocation solution of Algorithm 1 into account. P1 is a combinatorial optimization problem of exponential complexity $\mathcal{O}(K^M)$. We apply the mixed strategy repeated game framework [26] to solve P1 in a computationally efficient manner. In such a game, at each repeated game round, the CBS associations of each drone are

stochastically varied using a certain probability distribution. This enables the network controller to learn the achievable SR of each drone from different CBSs and select the most suitable CBSs for the drones.

A. Mixed-strategy Repeated Game Formulation

The drone-clusters in the n -th TS, where $n = 1, \dots, N$, are obtained from the solution of a mixed-strategy repeated game $\psi^{(n)} = (\mathcal{M}, \{\mathcal{A}_m^{(n)}\}_{m \in \mathcal{M}}, \{\hat{r}_m^{(n)}\}_{m \in \mathcal{M}})$. Here, the set of legitimate drones $\mathcal{M}$ is the set of players. Each player $m \in \mathcal{M}$ has a set of actions $\mathcal{A}_m^{(n)} = \{a_{m,1}^{(n)}, a_{m,2}^{(n)}, \dots, a_{m,K}^{(n)}\}$, where taking action $a_{m,k}^{(n)}$ implies the selection of the k -th CBS for the m -th player in the n -th TS. The joint action space of the players is denoted as $\mathcal{A}^{(n)} = \mathcal{A}_1^{(n)} \times \mathcal{A}_2^{(n)} \times \dots \times \mathcal{A}_M^{(n)}$. At each round of the game, the m -th player selects an action from $\mathcal{A}_m^{(n)}$, $\forall m \in \mathcal{M}$, leading to a joint action $\underline{a} \in \mathcal{A}^{(n)}$. The utility function of the m -th player for the joint action $\underline{a} \in \mathcal{A}^{(n)}$ is denoted as $\hat{r}_m^{(n)}(\underline{a}) = \frac{C_{sys}^{(n)}}{MKW}$, where $C_{sys}^{(n)}$ is the sum-SR of the network. At each game round, based on the joint action of the players, a set of non-overlapping drone-clusters is formed. Thereafter, Algorithm 1 is executed for these drone-clusters, and near-optimal transmit and jamming power allocations are derived. By plugging such power allocations to the objective function of P1, the utility function of the players is determined.

For each player, there is a probability mass function (PMF) over its action space, known as the mixed strategy profile. Let $\pi_m^{(n)} = \{\pi_{m,1}^{(n)}, \pi_{m,2}^{(n)}, \dots, \pi_{m,K}^{(n)}\}$ be the mixed strategy profile of the m -th player, where $\pi_{m,k}^{(n)} \in [0, 1]$ is the m -th player's probability of selecting the k -th CBS at the n -th TS, and $\sum_{k=1}^K \pi_{m,k}^{(n)} = 1$. The joint mixed strategy profile of the players is denoted by $\pi^{(n)} = \{\pi_1^{(n)}, \pi_2^{(n)}, \dots, \pi_M^{(n)}\}$. Due to the randomized actions of the players, the payoff of a joint mixed strategy profile corresponds to the expected utility of the players. The expected utility of the m -th player in the n -th TS, $\forall m \in \mathcal{M}$ and $n = 1, 2, \dots, N$, is obtained as

$$\bar{r}_m^{(n)}(\pi_m^{(n)}, \pi_{-m}^{(n)}) = \sum_{\underline{a} \in \mathcal{A}^{(n)}} \left(\prod_{j=1}^M \left(\sum_{k=1}^K \pi_{j,k}^{(n)} \varsigma_{j,k}(\underline{a}) \right) \right) \hat{r}_m^{(n)}(\underline{a}), \quad (38)$$

where $\pi_{-m}^{(n)}$ is the joint mixed strategy profile of all the players except the m -th player, and $\varsigma_{j,k}(\underline{a}) \in \{0, 1\}$ is a

binary variable such that $c_{j,k}(\underline{a}) = 1$ if $a_{j,k}^{(n)} \in \underline{a}$, otherwise $c_{j,k}(\underline{a}) = 0$.

From (38), the expected utility of a given player simultaneously depends on its own strategy and the strategies of the opponent players. Essentially, each player's optimal strategy is a function of the opponent players' strategies. We consider that the players are rational and choose the *best-response* strategies. The best-response strategy of the m -th player is defined as $BR_m(\pi_{-m}^{(n)}) \triangleq \arg \max_{\pi_m^{(n)}} \bar{r}_m^{(n)}(\pi_m^{(n)}, \pi_{-m}^{(n)})$. The Nash-equilibrium (NE) solution of the game $\psi^{(n)}$ is formally defined as follows [27, Def. 3. 10].

Definition 1: A joint mixed strategy profile $\hat{\pi}^{(n)} = \{\hat{\pi}_1^{(n)}, \hat{\pi}_2^{(n)}, \dots, \hat{\pi}_M^{(n)}\}$ is NE if, for each player $m \in \mathcal{M}$, $\hat{\pi}_m^{(n)} = BR_m(\hat{\pi}_{-m}^{(n)})$. In particular, in the NE, the following condition is satisfied for each player $m \in \mathcal{M}$.

$$\bar{r}_m^{(n)}(\hat{\pi}_m^{(n)}, \hat{\pi}_{-m}^{(n)}) \geq \bar{r}_m^{(n)}(\pi_m^{(n)}, \hat{\pi}_{-m}^{(n)}), \forall \pi_m^{(n)}. \quad (39)$$

In the NE, all the players jointly maximize the network sum-SR. Therefore, the NE solution of the game $\psi^{(n)}$ is sought to solve P1. To identify the NE solution of the game $\psi^{(n)}$, we first define the smooth best response (SBR) strategy of the m -th player, $\forall m \in \mathcal{M}$, as follows.

Definition 2: For the opponents' fixed joint strategy, $\pi_{-m}^{(n)}$, the SBR strategy of the m -th player is defined as $\beta_m^{(n)}(\pi_{-m}^{(n)}) \triangleq \{\beta_{m,1}^{(n)}(\bar{\mathbf{r}}_m), \beta_{m,2}^{(n)}(\bar{\mathbf{r}}_m), \dots, \beta_{m,K}^{(n)}(\bar{\mathbf{r}}_m)\}$ [28, eq. (2.2)]. Here,

$$\beta_{m,k}^{(n)}(\bar{\mathbf{r}}_m) = \frac{\exp(\kappa_m \bar{r}_m^{(n)}(k, \pi_{-m}^{(n)}))}{\sum_{k=1}^K \exp(\kappa_m \bar{r}_m^{(n)}(k, \pi_{-m}^{(n)}))}, \quad (40)$$

where $\bar{r}_m^{(n)}(k, \pi_{-m}^{(n)})$ is the expected utility of the m -th player obtained from the k -th individual action given that all other players select actions using the joint strategy $\pi_{-m}^{(n)}$; $\bar{\mathbf{r}}_m \triangleq [\bar{r}_m^{(n)}(1, \pi_{-m}^{(n)}), \bar{r}_m^{(n)}(2, \pi_{-m}^{(n)}), \dots, \bar{r}_m^{(n)}(K, \pi_{-m}^{(n)})]$; $\kappa_m > 0$ is a positive parameter.

Let us consider that $\tilde{\pi}^{(n)} = \{\tilde{\pi}_1^{(n)}, \tilde{\pi}_2^{(n)}, \dots, \tilde{\pi}_M^{(n)}\}$ is a stable joint mixed strategy profile for the players, where $\tilde{\pi}_m^{(n)} = \beta_m^{(n)}(\tilde{\pi}_{-m}^{(n)})$, $\forall m \in \mathcal{M}$. It is confirmed by Proposition 3 of Section V that $\tilde{\pi}^{(n)}$ can converge to the ε -equilibrium, where no player can improve its expected utility more than ε by unilaterally deviating from the converged strategy. Specifically, ε can be an arbitrary small value, and hence the ε -equilibrium can be made sufficiently close to the NE. As a result, $\tilde{\pi}^{(n)}$ can approach the NE solution of the game $\psi^{(n)}$.

B. Multi-agent RL Framework to Learn the NE Solution

To compute the NE strategy of the players using (40), prior information about (i) the joint mixed strategy profile used by the opponent players and (ii) the achievable utility of the joint actions in the action space $\mathcal{A}^{(n)}$ are required. Typically such prior information is unknown, and it is non-trivial to compute (40). To circumvent this impediment, we propose a multi-agent RL framework with M virtual agents, where each virtual agent corresponds to a player (i.e., drone). For each agent, actor and

critic RL processes are iteratively executed. Here, the actor and critic RL processes learn the mixed strategy profiles and the value functions of the available actions, respectively [26].

Critic RL process: Let $V_{m,k}^{(n)}[l]$ be the value function for the k -th action of the m -th agent during the l -th round of the game. Specifically, the value functions estimate the expected utility of the available actions. Let us denote $\mathbf{v}_m^{(n)}[l] = [V_{m,1}^{(n)}[l], \dots, V_{m,K}^{(n)}[l]]$. The critic RL process updates the value functions, $\forall m \in \mathcal{M}$ and $k = 1, 2, \dots, K$, as

$$V_{m,k}^{(n)}[l+1] = V_{m,k}^{(n)}[l] + \nu^{(1)}[l] \mathbb{I}_{\{\alpha_{m,k}^{(n)}=1\}} \frac{\hat{r}_m^{(n)} - V_{m,k}^{(n)}[l]}{\pi_{m,k}^{(n)}[l]}, \quad (41)$$

where $\nu^{(1)}[l]$ is the critic's learning step-size at the l -th round; $\pi_{m,k}^{(n)}[l] \in (0, 1)$ is the actor's estimated probability of selecting the k -th action at the l -th round; $\mathbb{I}_{\{\alpha_{m,k}^{(n)}=1\}} = 1$ if the k -th action is selected for the m -th agent during the l -th round, otherwise $\mathbb{I}_{\{\alpha_{m,k}^{(n)}=1\}} = 0$.

Actor RL process: Let $\pi_m^{(n)}[l] = [\pi_{m,1}^{(n)}[l], \pi_{m,2}^{(n)}[l], \dots, \pi_{m,K}^{(n)}[l]]$ be the estimated mixed strategy profile of the m -th agent during the l -th round of the game. The actor RL process updates the estimation of the mixed strategy profile, $\forall m \in \mathcal{M}$ and $k = 1, 2, \dots, K$, as

$$\pi_{m,k}^{(n)}[l+1] = \pi_{m,k}^{(n)}[l] + \nu^{(2)}[l] \left(\beta_{m,k}^{(n)}(\mathbf{v}_m^{(n)}[l]) - \pi_{m,k}^{(n)}[l] \right), \quad (42)$$

where $\nu^{(2)}[l]$ is the actor's learning step-size at the l -th round.

In (41) and (42), the value function and mixed strategy profile are initialized as $\mathbf{v}_m^{(n)}[0] = [0, 0, \dots, 0]$ and $\pi_m^{(n)}[0] = [\frac{1}{K}, \frac{1}{K}, \dots, \frac{1}{K}]$, respectively, $\forall m \in \mathcal{M}$. For the convergence of (41) and (42), the learning step-sizes should satisfy the following conditions: (i) $\sum_{l=1}^{\infty} \nu^{(1)}[l] = \infty$, $\sum_{l=1}^{\infty} (\nu^{(1)}[l])^2 < \infty$, (ii) $\sum_{l=1}^{\infty} \nu^{(2)}[l] = \infty$, $\sum_{l=1}^{\infty} (\nu^{(2)}[l])^2 < \infty$, and (iii) $\lim_{l \rightarrow \infty} \frac{\nu^{(1)}[l]}{\nu^{(2)}[l]} = 0$ [28]. The proposed multi-agent RL framework consists of three steps. In the **action selection step**, for the m -th agent, where $m \in \mathcal{M}$, an action from the $\mathcal{A}_m^{(n)}$ set is sampled using current mixed strategy profiles. In the **utility update step**, the drone-clusters are determined based on the selected actions of the agents. Algorithm 1 is executed for these drone-clusters, and using the derived power allocations, the utility function of the agents is updated. In the **actor-critic update step**, by plugging the utility function in (41) and (42), the value functions and mixed strategy profiles for all the agents are updated. By iteratively repeating the aforementioned three steps at each game round, the network controller learns the NE solution of the game $\psi^{(n)}$.

Remark 1: The parameter κ_m in (40) controls the trade-off between exploration and exploitation of the m -th player, $\forall m \in \mathcal{M}$. It is evident that $\lim_{\kappa_m \rightarrow 0} \beta_{m,k}^{(n)}(\bar{\mathbf{r}}_m) = \frac{1}{K}$, $\forall k = 1, 2, \dots, K$. Therefore, when $\{\kappa_m\}$ is small, the mixed strategy profiles converge to a uniform PMF, and the agents randomly explore different actions with equal probability. In contrast, one can readily show that $\lim_{\kappa_m \rightarrow \infty} \beta_{m,k^*}^{(n)}(\bar{\mathbf{r}}_m) = 1$ and $\lim_{\kappa_m \rightarrow \infty} \beta_{m,k \neq k^*}^{(n)}(\bar{\mathbf{r}}_m) = 0$, where $k^* = \arg \max_{k \in \mathcal{K}} \bar{r}_m^{(n)}(k, \pi_{-m}^{(n)})$. Thus, when $\{\kappa_m\}$ is asymp-

Algorithm 2 Security and Interference Management with REinforcement-learning and NOMA (SIREN) Algorithm

- 1: **Input:** Maximum number of iterations, L_{max} ; coordinates of the stop-over points of the drones; number of TSs, N .
 - 2: **Initialize:** Lagrangian multipliers λ and ρ .
 - 3: **for** Each $n = 1, 2, \dots, N$ **TS do**
 - 4: **Initialize:** $\nu_m^{(n)}[0]$ and $\pi_m^{(n)}[0]$, parameters κ_m , $\forall m \in \mathcal{M}$; iteration index $l = 0$; Lagrangian multiplier μ .
 - 5: **repeat**
 - 6: For each $m \in \mathcal{M}$ drone, sample a CBS using the mixed strategy profile $\pi_m^{(n)}[l]$. Determine a set of drone-clusters, $\{\mathcal{S}_1^{(n)}, \mathcal{S}_2^{(n)}, \dots, \mathcal{S}_K^{(n)}\}$, based on the sampled drone-to-CBS associations.
 - 7: Update the transmit and jamming power allocation by plugging the updated drone-clusters to Algorithm 1.
 - 8: Determine the utility function (i.e., network sum-rate) for the updated drone-clusters and power allocations.
 - 9: Update the value functions and mixed strategy profiles using (41) and (42), respectively.
 - 10: Update the parameter $\kappa_m = a^l - 1$, where $a > 1$
 - 11: Update the Lagrangian multiplier μ using (37); $l = l + 1$;
 - 12: **until** $l > L_{max}$
 - 13: Obtain the converged drone-clusters by associating each drone with the CBS having maximum value function. Obtain the final transmit and jamming power allocation.
 - 14: Update the Lagrangian multipliers λ and ρ using (35) and (36), respectively.
 - 15: **end for**
 - 16: **Output:** Drone-clusters, transmit power allocation of the drones, and jamming power allocation of the CBSs for $n = 1, 2, \dots, N$ TSs.
-

totically large, the best action for each agent is selected in a deterministic manner. To learn the NE mixed strategy profiles, it is imperative to make the agents explore available actions during the initial search, and exploit the best possible actions during the later search [28]. In this context, the value of κ_m , $\forall m \in \mathcal{M}$, is gradually increased at each round of the game.

V. OVERALL RESOURCE ALLOCATION ALGORITHM

A. Overview of the Proposed SIREN Algorithm

We propose a centralized resource allocation algorithm, entitled SIREN, to solve the joint optimization problem P0. The overall steps of SIREN are summarized in Algorithm 2. In each TS, SIREN provides a set of non-overlapping drone-clusters, the transmit power allocation of the drones, and the jamming power allocation of the CBSs. The procedure to determine the resource allocations in a given TS using SIREN is briefly described as follows. In particular, the TS is divided into three intervals, i.e., the transition, scheduling, and transmission intervals. In the transition interval, the drones move from their current stop-over points to the next stop-over points, and collect data. In the scheduling interval, Steps 4-12 of SIREN are iteratively executed in the centralized network controller until the maximum number of iterations

is reached. Following this, the network controller determines a set of non-overlapping drone-clusters at Step 13 of SIREN, and broadcasts the drone-to-CBS association matrix to all the CBSs in the network. Subsequently, each CBS determines the jamming power and the associated drones' transmit power by executing Algorithm 1. Finally, the CBSs inform the associated drones about their transmit power over reliable control channels. Meanwhile, the network controller updates the Lagrangian multipliers, λ and ρ , as per Step 14 of SIREN. In the transmission interval, the drones transmit the collected data to their associated CBSs using the scheduled transmit power, and the CBSs also simultaneously transmit AN signals using the scheduled jamming power. The aforesaid resource allocation procedure is repeated in each TS. We emphasize that to determine the resource allocations for a given TS, SIREN does not need to know the resource allocation decisions of future TSs. Essentially, SIREN is an online algorithm.

B. Properties of the Proposed SIREN Algorithm

1) *Convergence and Optimality:* We first introduce the following two propositions to confirm the effectiveness of SIREN for solving the sub-problems P1 and P2.

Proposition 2: For a given set of drone-clusters, SIREN converges to a near-optimal solution to P2.

Proof: The proof is provided in Appendix B.

Proposition 3: For sufficiently large numbers of iterations, i.e., $L_{max} \rightarrow \infty$, SIREN converges to the NE solution of the drone-clustering game $\psi^{(n)}$, $\forall n = 1, 2, \dots, N$.

Proof: The proof is provided in Appendix C.

Leveraging these two propositions, we confirm the local optimality of SIREN as follows.

Proposition 4: For sufficiently large numbers of iterations, i.e., $L_{max} \rightarrow \infty$, SIREN converges to a local optimal solution of the joint optimization problem P0.

Proof: The proof is provided in Appendix D.

2) *Computational Complexity:* We first determine the required computational complexity for SIREN in each TS. The computational complexity of SIREN in each TS is dominated by Steps 7, 9, 10, 11, 13 and 14. In particular, the required computational complexity for executing Step 7 is $\mathcal{O}(\Delta_{max}(K + M))$. Meanwhile, for each drone, a total of $(K + 1)$ computations are required to update the value functions and mixed strategy profiles. Therefore, the required computational complexity of Step 9 is $\mathcal{O}(MK)$. Finally, Steps 10 and 11 of SIREN require $\mathcal{O}(M)$ and $\mathcal{O}(K)$ computations, respectively. It is recalled that the aforementioned steps are repeated for L_{max} number of times in each TS, where $L_{max} \gg 1$. Conversely, Steps 13 and 14, whose required computational complexity is $\mathcal{O}(\Delta_{max}(K + M))$ and $\mathcal{O}(M)$, respectively, are executed only once in each TS. Therefore, the required computational complexity for SIREN in each TS is approximated as $\mathcal{O}(L_{max}(MK + \Delta_{max}(K + M)))$. As a result, the overall computational complexity of SIREN is obtained as $\mathcal{O}(NL_{max}(MK + \Delta_{max}(K + M)))$. Evidently, SIREN requires a polynomial computational complexity to obtain a converged solution to P0.

TABLE I: Simulation Parameters

Jamming power limit, Q_{max}	10 Watt
Maximum transmit power limit, P_{max}	5 Watt
Drone's average transmit power, P_{avg}	3 Watt
SIC error coefficient, ϵ_c	10^{-5}
SI cancellation coefficient of the CBSs, ϵ	-90 dBm
Acceptable interference level at the CBSs, I_{th}	-80 dBm
Target SR, R_s	5 Mbps
AWGN variance, σ^2	-110 dBm

3) *Signaling Overhead*: In each TS, SIREN requires the following information exchanges. First, at the end of scheduling interval of a TS, the network controller broadcasts the drone-to-CBS association matrix to all the CBSs in the network. This requires a total of MK information exchanges between the network controller and CBSs. Second, at the end of a TS, each CBS coordinates the transmit power allocation of the associated drones using reliable control channels. This requires a total of M information exchanges between the CBSs and drones. Consequently, a total of $NM(K+1)$ information exchanges are required by the SIREN algorithm.

VI. NUMERICAL RESULTS

A. Simulation Setting

For the numerical simulations, we consider a $1500m \times 1500m$ geographical area, where six CBSs are placed at the vertices of a $600m \times 600m$ hexagonal region. We simulate five different system configurations consisting of $M = 5, 10, 15, 20$ and 25 data transmitting drones. Leveraging the circle packing tool [29], the entire geographical area of each system configuration is divided into M non-overlapping sensing zones. Each sensing zone consists of $N = 30$ uniformly distributed stop-over points. For each system configuration, we repeat 25 different simulation trials with independently generated stop-over points at the sensing zones, and present the average WC-ASR over all the independent simulation trials. A circular flight trajectory is considered for the eavesdropping drone. The altitudes of the legitimate and eavesdropping drones are $H_d = 80m$ and $H_e = 50m$, respectively. The height of each CBS is $H_b = 10m$. The estimation error of the eavesdropping drone's 2D positions is $\chi = 10m$, and the A2G/G2A and A2A channels' gains at the reference distance are $\beta_1 = 10^{-5}$ and $\beta_2 = 10^{-4}$, respectively [16]. The SI channel gain, f_d , follows a Rayleigh distribution. Each CBS employs an orthogonal RRB of bandwidth $W = 10$ MHz for aerial communications. Each CBS also shares its aerial RRB with all the other CBSs in the network for uplink cellular communications. The remaining simulation parameters, summarized in TABLE I, are selected based on [5], [6], [16]. Finally, to execute the proposed Algorithms, we set the parameters $\nu^{(1)}(l) = \frac{1}{(l+1)^{0.7}}$, $\nu^{(2)}(l) = \frac{1}{(l+1)^{0.6}}$, $a = 1.1$, $J_{max} = 15$, $T_{max} = 100$, $I_{max} = 100$, and $L_{max} = 50$.

B. Performance Comparison With the Interference-aware Benchmark Power Allocation Schemes

In this sub-section, we demonstrate the efficacy of SIREN in maximizing the achievable WC-ASR of the network compared

Fig. 2: Achievable WC-ASR comparison between SIREN and the interference-aware benchmark power allocation schemes.

to three different interference-aware power allocation schemes. To make a fair comparison, the entire framework of SIREN is used for each benchmark power allocation scheme, while modifying only Step 7 of Algorithm 2. In Fig. 2-a, we compare the achievable WC-ASR of SIREN and iterative function evaluation (IFE)-based power allocation schemes [30] by varying the numbers of drones. In the IFE scheme, the jamming and transmit power allocations are alternately updated, where the CBSs' jamming power allocation is determined using Steps 4-9 of Algorithm 1 and the drones' transmit power allocation is determined by satisfying the first order optimality condition of (16) with respect to p . Fig. 2-a shows that for perfect SIC at the CBSs, both SIREN and IFE schemes obtain large WC-ASR, as expected. It also shows that the IFE scheme provides a higher WC-ASR for small numbers of drones. Conversely, SIREN outperforms the IFE scheme for large numbers of drones. In particular, the inter-drone interference at the drone-clusters becomes substantial as the number of drones increases. Meanwhile, SIREN optimizes the drones' transmit power by leveraging the quadratic transformation of fractional programming problems, which is highly effective for alleviating interference in the network [25]. Thanks to the efficient management of inter-drone interference, SIREN achieves an improved WC-ASR for large numbers of drones. From Fig. 2-a it is observed that when the network has 25 drones, SIREN achieves 51.64% and 42.98% higher WC-ASR than the IFE scheme with perfect and imperfect SIC cases, respectively.

In Fig. 2-b, we compare the achievable WC-ASR of SIREN, greedy binary power control (GBPC), and fractional power control with fixed jamming power (FPC-FJ) methods by varying the numbers of drones. In GBPC [31], the jamming power of the CBSs and the transmit power of the legitimate drones are set to either the maximum or minimum values. To implement GBPC, the minimum and maximum values of the jamming power are set to 2 Watt and 10 Watt, respectively; the minimum and maximum values of the transmit power are

Fig. 3: Achievable WC-ASR of the SIREN, quasi-exhaustive search, and greedy matching based drone-clustering methods.

set to 0.5 Watt and 5 Watt, respectively; Algorithm 1 of [31] is executed to alternately update the jamming and transmit power allocations at each drone-cluster until the maximum number of iterations is reached. In FPC-FJ, the jamming power of the CBS is set to a fixed value of 2 Watt, and the transmit power of the legitimate drones is optimized using Algorithm 3 of [25]. Since GBPC switches between the given maximum and minimum powers, the resultant jamming and transmit powers can be far from the optimal values. Meanwhile, FPC-FJ only takes the interference links into account, and ignores both eavesdropping and jamming links. Intuitively, both GBPC and FPC-FJ are sub-optimal, and thereby SIREN achieves substantial performance gains over both GBPC and FPC-FJ, especially for large numbers of drones. Fig. 2-b depicts that for a network of 25 drones, SIREN achieves 1.99 times and 3.96 times more WC-ASR than the GBPC and FPC-FJ methods, respectively. We conclude that SIREN is interference resilient and particularly advantageous for dense IoD networks.

C. Performance Comparison With the Quasi-exhaustive Search and Matching Empowered Drone-clustering Methods

In Fig. 3-a, we compare the achievable WC-ASR of SIREN and a quasi-exhaustive search empowered drone-clustering method for various numbers of drones. In the quasi-exhaustive search method, in each TS, the network controller determines (almost) all the possible combinations of the drone-clusters, calculates the transmit and jamming power allocations for each combination of drone-clusters, and selects the drone-cluster combination providing the largest WC-ASR of the network. Intuitively, the quasi-exhaustive search method can obtain the optimal WC-ASR, and thus outperforms SIREN. We can show that the computational complexity of the quasi-exhaustive search method is a quadratic function of the number of drones. Therefore, the quasi-exhaustive search method requires substantially high computational complexity for large numbers of drones in the network. Conversely, the required computational complexity of SIREN increases linearly with the numbers of

Fig. 4: Achievable WC-ASR comparison between SIREN and the benchmark RL empowered drone-clustering schemes.

drones. Despite a significant reduction of the computational complexity, SIREN only experiences a small performance loss compared to the quasi-exhaustive search method. Fig. 3-a shows that for 25 drones in the network, SIREN exhibits 3.6% and 6.9% smaller WC-ASR than the quasi-exhaustive search method with perfect and imperfect SIC, respectively. Evidently, SIREN achieves a near-optimal WC-ASR for the network in a computationally efficient manner.

In Fig. 3-b, we plot the achievable WC-ASR of SIREN and a greedy matching empowered drone-clustering method for various numbers of drones. Here, we consider drone-clustering as a two-sided matching problem where each drone aims to associate with the nearest CBS, and each CBS can be associated with a maximum of N_d nearest drones. In particular, we set $N_d = 5$. Such a matching problem is near-optimally solved using a greedy algorithm. It is noted that the greedy matching method avoids iterating between the drone-clustering and power allocation phases, and thus requires a low computational complexity. However, due to the distance based drone-to-CBS association, the drone-clusters generated by the greedy matching method can exhibit large inter-drone interference. As a result, the achievable WC-ASR of the greedy matching method is significantly reduced as the number of drones increases. Fig. 3-b depicts that for 25 drones, SIREN achieves 9.43 times and 9.18 times more WC-ASR than the greedy matching empowered drone-clustering method with perfect and imperfect SIC, respectively. This confirms that SIREN is significantly more efficient than the greedy matching empowered drone-clustering method.

D. Performance Comparison With Benchmark RL Empowered Drone-clustering Schemes

In this sub-section, the superiority of SIREN over two different RL empowered drone-clustering schemes is revealed. In the first benchmark scheme, the drone-clustering sub-problem P1 is solved by leveraging the stochastic learning automata (SLA) framework [32]. In the SLA empowered scheme, the

network controller determines the resource allocation for each TS by iteratively carrying out the following three steps: (i) Update the mixed strategy profiles of the drones by applying [32, eqs. (4), (5)]; (ii) determine a set of drone-clusters using the updated mixed strategy profiles; (iii) calculate the transmit and jamming power allocations by executing Algorithm 1 for the updated drone-clusters. Fig. 4-a compares the achievable WC-ASR of SIREN and SLA empowered scheme for various numbers of drones and six CBSs in the network. In the SLA empowered scheme, at each TS, the mixed strategy profiles of the drones converge to stable PMFs. However, unlike SIREN, SLA does not provide any guarantee on the convergence of the mixed strategy profiles to the SBR strategies. It is noted that the SBR strategies obtain a pure strategy NE solution, and thus near-optimally solve the sub-problem P1. Intuitively, compared to SIREN, the drone-clusters obtained by the SLA empowered method are sub-optimal. Accordingly, SIREN offers notable performance improvements over the SLA empowered method for both small and large numbers of drones. For instance, Fig. 4-a illustrates that for 25 drones, SIREN obtains 20% higher WC-ASR than the SLA empowered method.

In the second benchmark scheme, the drone-clustering sub-problem is solved by applying the **multi-armed bandit (MAB)** learning with **upper confidence bound (UCB)** policy [33]. Particularly, the MAB learning problem has a set of bandits, where each bandit has a set of arms to play at each game round. Here, the goal is to learn suitable strategies for the bandits to play the arms so that the expected reward is maximized. To address the exploration-exploitation dilemma of the bandits, the UCB policy is utilized to select the arms. Such MAB-UCB method can be utilized to solve the sub-problem P1 by considering the drones as bandits and the CBSs as arms. Specifically, to obtain resource allocations using the MAB-UCB method, the network controller iteratively repeats the following two steps in each TS: (i) Select the CBSs for the drones by applying the UCB method [33, eq. (24)] and update the drone-clusters, and (ii) calculate the transmit and jamming power allocations by executing Algorithm 1 for the updated drone-clusters. In Fig. 4-a, we compare the achievable WC-ASR of SIREN and the MAB-UCB method for different numbers of drones. We can readily demonstrate that the MAB-UCB method generates stable drone-clusters. However, such drone-clusters are not necessarily the NE solution of sub-problem P1. Accordingly, the MAB-UCB method exhibits noticeable performance loss compared to the proposed SIREN algorithm. For instance, Fig. 4-a illustrates that for 25 drones, SIREN obtains 34.71% higher WC-ASR than the MAB-UCB method.

Fig. 4-b compares the achievable WC-ASR of SIREN, SLA, and MAB-UCB methods for three CBSs and different numbers of drones in the network. Due to the lower diversity offered by A2G links, the achievable WC-ASR of all three schemes in Fig. 4-b is reduced compared to Fig. 4-a. Nevertheless, SIREN outperforms both SLA and MAB-UCB methods for small numbers of CBSs as well. For instance, Fig. 4-b shows that for 20 drones, SIREN obtains 8.25% and 38.09% higher WC-ASR than the SLA and MAB-UCB methods, respectively. Overall, SIREN is superior than both SLA and MAB-UCB

Fig. 5: Achievable WC-ASR of SIREN versus SI cancellation coefficient, ϵ , for different jamming power allocations.

empowered drone-clustering schemes.

E. Impact of SI Cancellation and Convergence of SIREN

Fig. 5 presents the achievable WC-ASR of SIREN for different values of the SI cancellation coefficient, ϵ , in a network of 20 drones and six CBSs. When the values of ϵ increase, the residual SI at the CBSs caused by the cooperative jamming increases and the channel capacity of the legitimate links strictly decreases. Hence, as observed from Fig. 5, the achievable WC-ASR of SIREN also decreases with the increases in ϵ . It is recalled that SIREN adapts the jamming power of the CBSs while taking the residual SI into account. To show the efficacy of SIREN over the fixed jamming power allocation strategy, the achievable WC-ASR of two SI agnostic schemes is plotted in Fig. 5. In the first SI agnostic scheme, denoted as “Max-JAM”, the CBSs always transmit the AN signals using the maximum jamming power. In the second SI agnostic scheme, denoted as “Low-JAM”, the CBSs always transmit the AN signals using a small jamming power. The fixed jamming power of the CBSs in the Low-JAM scheme is set to 0.1 Watt. When the values of ϵ decrease, the residual SI at the CBSs is reduced and SIREN enables the CBSs to transmit the maximum jamming power. Accordingly, Fig. 5 shows that for sufficiently small values of ϵ , both SIREN and the Max-JAM scheme achieve almost the same WC-ASR. However, due to the severe residual SI at the CBSs, the achievable WC-ASR of the Max-JAM scheme is significantly reduced in the large ϵ regime. Fig. 5 depicts that for $\epsilon = -50$ dBm, SIREN achieves 94.92 times higher WC-ASR than the Max-JAM scheme. Meanwhile, thanks to the small jamming power, the Low-JAM scheme exhibits small residual SI at the CBSs. In consequence, the Low-JAM scheme obtains higher WC-ASR than the Max-Jam scheme when ϵ is large. Conversely, the Max-JAM scheme obtains higher WC-ASR than the Low-JAM scheme in the small ϵ regime. Moreover, the proposed SIREN outperforms the Low-JAM scheme for both small and large values of ϵ . For instance, Fig. 5 shows that for $\epsilon = -50$ dBm, SIREN achieves 4.48 times higher WC-ASR than the Low-JAM scheme. In essence, thanks to the SI

Fig. 6: Iteration numbers of SIREN versus the drone's CBS selection probability in a network of 20 drones and six CBSs.

awareness, SIREN is more efficient than the fixed jamming power allocation schemes.

Fig. 6 illustrates the evolution of the CBS selection probability of a single drone with the iterations in SIREN. At the beginning, the CBS selection probability of the considered drone is $\{\frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}\}$, i.e., the probability of associating the considered drone with a certain CBS is the same for all the CBSs in the network. As the number of iterations increases, the network controller's knowledge of the drone's achievable SR from different CBSs is improved, and thereby the network controller selects the most suitable CBS for the drone. Consequently, the PMF used to select the CBS for the drone(s) becomes deterministic. It is observed in Fig. 6, as the iterations increase, the PMF used to select a CBS for the considered drone almost converges to $\{0, 1, 0, 0, 0, 0\}$, i.e., the drone is associated with the second CBS in the network. Likewise, we can show that the CBS selection probability of other drones also becomes deterministic. This implies that as the number of iterations increases, the drone-clusters become stable. Moreover, when the drone-clusters become stable, the transmit and jamming power allocations also become invariant, and consequently the overall resource allocation converges. We therefore conclude that the convergence of SIREN to a stable resource allocation is guaranteed.

VII. CONCLUSION

We investigated a resource optimization scheme to manage interference and enhance the PLS of multi-drone cellular-connected IoD networks. A joint optimization problem was formulated to maximize the WC-ASR of the network while considering clustering of the legitimate drones, NOMA-enabled data transmission, and cooperative jamming from the FD CBSs. To address the computational intractability of the joint optimization problem, a two-level iterative optimization approach was devised. Specifically, a multi-agent RL framework was employed to optimize the drone-clusters, whereas the transmit power of the legitimate drones and jamming power of the CBSs were optimized by leveraging fractional programming, SCA, and alternating optimization techniques.

A centralized and convergent resource allocation algorithm, entitled SIREN, was proposed to solve the joint optimization problem in an online manner. Simulation results revealed that the proposed SIREN algorithm attains a small performance loss compared to the quasi-exhaustive search empowered method with a significantly reduced computational complexity. Simulation results also confirmed that in dense IoD networks, the proposed SIREN algorithm manages inter-drone interference more efficiently than the existing interference-aware power allocation schemes, and achieves a higher WC-ASR than the well-known SLA and MAB-UCB empowered drone-clustering schemes.

APPENDIX A

It is noted that the objective function of (20) is a concave function with respect to the variable $\mathbf{u}$. Therefore, an optimal solution to (20) is obtained at the stationary point of the objective function. However, a closed-form expression of the stationary point is intractable. Hence, we iteratively update the solution to (20) until a stationary point is reached. We first evaluate the partial derivative of the objective function of (20) as

$$\frac{\partial \mathcal{F}_k^{(2)}(\mathbf{p}, \mathbf{u}, \mathbf{u}^{(t)})}{\partial u_k} = \sum_{m \in \mathcal{S}_k} \frac{\omega_m \left(\gamma_{m,k,n}^{(1)}(\mathbf{p}, u_k) \right)^2}{1 + \gamma_{m,k,n}^{(1)}(\mathbf{p}, u_k)} \frac{\epsilon |f_d|^2}{P_{m,k}^{(n)} h_{m,k}^{(n)}} \frac{1}{u_k^2} - \sum_{m \in \mathcal{S}_k} \omega_m \Delta_m^{(1)}(u_k^{(t)}). \quad (\text{A.1})$$

By setting the R.H.S. of (A.1) to zero, we obtain

$$u_k = \sqrt{\frac{\sum_{m \in \mathcal{S}_k} \frac{\omega_m \left(\gamma_{m,k,n}^{(1)}(\mathbf{p}, u_k) \right)^2}{1 + \gamma_{m,k,n}^{(1)}(\mathbf{p}, u_k)} \frac{\epsilon |f_d|^2}{P_{m,k}^{(n)} h_{m,k}^{(n)}}}{\sum_{m \in \mathcal{S}_k} \omega_m \Delta_m^{(1)}(u_k^{(t)})}}. \quad (\text{A.2})$$

A solution to (A.2) satisfies the stationarity condition, i.e., $\frac{\partial \mathcal{F}_k^{(2)}(\mathbf{p}, \mathbf{u}, \mathbf{u}^{(t)})}{\partial u_k} = 0$. It is observed that by plugging a given value of the variable u_k to the R. H. S. of (A.2) and projecting the R. H. S. of (A.2) to the feasible region of (20), the function iteration in (21) is obtained. Evidently, a fixed point of the function iteration in (21) approaches the stationary point of the objective function of (20). Therefore, to prove of *Proposition 1*, it is suffice to show that the function iteration in (21) converges to a fixed point.

Using some straightforward algebraic manipulations, one can readily verify that the function $f(\cdot)$ in (21) satisfies the following conditions.

- 1) $f(u) > 0, \forall u > 0$.
- 2) $f(u) \geq f(u'), \forall u \geq u'$.
- 3) $\rho f(u) \geq f(\rho u), \forall \rho \geq 1$ and $\forall u > 0$.

Meanwhile, a fixed point of (21) also exists because the optimization problem (20) has at least one local optimal solution. Combining these two facts together and exploiting [34, Thms. 2, 8], we conclude that the iterative solution derived from (21) is guaranteed to converge to a fixed point. This implies that the function iteration in (21) must converge to either a local or global optimal solution to (20). ■

APPENDIX B

We emphasize that a near-optimal solution to P2 is guaranteed as long as P4 is near-optimally solved in each TS. It is noted that at each iteration of a TS, SIREN generates a set of drone-clusters and solves P4 by executing Algorithm 1 for the generated drone-clusters. Therefore, to prove *Proposition 2*, it is suffice to prove that Algorithm 1 obtains a near-optimal solution to P4.

Without loss of generality, we denote the transmit and jamming power allocation vectors at the i -th and $(i+1)$ -th iterations of the outer loop of Algorithm 1 by $(\mathbf{p}^{(i)}, \mathbf{q}^{(i)})$ and $(\mathbf{p}^{(i+1)}, \mathbf{q}^{(i+1)})$, respectively. In particular, Step 5 of Algorithm 1 assigns $u_k^{(0)} = 1/Q_k^{(n),i}$, $\forall k \in \mathcal{K}$, where $Q_k^{(n),i}$ is the k -th element of the jamming power allocation vector $\mathbf{q}^{(i)}$. On the other hand, $\mathbf{q}^{(i+1)}$ is the updated jamming power allocation vector obtained at Step 9 of Algorithm 2. Therefore, in accordance with Section III. B(2), we can write $u_k^{(T_{max})} = 1/Q_k^{(n),i+1}$, $\forall k \in \mathcal{K}$, where $Q_k^{(n),i+1}$ is the k -th element of the jamming power allocation vector $\mathbf{q}^{(i+1)}$.

For the jamming power allocation of the k -th CBS, $\forall k \in \mathcal{K}$, we obtain the following set of inequalities.

$$\begin{aligned} \mathcal{F}_k(\mathbf{p}^{(i)}, \mathbf{q}^{(i)}) &\triangleq \mathcal{F}_k(\mathbf{p}^{(i)}, Q_k^{(n),i}) \\ &\stackrel{(a)}{=} \mathcal{F}_k^{(1)}(\mathbf{p}^{(i)}, u_k^{(0)}) \\ &\stackrel{(b)}{=} \mathcal{F}_k^{(2)}(\mathbf{p}^{(i)}, u_k^{(0)}, u_k^{(0)}) \\ &\stackrel{(c)}{\leq} \mathcal{F}_k^{(2)}(\mathbf{p}^{(i)}, u_k^{(T_{max})}, u_k^{(T_{max}-1)}) \quad (\text{A.3}) \\ &\stackrel{(d)}{\leq} \mathcal{F}_k^{(1)}(\mathbf{p}^{(i)}, u_k^{(T_{max})}) \\ &\stackrel{(e)}{=} \mathcal{F}_k(\mathbf{p}^{(i)}, Q_k^{(n),i+1}) \\ &\triangleq \mathcal{F}_k(\mathbf{p}^{(i)}, \mathbf{q}^{(i+1)}). \end{aligned}$$

In (A.3), (a) is obtained by substituting $u_k^{(0)} = 1/Q_k^{(n),i}$ in (17); (b) is obtained using the fact $\mathcal{F}_k^{(2)}(\cdot, \cdot, \cdot)$ is a minorization function of $\mathcal{F}_k^{(1)}(\cdot, \cdot)$; (c) is obtained using the following inequality, $\mathcal{F}_k^{(2)}(\mathbf{p}, u_k^{(0)}, u_k^{(0)}) \leq \mathcal{F}_k^{(2)}(\mathbf{p}, u_k^{(1)}, u_k^{(0)}) \leq \mathcal{F}_k^{(2)}(\mathbf{p}, u_k^{(1)}, u_k^{(1)}) \leq \dots \leq \mathcal{F}_k^{(2)}(\mathbf{p}, u_k^{(T_{max})}, u_k^{(T_{max}-1)})$, derived according to the concave-convex procedure; (d) is obtained using the first order concavity condition; finally, (e) is obtained by substituting $u_k^{(T_{max})} = 1/Q_k^{(n),i+1}$ in (17).

Meanwhile, we can readily verify that the following condition holds for the transmit power allocation of the k -th drone-cluster, $\forall k \in \mathcal{K}$.

$$\begin{aligned} \mathcal{F}_k(\mathbf{p}^{(i)}, \mathbf{q}^{(i+1)}) &= \mathcal{W}_1(\mathbf{a}(\mathbf{p}^{(i)}, \mathbf{q}^{(i+1)})) \\ &+ \mathcal{W}_3(\mathbf{p}^{(i)}, \mathbf{q}^{(i+1)}, \mathbf{a}(\mathbf{p}^{(i)}, \mathbf{q}^{(i+1)}), \boldsymbol{\eta}(\mathbf{p}^{(i)}, \mathbf{q}^{(i+1)})) \quad (\text{A.4}) \\ &- \psi(\mathbf{p}^{(i)}) \end{aligned}$$

where $\mathbf{a}(\mathbf{p}^{(i)}, \mathbf{q}^{(i+1)})$ is obtained by plugging $\mathbf{p}^{(i)}$ and $\mathbf{q}^{(i+1)}$ to (32), and $\boldsymbol{\eta}(\mathbf{p}^{(i)}, \mathbf{q}^{(i+1)})$ is obtained by plugging $\mathbf{p}^{(i)}$, $\mathbf{q}^{(i+1)}$, and $\mathbf{a}(\mathbf{p}^{(i)}, \mathbf{q}^{(i+1)})$ to (33), respectively. We also observe that Algorithm 1 alternately updates the variable blocks, $\{\mathbf{a}, \boldsymbol{\eta}, \mathbf{p}\}$, in the inner loop-II. For the sake of presentation, we denote $\hat{\mathbf{p}}^{(t)}$ as the power allocation vector obtained from t -th iteration of the inner loop-II. It is recalled that the total number of iterations of the inner loop-II is T_{max} . Therefore,

we can write $\mathbf{p}^{(i)} \triangleq \hat{\mathbf{p}}^{(1)}$ and $\mathbf{p}^{(i+1)} \triangleq \hat{\mathbf{p}}^{(T_{max})}$.

For the transmit power allocation of the k -th drone-cluster, $\forall k \in \mathcal{K}$, we obtain the set of inequalities given as (A.5) at the top of the next page. In (A.5), (a') is obtained by applying (A.4); (b') is obtained using the fact that $\mathcal{W}_3'$ in (30) is a minorization function of $\mathcal{W}_3$ in (29); (c') is obtained using the facts that (31) is a concave optimization problem with respect to the variable $\mathbf{p}$ and (34) maximizes (31); (d') is obtained by utilizing the first order concavity condition; (e') is obtained using the fact that $\boldsymbol{\eta}(\hat{\mathbf{p}}^{(t+1)}, \mathbf{q}^{(i+1)})$, calculated by plugging $\hat{\mathbf{p}}^{(t+1)}$, $\mathbf{q}^{(i+1)}$ and $\mathbf{a}(\hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)})$ to (33), maximizes the function $\mathcal{W}_3$; (f') is obtained using the fact that $\mathbf{a}(\hat{\mathbf{p}}^{(t+1)}, \mathbf{q}^{(i+1)})$, calculated by plugging $\hat{\mathbf{p}}^{(t+1)}$ and $\mathbf{q}^{(i+1)}$ to (32), maximizes the sum of functions $\mathcal{W}_1$ and $\mathcal{W}_3$; finally, (g') is obtained by applying (A.4).

Based on (A.5), we obtain the following inequality for the transmit power allocation of the k -th drone-cluster, $\forall k \in \mathcal{K}$.

$$\begin{aligned} \mathcal{F}_k(\mathbf{p}^{(i)}, \mathbf{q}^{(i+1)}) &\triangleq \mathcal{F}_k(\hat{\mathbf{p}}^{(1)}, \mathbf{q}^{(i+1)}) \\ &\leq \mathcal{F}_k(\hat{\mathbf{p}}^{(2)}, \mathbf{q}^{(i+1)}) \leq \dots \leq \mathcal{F}_k(\hat{\mathbf{p}}^{(T_{max})}, \mathbf{q}^{(i+1)}) \quad (\text{A.6}) \\ &\triangleq \mathcal{F}_k(\mathbf{p}^{(i+1)}, \mathbf{q}^{(i+1)}). \end{aligned}$$

Finally, by combining (A.3) and (A.6), we can write

$$\begin{aligned} \mathcal{F}_0(\mathbf{p}^{(i)}, \mathbf{q}^{(i)}) &= \sum_{k \in \mathcal{K}} \mathcal{F}_k(\mathbf{p}^{(i)}, \mathbf{q}^{(i)}) \\ &\leq \sum_{k \in \mathcal{K}} \mathcal{F}_k(\mathbf{p}^{(i)}, \mathbf{q}^{(i+1)}) \\ &\leq \sum_{k \in \mathcal{K}} \mathcal{F}_k(\mathbf{p}^{(i+1)}, \mathbf{q}^{(i+1)}) \quad (\text{A.7}) \\ &= \mathcal{F}_0(\mathbf{p}^{(i+1)}, \mathbf{q}^{(i+1)}). \end{aligned}$$

Therefore, Algorithm 1 provides a sequence of the transmit and jamming power allocation variables that non-decreasingly improve the objective function of P4. Because of the transmit power budget constraints, an upper bound of the objective function of P4 always exists. Essentially, Algorithm 1 must converge to a near-optimal solution to P4. This completes the required proof of *Proposition 2*. $\blacksquare$

APPENDIX C

At each iteration, SIREN updates the drone-clusters by executing actor-critic RL processes, given by (41) and (42). Therefore, we need to prove that such actor-critic RL processes converge to the NE solution of the game $\psi^{(n)}$. The overall proof has the following two steps. In the first step, we show that when $L_{max} \rightarrow \infty$, the mixed strategy profiles of the drones converge to the SBR strategies, defined in *Definition 2*. In the second step, we show that the SBR strategies converge to the NE solution when $L_{max} \rightarrow \infty$.

First Step: We can readily show that $\lim_{l \rightarrow \infty} \frac{V_{m,k}^{(n)}[l]}{l} = 0$, $\forall m \in \mathcal{M}, k \in \mathcal{K}$. Therefore, the value function iteration in (41) is bounded. Applying [28, Lemma 3.1], we obtain that for the fixed strategy profile, the value function iteration in (41) almost surely converges to the stable solution of the following

$$\begin{aligned}
& \mathcal{F}_k(\hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)}) \\
& \stackrel{(a')}{=} \mathcal{W}_1(\mathbf{a}(\hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)})) + \mathcal{W}_3(\hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)}, \mathbf{a}(\hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)}), \boldsymbol{\eta}(\hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)})) - \psi(\hat{\mathbf{p}}^{(t)}) \\
& \stackrel{(b')}{=} \mathcal{W}_1(\mathbf{a}(\hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)})) + \mathcal{W}_3'(\hat{\mathbf{p}}^{(t)}, \hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)}, \mathbf{a}(\hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)}), \boldsymbol{\eta}(\hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)})) - \psi(\hat{\mathbf{p}}^{(t)}) \\
& \stackrel{(c')}{\leq} \mathcal{W}_1(\mathbf{a}(\hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)})) + \mathcal{W}_3'(\hat{\mathbf{p}}^{(t+1)}, \hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)}, \mathbf{a}(\hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)}), \boldsymbol{\eta}(\hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)})) - \psi(\hat{\mathbf{p}}^{(t+1)}) \\
& \stackrel{(d')}{\leq} \mathcal{W}_1(\mathbf{a}(\hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)})) + \mathcal{W}_3(\hat{\mathbf{p}}^{(t+1)}, \mathbf{q}^{(i+1)}, \mathbf{a}(\hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)}), \boldsymbol{\eta}(\hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)})) - \psi(\hat{\mathbf{p}}^{(t+1)}) \\
& \stackrel{(e')}{\leq} \mathcal{W}_1(\mathbf{a}(\hat{\mathbf{p}}^{(t)}, \mathbf{q}^{(i+1)})) + \mathcal{W}_3(\hat{\mathbf{p}}^{(t+1)}, \mathbf{q}^{(i+1)}, \mathbf{a}(\hat{\mathbf{p}}^{(t+1)}, \mathbf{q}^{(i+1)}), \boldsymbol{\eta}(\hat{\mathbf{p}}^{(t+1)}, \mathbf{q}^{(i+1)})) - \psi(\hat{\mathbf{p}}^{(t+1)}) \\
& \stackrel{(f')}{\leq} \mathcal{W}_1(\mathbf{a}(\hat{\mathbf{p}}^{(t+1)}, \mathbf{q}^{(i+1)})) + \mathcal{W}_3(\hat{\mathbf{p}}^{(t+1)}, \mathbf{q}^{(i+1)}, \mathbf{a}(\hat{\mathbf{p}}^{(t+1)}, \mathbf{q}^{(i+1)}), \boldsymbol{\eta}(\hat{\mathbf{p}}^{(t+1)}, \mathbf{q}^{(i+1)})) - \psi(\hat{\mathbf{p}}^{(t+1)}) \\
& \stackrel{(g')}{=} \mathcal{F}_k(\hat{\mathbf{p}}^{(t+1)}, \mathbf{q}^{(i+1)})
\end{aligned} \tag{A.5}$$

ordinary differential equation (ODE) as $L_{max} \rightarrow \infty$.

$$\dot{v}_{m,k}^{(n)} = \bar{r}_m^{(n)}(k, \boldsymbol{\pi}_{-m}^{(n)}) - v_{m,k}^{(n)}, \forall m \in \mathcal{M}, k \in \mathcal{K}, \tag{A.8}$$

where $v_{m,k}^{(n)} \triangleq \lim_{l \rightarrow \infty} V_{m,k}^{(n)}[l]$ and $\dot{v}_{m,k}^{(n)} \triangleq \lim_{l \rightarrow \infty} \frac{\partial V_{m,k}^{(n)}[l]}{\partial l}$. Meanwhile, we can readily justify that the actor RL process in (42) satisfies all the conditions mentioned in [35, Thm. 2.2]. Therefore, when $L_{max} \rightarrow \infty$, the mixed strategy profile obtained from (42) almost surely converges to the stable solution of the following ODE as well.

$$\dot{\pi}_{m,k}^{(n)} = \beta_{m,k}^{(n)}(v_m^{(n)}) - \pi_{m,k}^{(n)}, \forall m \in \mathcal{M}, k \in \mathcal{K}, \tag{A.9}$$

where $\pi_{m,k}^{(n)} \triangleq \lim_{l \rightarrow \infty} \pi_{m,k}^{(n)}[l]$, $\dot{\pi}_{m,k}^{(n)} \triangleq \lim_{l \rightarrow \infty} \frac{\partial \pi_{m,k}^{(n)}[l]}{\partial l}$, and $\mathbf{v}_m^{(n)} = [v_{m,1}^{(n)}, \dots, v_{m,K}^{(n)}]$. In the proposed RL framework, the actor RL process is slowly updated compared to the critic RL process. Therefore, the convergence of the actor RL process can be analyzed assuming that the critic RL process is already converged to a stable point [26]. Using the fact that $\dot{v}_{m,k}^{(n)} = 0$ at the stationary point of (A.8), the converged value function is obtained as

$$v_{m,k}^{(n)} = \lim_{l \rightarrow \infty} V_{m,k}^{(n)}[l] = \bar{r}_m^{(n)}(k, \boldsymbol{\pi}_{-m}^{(n)}), \forall m \in \mathcal{M}, k \in \mathcal{K}. \tag{A.10}$$

As a result, the converged value function vector is obtained as $\mathbf{v}_m^{(n)} = [\bar{r}_m^{(n)}(1, \boldsymbol{\pi}_{-m}^{(n)}), \dots, \bar{r}_m^{(n)}(K, \boldsymbol{\pi}_{-m}^{(n)})] \triangleq \bar{\mathbf{r}}_m$.

Substituting the converged $\mathbf{v}_m^{(n)}$ to (A.9) and using the fact that $\dot{\pi}_{m,k}^{(n)} = 0$ at the stationary point of (42), the converged mixed strategy profile is obtained as

$$\pi_{m,k}^{(n)} = \lim_{l \rightarrow \infty} \pi_{m,k}^{(n)}[l] = \beta_{m,k}^{(n)}(\bar{\mathbf{r}}_m), \forall m \in \mathcal{M}, k \in \mathcal{K}. \tag{A.11}$$

Consequently, when the number of iterations is asymptotically large, i.e. $L_{max} \rightarrow \infty$, the mixed strategy profiles of the drones jointly converge to the SBR strategies.

Second Step: For the sake of presentation, let us denote $\tilde{\boldsymbol{\pi}}^{(n)} = \{\tilde{\pi}_1^{(n)}, \tilde{\pi}_2^{(n)}, \dots, \tilde{\pi}_M^{(n)}\}$ and $\boldsymbol{\pi}^{(n)} = \{\pi_1^{(n)}, \pi_2^{(n)}, \dots, \pi_M^{(n)}\}$ by the SBR and any other mixed strategy profile for the players in the game $\psi^{(n)}$, respectively. We emphasize that the SBR strategy, defined in (40), is indeed a Boltzmann-Gibbs distribution, which is obtained as

a solution to the following optimization problem.

$$\begin{aligned}
\tilde{\boldsymbol{\pi}}^{(n)} &= \arg \max_{\boldsymbol{\pi}_{-m}^{(n)} \geq 0} \bar{r}_m^{(n)}(\boldsymbol{\pi}_m^{(n)}, \boldsymbol{\pi}_{-m}^{(n)}) \\
&+ \frac{1}{\kappa_m} \sum_{k=1}^K \pi_{m,k}^{(n)} \ln(\pi_{m,k}^{(n)}), \forall m \in \mathcal{M} \\
\text{s.t. } &\sum_{k=1}^K \pi_{m,k}^{(n)} = 1.
\end{aligned} \tag{A.12}$$

Since the SBR strategy is an optimal solution to (A.12), we obtain the following inequality for the m -th player, $\forall m \in \mathcal{M}$.

$$\begin{aligned}
& \bar{r}_m^{(n)}(\tilde{\boldsymbol{\pi}}_m^{(n)}, \tilde{\boldsymbol{\pi}}_{-m}^{(n)}) + \frac{1}{\kappa_m} \sum_{k=1}^K \tilde{\pi}_{m,k}^{(n)} \ln(\tilde{\pi}_{m,k}^{(n)}) \\
& \geq \bar{r}_m^{(n)}(\boldsymbol{\pi}_m^{(n)}, \tilde{\boldsymbol{\pi}}_{-m}^{(n)}) + \frac{1}{\kappa_m} \sum_{k=1}^K \pi_{m,k}^{(n)} \ln(\pi_{m,k}^{(n)}) \\
& \implies \bar{r}_m^{(n)}(\boldsymbol{\pi}_m^{(n)}, \tilde{\boldsymbol{\pi}}_{-m}^{(n)}) - \bar{r}_m^{(n)}(\tilde{\boldsymbol{\pi}}_m^{(n)}, \tilde{\boldsymbol{\pi}}_{-m}^{(n)}) \\
& \leq \frac{1}{\kappa_m} \left(\sum_{k=1}^K \tilde{\pi}_{m,k}^{(n)} \ln(\tilde{\pi}_{m,k}^{(n)}) - \sum_{k=1}^K \pi_{m,k}^{(n)} \ln(\pi_{m,k}^{(n)}) \right).
\end{aligned} \tag{A.13}$$

The R. H. S. of (A.13) is the difference between two entropy functions. Using the fact that the maximum value of an entropy function is $\ln K$, we can express (A.14) as

$$\begin{aligned}
& \bar{r}_m^{(n)}(\boldsymbol{\pi}_m^{(n)}, \tilde{\boldsymbol{\pi}}_{-m}^{(n)}) - \bar{r}_m^{(n)}(\tilde{\boldsymbol{\pi}}_m^{(n)}, \tilde{\boldsymbol{\pi}}_{-m}^{(n)}) \\
& \leq \frac{1}{\kappa_m} \ln K \triangleq \varepsilon, \forall \boldsymbol{\pi}_m^{(n)}.
\end{aligned} \tag{A.14}$$

Eq. (A.14) depicts that the SBR strategy converges to the ε -equilibrium. As per Step 8 of the SIREN algorithm, we obtain $\kappa_m \rightarrow \infty$ and $\varepsilon \rightarrow 0$ when $L_{max} \rightarrow \infty$. This implies that as $L_{max} \rightarrow \infty$, no player can achieve a higher payoff by deviating from the SBR strategy. Therefore, for sufficiently large numbers of iterations, i.e., $L_{max} \rightarrow \infty$, the SBR strategies must converge to the NE solution of the game $\psi^{(n)}$, $\forall n$. This completes the proof of *Proposition 3*. ■

APPENDIX D

We first prove that as $L_{max} \rightarrow \infty$, SIREN converges to a stable solution in each TS. It is recalled that at each iteration of

a TS, the mixed strategy profiles of the drones are iteratively updated. In particular, SIREN has the following two properties.

- First, in accordance with *Proposition 3*, when $L_{max} \rightarrow \infty$, a set of NE mixed strategy profiles is achieved.
- Second, in accordance with Step 8 of SIREN, we obtain $\kappa_m \rightarrow \infty, \forall m \in \mathcal{M}$, when the number of iterations is asymptotically large (i.e., $L_{max} \rightarrow \infty$). It is justified in *Remark 1*, when $\kappa_m \rightarrow \infty$, the m -th drone's probability of selecting the most suitable CBS approaches one. In other words, when $\kappa_m \rightarrow \infty, \forall m \in \mathcal{M}$, a suitable CBS is selected for each drone in a deterministic manner.

By combining the aforementioned properties, it is evident that for numerous iterations per TS, a pure strategy NE solution is achieved. Essentially, when $L_{max} \rightarrow \infty$, the drone-clusters become jointly stable. Due to the stable drone-clusters, the transmit and jamming power allocations become invariant as well. Consequently, given that L_{max} is sufficiently large, the convergence of SIREN to a stable solution in each TS is guaranteed.

Next, we prove that the converged solution obtained by SIREN is also a local optimal solution to P0. It is recalled that P0 is solved using a two-level alternating optimization approach, where the sub-problems P1 and P2 are the upper-level and lower-level sub-problems, respectively. Specifically, as per *Proposition 2*, SIREN near-optimally solves the sub-problem P2 for a given set of drone-clusters generated by sub-problem P1. Therefore, the local optimality of P0 is guaranteed as long as the sub-problem P1 is optimally solved. During the n -th TS, where $n \in \{1, 2, \dots, N\}$, the solution to P1 is obtained by solving the game $\psi^{(n)}$. In the game $\psi^{(n)}$, $\forall n$, all the players (i.e., the drones) share a common utility function. In particular, for the m -th player, $\forall m \in \mathcal{M}$, we readily obtain $\hat{r}_m^{(n)}(a_m, a_{-m}) - \hat{r}_m^{(n)}(a'_m, a_{-m}) = \frac{1}{MK\beta} (C_{sys}^{(n)}(a_m, a_{-m}) - C_{sys}^{(n)}(a'_m, a_{-m}))$, where a_m and a'_m denote two different actions (i.e., cluster selections) of the m -th player, a_{-m} denotes actions of all the players except the m -th player, and $C_{sys}^{(n)}$ is the network sum-SR of the n -th TS. In accordance with [27, Def. 3.17], the game $\psi^{(n)}$ is a weighted-potential game with a potential function $C_{sys}^{(n)}$. In a weighted-potential game, the pure strategy NE solution must either locally or globally maximize the potential function [27]. It is noted that $C_{sys}^{(n)}$ is the objective function of P1. Meanwhile, it is already proved that the stable drone-clusters of each TS, obtained from SIREN, converge to the pure strategy NE solution of the game $\psi^{(n)}$. Therefore, such stable drone-clusters must maximize $C_{sys}^{(n)}$ and solve P1 either locally or globally. The aforementioned fact corroborates that the converged solution obtained by SIREN is a local optimal solution to P0. ■

ACKNOWLEDGMENT

This work was supported in part by funding from the Innovation for Defence Excellence and Security (IDEaS) program from the Department of National Defence (DND). Any opinions and conclusions in this work are strictly those of the author(s) and do not reflect the views, positions, or policies

of-and are not endorsed by-IDEaS, DND, or the Government of Canada.

REFERENCES

- [1] H. Zhang *et al.*, "Cooperation techniques for a cellular Internet of unmanned aerial vehicles," *IEEE Wireless Commun.*, vol. 26, no. 5, pp. 167-173, Oct. 2019.
- [2] U. Challita *et al.*, "Machine learning for wireless connectivity and security of cellular-connected UAVs," *IEEE Wireless Commun.*, vol. 26, no. 1, pp. 28-35, Feb. 2019.
- [3] W. Mei and R. Zhang, "Aerial-ground interference mitigation for cellular-connected UAV," *IEEE Commun. Mag.*, vol. 28, no. 1, pp. 161-173, Feb. 2021.
- [4] J. Yao and N. Ansari, "QoS-aware power control in Internet of drones for data collection service," *IEEE Trans. Veh. Technol.*, vol. 68, no. 7, pp. 6649 - 6656, Jul. 2019.
- [5] J. Yao and N. Ansari, "Online task allocation and flying control in Fog-aided Internet of drones," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5562 - 5569, May 2020.
- [6] Y. Huang *et al.*, "Cognitive UAV communication via joint maneuver and power control," *IEEE Trans. Wireless Commun.*, vol. 67, no. 11, pp. 7872-7888, Nov. 2019.
- [7] W. Mei *et al.*, "Cellular-connected UAV: Uplink association, power control and interference coordination," *IEEE Trans. Wireless Commun.*, vol. 18, no. 11, pp. 5380-5393, Nov. 2019.
- [8] A. S. Matar and X. Shen, "Joint subchannel allocation and power control in licensed and unlicensed spectrum for multi-Cell UAV-cellular network," *IEEE J. Sel. Areas Commun.* (in press).
- [9] B. Li *et al.*, "Secure UAV communication networks over 5G," *IEEE Wireless Commun.*, vol. 26, no. 5, pp. 114-120, Oct. 2019.
- [10] J. Yao and N. Ansari, "Secure federated learning by power control for internet of drones," *IEEE Trans. on Cognitive Commun. and Netw.* (in press).
- [11] G. Zhang *et al.*, "Securing UAV communications via joint trajectory and power control," *IEEE Trans. Wireless Commun.*, vol. 18, no. 2, pp. 1376-1389, Feb. 2019.
- [12] Y. Cai, "Joint trajectory and resource allocation design for energy-efficient secure UAV communication systems," *IEEE Trans. Commun.*, vol. 68, no. 7, pp. 4536-4552, Jul. 2020.
- [13] W. Wang *et al.*, "Robust 3D-trajectory and time switching optimization for dual-UAV-enabled secure communications," *IEEE J. Sel. Areas Commun.* (in press).
- [14] K. Xu *et al.*, "Low-complexity joint power allocation and trajectory design for UAV-enabled secure communications with power splitting," *IEEE Trans. Commun.*, vol. 69, no. 3, pp. 1896-1911, Mar. 2021.
- [15] C. Zhong *et al.*, "Secure UAV communication with cooperative jamming and trajectory control," *IEEE Commun. Lett.*, vol. 23, no. 2, pp. 286-289, Feb. 2019.
- [16] W. Wang *et al.*, "Energy-constrained UAV-assisted secure communications with position optimization and cooperative jamming," *IEEE Trans. Commun.*, vol. 68, no. 7, pp. 4476-4489, July 2020.
- [17] Y. Zhou *et al.*, "Secure communications for UAV-enabled mobile edge computing systems," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 376-388, Jan. 2020.
- [18] Y. Zhou *et al.*, "Robust trajectory and transmit power optimization for secure UAV-enabled cognitive radio networks," *IEEE Trans. Commun.*, vol. 68, no. 7, pp. 4022-4034, July 2020.
- [19] Y. Liu *et al.*, "UAV communications based on non-orthogonal multiple access," *IEEE Wireless Commun.*, vol. 26, no. 1, pp. 52-57, Feb. 2019.
- [20] X. Sun *et al.*, "Physical layer security in UAV systems: Challenges and opportunities," *IEEE Wireless Commun.*, vol. 26, no. 5, pp. 40-47, Oct. 2019.
- [21] N. Zhao *et al.*, "Security enhancement for NOMA-UAV networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 3994-4005, Apr. 2020.
- [22] H-Ming Wang and X. Zhang, "UAV secure downlink NOMA transmissions: A secure users oriented perspective," *IEEE Trans. Commun.*, vol. 68, no. 9, pp. 5732-5746, Sept. 2020.
- [23] Y. Li *et al.*, "Joint resource, trajectory, and artificial noise optimization in secure driven 3D UAVs with NOMA and imperfect CSI," *IEEE J. Sel. Areas Commun.* (in press).
- [24] D. P. Bertsekas and N. R. Sandell, "Estimates of the duality gap for large-scale separable nonconvex optimization problems," *IEEE Conf. Decision Control*, Orlando, FL, Dec. 1982, pp. 782-785.

- [25] K. Shen and W. Yu, "Fractional programming for communication systems-Part I: Power Control and beamforming," *IEEE Trans. Signal Process.*, vol. 66, no. 10, pp. 2616-2630, May 2018.
- [26] S. Perkins *et al.*, "Mixed-strategy learning with continuous action sets," *IEEE Trans. Autom. Control*, vol. 62, no. 1, pp. 379-384, Jan. 2017.
- [27] Z. Han *et al.*, *Game Theory in Wireless and Communication Networks: Theory, Models and Applications*, Cambridge, UK: Cambridge University Press, 2011.
- [28] D. S. Leslie and E. J. Collins, "Individual Q-learning in normal form games," *SIAM J. Cont. Opt.*, vol. 44, no. 2, pp. 495-514, Aug. 2005.
- [29] <http://hydra.nat.uni-magdeburg.de/packing/csq/csq.html>.
- [30] H. Dahrouj *et al.*, "Power spectrum optimization for interference mitigation via iterative function evaluation," *EURASIP J. Wireless Commun. Netw.*, vol. 2012, no. 244, pp. 1-14, Aug. 2012.
- [31] A. Gjendemsj *et al.*, "Binary power control for sum rate maximization over multiple interfering links," *IEEE Trans. Wireless Commun.*, vol. 7, no. 8, pp. 3164-3173, Aug. 2008.
- [32] P. Vrancx *et al.*, "Decentralized learning in Markov games," *IEEE Trans. Systems, Man, and Cybernetics*, vol. 38, no. 4, pp. 976-981, Aug. 2008.
- [33] A. H. Arani *et al.*, "Reinforcement learning for energy-efficient trajectory design of UAVs," *IEEE Internet of Things J.* (in press).
- [34] R. Yates, "A framework for uplink power control in cellular radio systems", *IEEE J. Sel. Areas Commun.*, vol. 13, no. 7, pp. 1341-1347, Sept. 1995.
- [35] V. S. Borkar and S. P. Meyn, "The O.D.E. method for convergence of stochastic approximation and reinforcement learning," *SIAM J. Cont. Opt.*, vol. 38, no. 2, pp. 447-469, Jan. 2000.