UNIVERSITY OF CALIFORNIA,
IRVINE

Leveraging UAVs for 6G Networks

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Electrical and Computer Engineering

by

Carles Diaz Vilor

Dissertation Committee:
Professor Hamid Jafarkhani, Chair
Professor A. Lee Swindlehurst
Assistant Professor Zhiying Wang

2024

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

ACKNOWLEDGMENTS

First, I would like to express my deepest gratitude to my advisor, Prof. Hamid Jafarkhani, for his support and guidance during this journey. Definitely, choosing to pursue a Ph.D. under his guidance has been one of the best decisions I have ever made. He provided me with enough freedom to better understand myself and explore new interests while being responsible with my duties as a student. His mentorship has been invaluable, providing not only academic insights but also enhancing my personal and professional growth. During my studies, I also had the opportunity to collaborate with Prof. Angel Lozano, whose knowledge and insights have been pivotal in the development of my thesis, and with Prof. Ahmed Eltawil, from whom I learned a lot.

In addition, I extend my thanks to the committee members, Prof. A. Lee Swindlehurst and Prof. Zhiying Wang for their constructive feedback, valuable insights, and the time invested in shaping my research.

I sincerely thank my friends for their encouragement and support. Also, my deepest gratitude to my lab mates: Mehdi Ganji, Xun Zou, Lisi Jiang, Saeed Karimi Bidhendi, Weiqi Li, Tomas Ortega, Hossein Maleki, Mojtaba Ahmadi Almasi, Reza Barzegaren, and Berkay Guler for their helpful comments. Special thanks to the Balsells and Holmes Fellowships, and the National Science Foundation awards CCF-1815339, CNS-2209695 for funding my research.

Also, my deepest appreciation goes out to my parents for their encouragement and love. Last but not least, I want to express my gratitude to the rest of my family for their support and encouragement throughout this journey.

VITA

Carles Diaz Vilor

EDUCATION

Doctor of Philosophy in Electrical and Computer Engineering **2024**
University of California, Irvine *Irvine, California*

Master of Science in Telecommunications Engineering **2019**
Universitat Politecnica de Catalunya *Barcelona, Spain*

Bachelor of Science in Telecommunications Engineering **2017**
Universitat Politecnica de Catalunya *Barcelona, Spain*

RESEARCH EXPERIENCE

Graduate Student Researcher **2019–2024**
University of California, Irvine *Irvine, California*

Wireless R&D Intern **Summer-2023**
Qualcomm Inc. *San Diego, California*

Math & Algorithms Intern **Summer-2022**
Nokia - Bell Labs *Murray Hill, New Jersey*

TEACHING EXPERIENCE

Teaching Assistant **Winter–2022**
University of California, Irvine *Irvine, California*

REFEREED JOURNAL PUBLICATIONS

Cell-Free UAV Networks: Asymptotic Analysis and Deployment Optimization **2022**
IEEE Transactions on Wireless Communications

Cell-Free UAV Networks with Wireless Fronthaul: Analysis and Optimization **2023**
IEEE Transactions on Wireless Communications

Sensing and Communication in UAV Cellular Networks: Design and Optimization **2023**
IEEE Transactions on Wireless Communications

A Reinforcement Learning approach for Wildfire Tracking with UAV Swarms **2024**
Submitted a revision; IEEE Transactions on Wireless Communications

Multi-UAV Reinforcement Learning for Data Collection in Cellular MIMO Networks **2024**
Submitted; IEEE Transactions on Wireless Communications

Multi-UAV Energy-Efficient Wildfire Coverage Optimization **2024**
In preparation; IEEE Transactions on Wireless Communications

REFEREED CONFERENCE PUBLICATIONS

Optimal 3D-UAV Trajectory and Resource Allocation of DL UAV-GE Links with Directional Antennas **Dec. 2020**
IEEE Global Communications Conference

On the Deployment Problem in Cell-Free UAV Networks **Dec. 2021**
IEEE Global Communications Conference

A Reinforcement Learning approach for Wildfire Tracking with UAV Swarms **May 2024**
IEEE Communication Theory Workshop

ABSTRACT OF THE DISSERTATION

Leveraging UAVs for 6G Networks

By

Carles Diaz Vilor

Doctor of Philosophy in Electrical and Computer Engineering

University of California, Irvine, 2024

Professor Hamid Jafarkhani, Chair

Advancing towards 6G networks emphasizes integrating communication with sensing functionalities, promising unparalleled connectivity, efficiency, and intelligence in forthcoming networks. In this context, uncrewed aerial vehicles (UAVs) emerge as pivotal assets, offering versatile solutions for both communication and sensing tasks. Leveraging their mobility and flexibility, optimizing the UAV deployment or trajectory can enhance the network performance to meet the new demands.

Anticipating next-generation 6G networks, traditional cellular architectures' constraints have led to the exploration of new network topologies, including cell-free architectures. These architectures abandon the concept of cell, allowing users to connect to multiple base stations and mitigating the effects of cellular boundaries for fairer scenarios. Combining cell-free architectures with UAVs offers substantial performance gains by leveraging UAV adaptability for dynamic coverage and capacity optimization. To fully leverage this potential, we propose a comprehensive framework for cell-free UAV networks. Initially, UAVs operate as flying base stations within a framework of perfect fronthaul connectivity. This paradigm is extended to accommodate wireless fronthaul scenarios, prompting UAVs to function as flying relays instead of flying base stations.

Moreover, UAVs hold significant potential beyond their role in communication. Equipped

with sensors and video cameras, UAVs can serve a dual purpose, enabling efficient data collection and sensing tasks. One critical application is wildfire tracking, addressing the pressing need for early detection and monitoring of wildfires. With the escalating frequency and intensity of wildfires globally, efficient wildfire tracking has become imperative for mitigating their devastating impact. Integrating the strengths of cell-free UAV networks with artificial intelligence, our aim is to optimize UAV trajectories to achieve two primary objectives: (i) cover the fire perimeter with cameras and (ii) ensure reliable transmission of captured images to the network. This design significantly enhances resilience, allowing UAVs to transmit images even if certain base stations are compromised by fire incidents. However, the complexity of the overall problem presents a challenge, leading to the utilization of reinforcement learning in this scenario.

In addition to the aforementioned applications in cell-free networks and wildfire tracking, this dissertation also explores similar scenarios with cellular connectivity. This includes exploring the integration of communication, sensing, and data collection functionalities within traditional cellular networks. These methodologies, which involve optimizing trajectories via traditional techniques or more sophisticated such as reinforcement learning, contribute to enhancing the efficiency and reliability of cellular networks as well.

Chapter 1

Introduction

In the realm of 6G networks, data collection, sensing, and communication stand as pivotal elements for fostering unprecedented advancements in communication technologies [1–3]. New avenues for investigation include the utilization of new spectrum, cell-free solutions, disaggregation and virtualization, energy efficiency, and the application of artificial intelligence (AI) to enhance connectivity across the globe.

This thesis relates a variety of the aforementioned topics under the umbrella of uncrewed aerial vehicles (UAVs). Precisely, UAVs emerge as an ideal solution for diverse challenges in wireless communication networks because of their inherent versatility and mobility [4–8]. In the context of wireless communications and sensor networks, UAVs can navigate diverse terrains and reach locations that traditional methods cannot. In fact, the UAV deployment/trajectory optimization in wireless networks is a problem of growing interest, with different groups studying its impact under different setups [9–30]. Within the wireless context, UAVs can serve as flying base stations, relays, sensors, and more, depending on the specific application requirements. Precisely, all chapters study the UAV deployment/trajectory optimization problem, alongside the resource allocation management, for different scenarios.

Chapters 2 and 3 focus on the static UAV deployment problem in cell-free networks. These type of network architectures, in which every user can potentially communicate with every base station, have received a lot of attention [11, 31–43]. By leaving behind the concept of a cell, it becomes possible to (*i*) provide a much more uniform degree of service, (*ii*) turn interference into useful signal, and (*iii*) further densify the network. Therefore, Chapter 2 considers the uplink scenario in fully and partially centralized cell-free networks, where UAVs function as flying base stations (FBSs). In this setup, a subset of FBSs receives signals from each user, and each FBS serves a subset of users. Exact asymptotic expressions, valid for large subset sizes, and approximate expressions for finite dimensions under Rician fading are derived for the spectral efficiency, which quantifies the amount of information that can be reliably transmitted per unit of bandwidth. Leveraging these expressions, the FBS deployment problem is explored for various receiver architectures. Despite the nonconvex nature of the deployment problem, a combination of gradient-based and Gibbs sampling algorithms yields superior performance compared to a square grid deployment. This performance advantage extends to both minimum and aggregate spectral efficiency metrics for both fully and partially centralized cell-free networks.

The third chapter serves as a natural extension of the second, where the focus shifts to incorporating a wireless fronthaul. This chapter delves into the uplink scenario of cell-free systems, where users establish connections with UAVs that relay information to a processing point via imperfect wireless fronthaul links. Three multiple access options are explored for the fronthaul: frequency division multiple access, spatial division multiple access, and combinations thereof. Deterministic equivalent expressions are derived for spectral efficiency under these fronthaul schemes, considering minimum mean-square error reception (MMSE). Subsequently, optimization subproblems concerning the 3D deployment of UAVs, user transmit powers, and UAV transmit powers are investigated. The joint optimization of these subproblems yields superior performance, with the 3D deployment emerging as the primary driver of improvement.

The fourth chapter considers UAVs as flying users with one main goal: wildfire tracking. Suitably equipped with cameras and sensors, UAVs can be instrumental for wildfire prediction, tracking, and monitoring, provided that uninterrupted connectivity can be guaranteed even if some of the ground access points (APs) are damaged by the fire itself [44–46]. A cell-free network structure, with UAVs connecting to a multiplicity of APs, is therefore ideal in terms of resilience. The fourth chapter proposes a trajectory optimization framework for a UAV swarm tracking a wildfire while maintaining cell-free connectivity with ground APs. Such optimization entails a constant repositioning of the multiplicity of UAVs as the fire evolves to ensure that the best possible view is acquired and transmitted reliably, while respecting altitude limits, avoiding collisions, and proceeding to recharge batteries as needed. Given the complexity and time-varying nature of this multi-UAV trajectory optimization, reinforcement learning is leveraged [47], specifically the twin-delayed deep deterministic policy gradient (TD3) algorithm [48]. The approach is shown to be highly effective for wildfire tracking and coverage and could be likewise applicable to monitor other natural and man-made phenomena, including weather events, earthquakes, or chemical spills.

Next, in Chapters 5 and 6, UAVs are used for sensing and data collection applications in cellular networks [49–52]. However, integrating UAVs in current cellular systems presents major challenges related to trajectory optimization and interference management among others. Chapter 5 considers a multi-cell network including a UAV, which senses and forwards the sensory data from different events to the central base station [53–55]. Particularly, Chapter 5 covers how to design the UAV's (*i*) 3D trajectory, (*ii*) power allocation, and (*iii*) sensing scheduling such that (a) a set of events are sensed, (b) interference to neighboring cells is kept at bay, and (c) the amount of energy required by the UAV is minimized. The resulting nonconvex optimization problem is tackled through a combination of (*i*) low-complexity binary optimization, (*ii*) successive convex approximation, and (*iii*) the Lagrangian method. A similar setup is studied in Chapter 6, where a multi-UAV cellular network is considered. However, Chapter 6 focuses on multiple UAVs for data collection tasks [7, 56–59]. Precisely,

we present an optimization framework that concurrently optimizes UAV trajectories and transmit powers, aiming to collect data from a variety of IoT sensors while (a) minimizing the UAVs flying time and (b) keeping interference to terrestrial networks at bay. The non-convex nature of this problem requires the use of sophisticated approaches, leading us to employ the TD3 algorithm. Experimental results verify the efficacy of the proposed approach, demonstrating its ability to significantly enhance data collection in IoT networks while minimizing the flying time and the interference to ground user links.

Finally, Chapter 7 summarizes the thesis with concluding remarks.

Chapter 2

Cell-Free UAV Networks: Asymptotic Analysis and Deployment Optimization

2.1 Introduction

The evolution towards beyond-5G wireless networks poses challenges such as cooperation, interference, centralization, scalability, and ultra-densification. There is a vast body of literature that addresses these issues within the confines of cellular networks, especially under the frameworks of cooperative communications and interference management [60–68]. These challenges motivate the interest in network structures that transcend the classical cellular paradigm.

One structure that seems suitable to address the aforementioned challenges is the cell-free architecture, in which every user can potentially communicate with every base station [11, 31–43]. More precisely, for the sake of scalability, subsets of users communicate

with subsets of base stations, and the central processing goes one step beyond cooperation. By leaving behind the concept of a cell, it becomes possible to (*i*) provide a much more uniform degree of service, (*ii*) turn interference into useful signal, and (*iii*) further densify the network. The foregoing works, and references therein, tackle different subproblems within cell-free structures, e.g., power allocation, subset creation, or pilot assignment, but a major opportunity remains: optimizing the base station deployment.

FBSs embodied by UAVs are attractive in that respect. Also, FBSs provide superior coverage and can be deployed quickly and on-demand; this latter aspect is especially relevant in situations such as natural or man-caused disasters, when the fixed infrastructure is damaged or outright non-existent.

The challenge of deploying FBSs under various optimality criteria has received considerable attention [14–23, 25, 27, 69–71]. However, with the exception of some recent works such as [11, 41], the UAV literature is restricted to the cellular paradigm and, to the best of our knowledge, the deployment of UAVs as FBSs in cell-free networks has not been thoroughly studied. This chapter aims at filling that gap.

Compared with terrestrial cell-free networks, their aerial counterparts present major differences and new challenges. First, air-to-ground links are dominated by the line-of-sight (LoS) component [72–74] and thus a Rician channel model is needed for these scenarios. This results in larger channel coherence bandwidths relative to those of ground networks [75–77]. The phase of the LoS component, often modelled deterministically to reflect the tracking and correction effected by the receiver's phase lock loop, may have to be modelled stochastically to reflect drifting [41–43].

Second, large arrays are not feasible, which precludes the deployment of massive MIMO; however, smaller directional antennas can be carried onboard [14, 16, 78]. Finally, the computational load at the cloud radio access network (C-RAN) gives rise to a tension between

fully and partially centralized cell-free structures. While the performance of fully centralized systems is superior, certain functionalities such as large matrix inversions or fast channel estimation may represent an excessive burden. Partially centralized systems where some of these tasks are distributed may therefore be preferable whenever the C-RAN is subject to certain constraints.

Focusing on the FBS deployment problem, ingredients that are important, namely imperfect channel state information (CSI) and directional antennas, are explicitly accounted for in this chapter. The analysis entails considering a large number of FBSs and ground users (GUs). However, as a network-wide receiver is not scalable, only a subset of FBSs can participate in the decoding of each user and each FBS can only be involved in the reception of a subset of users. Elaborating on certain random matrix theory results [79–83], deterministic equivalents (finite-dimensional approximations that become exact asymptotically in the subset sizes) for the spectral efficiency under either a deterministic or a random model for the LoS phases are provided. Various receiver architectures are entertained, namely fully centralized MMSE [35–37] and fully/partially centralized maximum ratio combiner (MRC) [32–35]. Capitalizing on these deterministic equivalents, the FBS deployment optimization is studied. Specifically, and with the aim of increasing fairness, the stated objective is to maximize the minimum spectral efficiency across the network. Although the problem is nonconvex, the combination of gradient based (GB) and Gibbs sampling (GS) techniques [84, 85] results in remarkable gains in terms of minimum and sum spectral efficiency. The main contributions of the chapter can be summarized as follows:

- The uplink of cell-free UAV networks is analyzed, including imperfect CSI, MMSE/MRC combining, and a realistic antenna radiation pattern for UAVs.
- Linear MMSE and MRC receivers are formulated and novel asymptotic expressions are provided for the spectral efficiency over Rician channels with deterministic and random phases in the LoS components.

- The cell-free UAV deployment problem is studied. In particular, the maximization of the minimum GU spectral efficiency is formulated. Capitalizing on the obtained expressions for the spectral efficiency, analytical forms are provided for the gradients thereof. Given the nonconvexity of the problem, the gradient updates are combined with the GS technique to avoid low-quality local solutions.

- The relationship between deployment gain and geometry is established. Extensive simulations show the dependence of the former on the latter, analyzing the impact of the number of FBSs and GUs, and of the directivity of the receiver antennas, among others elements.

The remainder of the chapter is organized as follows. Section 2.2 presents the complete system model of the cell-free UAV network and the communication process. Fully and partially centralized cell-free UAV architectures are studied in Sections 2.3 and 2.4, respectively. Section 2.5 focuses on the deployment optimization problem while numerical results are presented and discussed in Section 2.6. Concluding remarks are set forth in Section 2.7.

2.2 System Model

Consider a cell-free network featuring M FBSs, the mth one located at $\boldsymbol{q}_m = (x_m, y_m)$ with altitude H, serving K cochannel single-antenna GUs at $\boldsymbol{w}_k = (\mathrm{x}_k, \mathrm{y}_k)$. The channel coefficient between GU k and single-antenna FBS m is denoted by $g_{k,m}$, drawn from a Rician distribution comprising a dominant LoS component and a Rayleigh-distributed small-scale fading component, such that [86, Sec. 3.4.1]

$$g_{k,m} = \sqrt{\frac{\beta_0\, g_m(d_{k,m})}{d_{k,m}^{\kappa}} \frac{1}{K_{k,m}+1}} \left[\sqrt{K_{k,m}} e^{j\psi_{k,m}} + a_{k,m} \right], \tag{2.1}$$

where β_0 and κ are the path loss at a reference distance of 1 m and the path loss exponent, respectively, while $d_{k,m}$ denotes the distance. The Rician factor is $K_{k,m} = A_1 e^{A_2 \arcsin(\frac{H}{d_{k,m}})}$ for environment-dependent parameters A_1 and A_2 [73] whereas $a_{k,m} \sim \mathcal{N}_{\mathbb{C}}(0,1)$ accounts for the small-scale fading. The antenna gain at the mth FBS is

$$g_m(d_{k,m}) = 2\left(\alpha_m + 1\right) \frac{H^{\alpha_m}}{d_{k,m}^{\alpha_m}}, \tag{2.2}$$

where α_m controls the trade-off between coverage and directivity [78]. The generalization to multi-antenna FBS will be straightforward if the fading is IID within each FBS, as then a multi-antenna FBS can be regarded as multiple collocated single-antenna FBSs. If the fading is not IID within each FBS, then the problem formulation will be the same, with the final results affected by the correlation statistics.

If the LoS phase is perfectly tracked by the FBS, $g_{k,m} \sim \mathcal{N}_{\mathbb{C}}(\overline{g}_{k,m}, r_{k,m})$ with

$$\overline{g}_{k,m} = \mathbb{E}\{g_{k,m}\} \tag{2.3}$$

$$= \sqrt{2(\alpha_m + 1)\,\beta_0 \frac{H^{\alpha_m}}{d_{k,m}^{\alpha_m+\kappa}} \frac{K_{k,m}}{K_{k,m}+1}}\, e^{j\psi_{k,m}}, \tag{2.4}$$

and

$$r_{k,m} = \mathbb{E}\{|g_{k,m} - \overline{g}_{k,m}|^2\} \tag{2.5}$$

$$= 2\left(\alpha_m + 1\right) \beta_0 \frac{H^{\alpha_m}}{d_{k,m}^{\alpha_m+\kappa}} \frac{1}{K_{k,m}+1}. \tag{2.6}$$

Alternatively, if the LoS phase component is more appropriately modelled as random, i.e. $\psi_{k,m} \sim \mathcal{U}[0, 2\pi]$, the channel reduces to a zero-mean r.v. with

$$r_{k,m} = \mathbb{E}\{|g_{k,m}|^2\} \tag{2.7}$$

$$= 2\left(\alpha_m + 1\right) \beta_0 \frac{H^{\alpha_m}}{d_{k,m}^{\alpha_m+\kappa}}. \tag{2.8}$$

2.2.1 Channel Estimation

The channel coherence T_{c} depends on the maximum UAV velocity, $v_{\max}$, the carrier frequency f_{c}, and the speed of light, c, as $T_c \approx \frac{c/f_{\mathrm{c}}}{2v_{\max}}$ [86, Sec. 3.4]. The number of resource units within a fading block is $\tau_{\mathrm{c}} = T_{\mathrm{c}}B_{\mathrm{c}}$ with B_{c} the coherence bandwidth. At $f_c = 2.4$ GHz, and with conservative values $v_{\max} = 10$ m/s and $B_{\mathrm{c}} = 1$ MHz, we will have $\tau_{\mathrm{c}} = 6250$. Other reasonable figures for these parameters will result in similarly large coherence values. Therefore, a large number of orthogonal pilot dimensions are available and pilot contamination can be kept to a minimum [40]. Thereby neglecting contamination, upon observation at the mth FBS of the pilot transmitted by GU k, the linear MMSE channel estimate $\hat{g}_{k,m}$ satisfies $g_{k,m} = \hat{g}_{k,m} + \tilde{g}_{k,m}$, where [87, 88]

$$\hat{g}_{k,m} \sim \mathcal{N}_{\mathbb{C}}(\overline{g}_{k,m}, \gamma_{k,m}), \tag{2.9}$$

with

$$\gamma_{k,m} = \mathbb{E}\{|\hat{g}_{k,m} - \overline{g}_{k,m}|^2\} \tag{2.10}$$

$$= \frac{r_{k,m}^2}{r_{k,m} + \frac{\sigma^2}{p\,\tau}}, \tag{2.11}$$

for given τ and p, denoting the pilot length and transmit power, respectively, while σ^2 is the noise power at the receiver. The uncorrelated error follows $\tilde{g}_{k,m} \sim \mathcal{N}_{\mathbb{C}}(0, c_{k,m})$ with $c_{k,m} = r_{k,m} - \gamma_{k,m}$. Ultimately, if the LoS component experiences a random phase, the channel estimate is zero-mean and $\gamma_{k,m}$ is as in (2.10), with

$$r_{k,m} = 2\,(\alpha_m + 1)\,\beta_0\,\frac{H^{\alpha_m}}{d_{k,m}^{\alpha_m+\kappa}}. \tag{2.12}$$

2.2.2 UL Data Transmission

On a given time-frequency resource, the uplink channel matrix is

$$\boldsymbol{G} = \Big(\boldsymbol{g}_1, \ldots, \boldsymbol{g}_K\Big), \tag{2.13}$$

where $\boldsymbol{g}_k \in \mathbb{C}^{M\times 1}$ is the channel between GU k and all FBSs, satisfying $\boldsymbol{G} = \hat{\boldsymbol{G}} + \tilde{\boldsymbol{G}}$, with $\hat{\boldsymbol{G}}$ and $\tilde{\boldsymbol{G}}$ being the channel estimation and error matrices, respectively. The subset of FBSs participating in the reception of each GU is determined by the binary matrix $\boldsymbol{M}^{(s)} = (\boldsymbol{m}_1^{(\mathrm{s})}, \ldots, \boldsymbol{m}_K^{(\mathrm{s})}) \in \mathbb{Z}_2^{M\times K}$ with entries

$$\Big[\boldsymbol{M}^{(s)}\Big]_{m,k} = \begin{cases} 1 & \text{if } k \in \mathcal{U}_m \\ 0 & \text{otherwise} \end{cases}, \tag{2.14}$$

where $\mathcal{U}_m$ is the set of GUs regarded as signal by the mth FBS. Its complementary matrix is $\boldsymbol{M}^{(i)} = \mathbf{1} - \boldsymbol{M}^{(s)}$, with nonzero entries indicating the GUs that each FBS treats as noise. Pooling the observations from the M FBSs,

$$\boldsymbol{y} = \boldsymbol{M}^{(s)} \circ \boldsymbol{G}\boldsymbol{x} + \boldsymbol{M}^{(i)} \circ \boldsymbol{G}\boldsymbol{x} + \boldsymbol{n} \tag{2.15}$$

$$= \underbrace{\boldsymbol{M}^{(s)} \circ \hat{\boldsymbol{G}}\boldsymbol{x}}_{\text{signal}} + \underbrace{\big(\boldsymbol{M}^{(s)} \circ \tilde{\boldsymbol{G}} + \boldsymbol{M}^{(i)} \circ \boldsymbol{G}\big)\,\boldsymbol{x} + \boldsymbol{n}}_{\text{effective noise: } \boldsymbol{v}}, \tag{2.16}$$

where $\boldsymbol{x} = (\sqrt{p}s_1, \ldots, \sqrt{p}s_K)^{\mathrm{T}}$ with unit power symbols s_k, p denotes the transmit power, $\circ$ is the element-wise product and noise $\boldsymbol{n} \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \sigma^2\boldsymbol{I})$. The effective noise $\boldsymbol{v}$ is zero-mean with covariance $\boldsymbol{\Sigma} = \mathbb{E}\{\boldsymbol{v}\boldsymbol{v}^*\} = \boldsymbol{D}_1 + \boldsymbol{D}_2 + \sigma^2\boldsymbol{I}$ given

$$\boldsymbol{D}_1 = \mathbb{E}\Big\{\big(\boldsymbol{M}^{(s)} \circ \tilde{\boldsymbol{G}}\boldsymbol{x}\big)\big(\boldsymbol{M}^{(s)} \circ \tilde{\boldsymbol{G}}\boldsymbol{x}\big)^*\Big\} \tag{2.17}$$

$$= \mathrm{diag}\left\{\sum_{k\in\mathcal{U}_1} c_{k,1}p, \ldots, \sum_{k\in\mathcal{U}_1} c_{k,M}p\right\}, \tag{2.18}$$

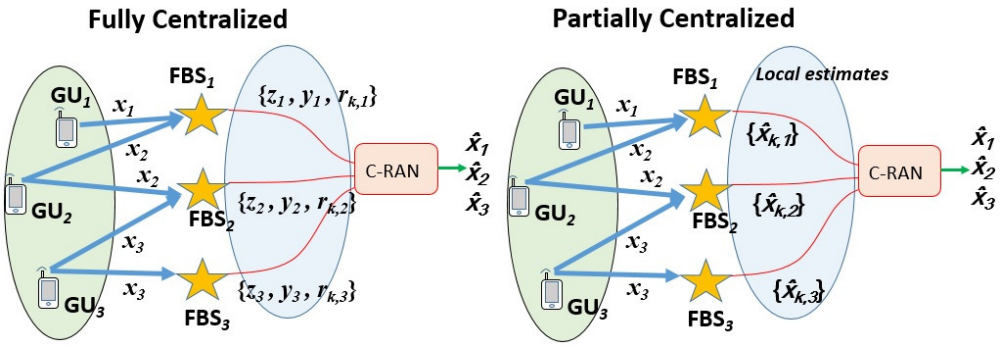


Figure 2.1: Fully and partially centralized networks.

and

$$\boldsymbol{D}_2 = \mathbb{E}\left\{\left(\boldsymbol{M}^{(i)} \circ \boldsymbol{G}\boldsymbol{x}\right)\left(\boldsymbol{M}^{(i)} \circ \boldsymbol{G}\boldsymbol{x}\right)^*\right\} \tag{2.19}$$
$$= (\boldsymbol{M}^{(i)} \circ \overline{\boldsymbol{G}})(\boldsymbol{M}^{(i)} \circ \overline{\boldsymbol{G}})^* p + \operatorname{diag}\left\{\sum_{k \notin \mathcal{U}_1} r_{k,1} p, \ldots, \sum_{k \notin \mathcal{U}_M} r_{k,M} p\right\}.$$

When the LoS components' phases are random, the channel estimates are zero-mean and the above holds with only minor modifications in the statistics of the involved terms.

Based on the C-RAN capabilities, two classes of architectures emerge, namely fully centralized (FC) and partially centralized (PC), as depicted in Fig. 2.1. While the former is superior in performance, the latter is suitable for settings in which some operations are best distributed, e.g., when the C-RAN is itself a flying/orbiting device or when the network of FBSs operates in an ad hoc fashion. In the sequel, we study FC networks where the FBSs convey high-resolution complex observations and PC networks where the FBSs convey the linear combination of data observations with locally gathered channel estimates.

2.3 Fully Centralized Cell-Free Networks

In FC networks, FBSs share the pilot observations, $\boldsymbol{z}_m$, and the collected data signals, y_m with the C-RAN. Let $\mathcal{F}_k = \left\{m : \left[\boldsymbol{M}^{(s)}\right]_{m,k} = 1 \ , \ m = 1, \ldots, M\right\}$ be the subset of FBSs involved in the reception of GU k at the C-RAN. From the rows of $\boldsymbol{y}$ whose indices are in $\mathcal{F}_k$, we obtain the $|\mathcal{F}_k| \times 1$ vector

$$\boldsymbol{y}_k = \boldsymbol{M}_k^{(s)} \circ \hat{\boldsymbol{G}}_k \boldsymbol{x} + \boldsymbol{v}_k, \tag{2.20}$$

where

$$\boldsymbol{M}_k^{(s)} = (\boldsymbol{m}_{k,1}^{(s)}, \ldots, \boldsymbol{m}_{k,K}^{(s)}) \in \mathbb{Z}_2^{|\mathcal{F}_k| \times K}, \tag{2.21}$$

while $\hat{\boldsymbol{G}}_k \in \mathbb{C}^{|\mathcal{F}_k| \times K}$ and $\boldsymbol{v}_k \in \mathbb{C}^{|\mathcal{F}_k| \times 1}$. The combiner that maximizes the SINR is the $|\mathcal{F}_k| \times 1$ MMSE filter [36]

$$\boldsymbol{w}_k = \left(\left(\boldsymbol{M}_k^{(s)} \circ \hat{\boldsymbol{G}}_k\right) \boldsymbol{P} \left(\boldsymbol{M}_k^{(s)} \circ \hat{\boldsymbol{G}}_k\right)^* p + \boldsymbol{\Sigma}_k \right)^{-1} \hat{\boldsymbol{g}}_k \, p, \tag{2.22}$$

where $\hat{\boldsymbol{g}}_k \in \mathbb{C}^{|\mathcal{F}_k| \times 1}$ and $\boldsymbol{\Sigma}_k$ are the downsized versions of the original M-dimensional channel estimate of GU k and the corresponding covariance of the effective noise. From (2.22), the SINR at GU k is

$$\mathrm{SINR}_k = \hat{\boldsymbol{g}}_k^* \left(\sum_{i \neq k} (\boldsymbol{m}_{k,i}^{(s)} \circ \hat{\boldsymbol{g}}_i)(\boldsymbol{m}_{k,i}^{(s)} \circ \hat{\boldsymbol{g}}_i)^* p + \boldsymbol{\Sigma}_k \right)^{-1} \hat{\boldsymbol{g}}_k \, p. \tag{2.23}$$

An alternative combiner is the MRC, $\boldsymbol{w}_k = \hat{\boldsymbol{g}}_k$. For given channel realizations, GU k then attains

$$\mathrm{SINR}_k = \frac{|\hat{\boldsymbol{g}}_k^* \hat{\boldsymbol{g}}_k|^2 \, p}{\sum\limits_{i \neq k} |\hat{\boldsymbol{g}}_k^* (\boldsymbol{m}_{k,i}^{(s)} \circ \hat{\boldsymbol{g}}_i)|^2 \, p + \hat{\boldsymbol{g}}_k^* \boldsymbol{\Sigma}_k \hat{\boldsymbol{g}}_k}. \tag{2.24}$$

From whichever form applies for the SINR, the FC ergodic spectral efficiency of GU k is

$$\mathrm{SE}_k^{\mathrm{fc}} = \left(1 - \frac{\tau}{\tau_c}\right) \mathbb{E}\{\log_2(1 + \mathrm{SINR}_k)\}, \tag{2.25}$$

where $\frac{\tau}{\tau_c}$ accounts for the pilot overhead.

2.3.1 Large-Dimensional Analysis

To evaluate (2.25), we consider the large-dimensional regime, $M, K \to \infty$ with finite M/K, and investigate the convergence of (2.23) and (2.24) to nonrandom limits. This requires that the subsets themselves grow with the network, i.e., $|\mathcal{F}_k|, |\mathcal{U}_m| \to \infty \; \forall \, k, m$, accounting for the non-zero entries in the random matrices. The premises for this convergence are different depending on whether the LoS phases are deterministic [79, 80] or random [82, 83], but they all need the covariance matrix of the channel estimate between User k and all FBSs, defined by

$$\boldsymbol{\Gamma}_k = \mathbb{E}\Big\{ \big(\boldsymbol{m}_k^{(\mathrm{s})} \circ (\hat{\boldsymbol{g}}_k - \overline{\boldsymbol{g}}_k)\big) \big(\boldsymbol{m}_k^{(\mathrm{s})} \circ (\hat{\boldsymbol{g}}_k - \overline{\boldsymbol{g}}_k)\big)^* \Big\} \tag{2.26}$$

$$= \mathrm{diag}\big\{ \gamma_{k,m} m_{k,m}^{(s)} \; \forall m \big\}, \tag{2.27}$$

and $\overline{\boldsymbol{G}}_k$ to satisfy some technical conditions. Specifically, we require that (a) the inverse of the resolvent matrix in (2.23) exists, which is ensured by the presence of $\boldsymbol{\Sigma}_k$, and that (b) $\boldsymbol{\Gamma}_k$ and $\overline{\boldsymbol{G}}$ have uniformly bounded spectral norms. The latter condition prevents the received energy from concentrating on a fixed subset of dimensions as $|\mathcal{F}_k|, |\mathcal{U}_m| \to \infty$ and thus the analysis in the sequel applies under this premise.

Let us first discuss the MMSE-based receiver with deterministic LoS phases, corresponding to (2.23). Recent works have shown the asymptotic behavior of bilinear forms based on the resolvent of large random matrices with nonzero mean, as is the case in (2.23), when

the variance profiles are separable [79, 80]. In cellular MIMO, correlations among the BS antennas are common to all users, based on which asymptotic expressions for $M, K \to \infty$ over Rician channels have been given [81]. In cell-free networks, however, the correlations among antennas at distinct BSs are different for every user, i.e., $[\mathbf{\Gamma}_k]_{m,m} \neq [\mathbf{\Gamma}_k]_{n,n}\, \forall m \neq n$. Therefore, the variance profile is non-separable and none of the previous analyses is valid for cell-free networks when using centralized MMSE with deterministic LoS phases.

On the other hand, when the LoS phase is random and uniformly distributed on $[0, 2\pi]$, the channel estimates are zero-mean. Therefore, application of the derivations in [82, 83] to a cell-free setup yields the following result.

Proposition 1. *For $|\mathcal{F}_k|, |\mathcal{U}_m| \to \infty\ \forall\, k, m$ and MMSE subset combining with random phases in the LoS components,* $\mathrm{SINR}_k - \overline{\mathrm{SINR}}_k \to 0$ *almost surely (a.s.) with*

$$\overline{\mathrm{SINR}}_k = \sum_{m \in \mathcal{F}_k} \frac{\gamma_{k,m}}{\sum\limits_{\substack{i \in \mathcal{U}_m \\ i \neq k}} \frac{\gamma_{i,m}}{1+e_{i,k}} + \sum\limits_{i \in \mathcal{U}_m} c_{i,m} + \sum\limits_{i \notin \mathcal{U}_m} r_{i,m} + \frac{\sigma^2}{p}}. \tag{2.28}$$

The coefficients $e_{j,k}$ are obtained iteratively with $e_{j,k} = \lim_{n\to\infty} e_{j,k}^{(n)}$, $e_{j,k}^{(0)} = M$, and

$$e_{j,k}^{(n)} = \sum_{m \in \mathcal{F}_k} \frac{m_{k,j,m}^{(s)} \gamma_{j,m}}{\sum\limits_{\substack{i \in \mathcal{U}_m \\ i \neq k}} \frac{\gamma_{i,m}}{1+e_{i,k}^{(n-1)}} + \sum\limits_{i \in \mathcal{U}_m} c_{i,m} + \sum\limits_{i \notin \mathcal{U}_m} r_{i,m} + \frac{\sigma^2}{p}}. \tag{2.29}$$

Proof. Details on how (2.28) emanates from [82, 83] can be found in Appendix A.1. □

Next, we turn our attention to the MRC case in (2.24), considering both deterministic and random phases in the LoS components.

Proposition 2. *For $|\mathcal{F}_k|, |\mathcal{U}_m| \to \infty\ \forall\, k, m$ with MRC subset combining and deterministic phases in the LoS components,* $\mathrm{SINR}_k - \overline{\mathrm{SINR}}_k \xrightarrow{a.s.} 0$ *with $\overline{\mathrm{SINR}}_k$ given in* (2.30) *on top of the next page.*

$$\overline{\mathrm{SINR}}_k = \frac{\mathrm{tr}(\overline{\boldsymbol{g}}_k\overline{\boldsymbol{g}}_k^* + \boldsymbol{\Gamma}_k)^2}{\sum\limits_{i\neq k}\mathrm{tr}\big[\big((\boldsymbol{m}_{k,i}^{(s)}\circ\overline{\boldsymbol{g}}_i)(\boldsymbol{m}_{k,i}^{(s)}\circ\overline{\boldsymbol{g}}_i)^* + \boldsymbol{\Gamma}_i\big)(\overline{\boldsymbol{g}}_k\overline{\boldsymbol{g}}_k^* + \boldsymbol{\Gamma}_k)\big] + \mathrm{tr}\big[(\overline{\boldsymbol{g}}_k\overline{\boldsymbol{g}}_k^* + \boldsymbol{\Gamma}_k)\frac{\boldsymbol{\Sigma}_k}{p}\big]}. \tag{2.30}$$

Proof. See Appendix A.2. □

Corollary 1. *For $|\mathcal{F}_k|, |\mathcal{U}_m| \to \infty \ \forall\, k, m$ and MRC subset combiner with random phases in the LoS components,* $\mathrm{SINR}_k - \overline{\mathrm{SINR}}_k \overset{a.s.}{\to} 0$ *with*

$$\overline{\mathrm{SINR}}_k = \frac{\big(\sum_{m\in\mathcal{F}_k}\gamma_{k,m}\big)^2}{\sum_{m\in\mathcal{F}_k}\gamma_{k,m}\big(\sum_i r_{i,m} - \gamma_{k,m} + \frac{\sigma^2}{p}\big)}. \tag{2.31}$$

Proof. The proof follows the procedure in Appendix A.2, by setting the mean components to zero, i.e., $\overline{g}_{k,m} = 0 \ \forall k, m$, and thus with $r_{k,m} = 2(\alpha_m + 1)\,\beta_0\,\frac{H^{\alpha_m}}{d_{k,m}^{\alpha_m+\kappa}}$. Note that $\gamma_{k,m}$ depends on $r_{k,m}$ as derived in (2.10). □

From the continuous mapping theorem [89], the ergodic spectral efficiency in (2.25) satisfies

$$\mathrm{SE}_k^{\mathrm{fc}} - \left(1 - \frac{\tau}{\tau_c}\right)\log_2(1 + \overline{\mathrm{SINR}}_k) \overset{\text{a.s.}}{\to} 0, \tag{2.32}$$

given the applicable form for $\overline{\mathrm{SINR}}_k$. The deterministic equivalents $\overline{\mathrm{SINR}}_k$ and $\mathrm{SE}_k^{\mathrm{fc}}$ depend only on the channel statistics.

2.3.2 Fully Centralized Problem Formulation

Armed with the deterministic equivalent expressions, we can turn to optimizing the FBS deployment. Defining the set of FBS locations by $\boldsymbol{Q} = \{\boldsymbol{q}_m,\ m = 1, \ldots, M\}$, and with the aim of increasing fairness in the network, we can formulate the maximization of the minimum

spectral efficiency as

$$\max_{\boldsymbol{Q}} \ \min_{k} \ \left(1-\frac{\tau}{\tau_c}\right)\mathbb{E}\{\log_2(1+\mathrm{SINR}_k)\}, \tag{2.33}$$

which is nonconvex. We combine GB and GS updates to iteratively relocate the FBSs. Given that the optimization of the spectral efficiency is equivalent to that of the SINR, we obtain the gradients w.r.t. the latter from Props. 1, 2 and Cor. 1. As the combination of GB and GS is common to the partially centralized case (see Sec. 2.5), in the remainder of this section we only derive the gradient expressions for later use.

FC MMSE Deployment

For fully centralized MMSE combining at the C-RAN and random phases in the LoS component, the optimization boils down to

$$\max_{\boldsymbol{Q}} \ \min_{k} \ \sum_{m\in\mathcal{F}_k} \frac{\gamma_{k,m}}{\sum\limits_{\substack{i\in\mathcal{U}_m\\ i\neq k}} \frac{\gamma_{i,m}}{1+e_{i,k}} + \sum\limits_{i\in\mathcal{U}_m} c_{i,m} + \sum\limits_{i\notin\mathcal{U}_m} r_{i,m} + \frac{\sigma^2}{p}}. \tag{2.34}$$

For fixed $e_{i,k}$, the gradient w.r.t. the mth FBS location is

$$\nabla_{\boldsymbol{q}_m}\overline{\mathrm{SINR}}_k = \frac{\nabla_{\boldsymbol{q}_m}\gamma_{k,m}\,\mathrm{Den}_m - \gamma_{k,m}\nabla_{\boldsymbol{q}_m}\mathrm{Den}_m}{\mathrm{Den}_m^2}, \tag{2.35}$$

where

$$\mathrm{Den}_m = \sum_{\substack{i\in\mathcal{U}_m\\ i\neq k}} \frac{\gamma_{i,m}}{1+e_{i,k}} + \sum_{i\in\mathcal{U}_m} c_{i,m} + \sum_{i\notin\mathcal{U}_m} r_{i,m} + \frac{\sigma^2}{p}, \tag{2.36}$$

and $\nabla_{\boldsymbol{q}_m}\mathrm{Den}_m$ is the gradient of (2.36). The derivatives of $\gamma_{k,m}$ and $r_{i,m}$ w.r.t. x_m, can be found in [10] and the derivatives w.r.t. y_m can be obtained similarly, completing (2.35).

FC MRC Deployment

For fully centralized MRC combining at the C-RAN and deterministic phases in the LoS component, the optimization problem reduces to

$$\max_{\boldsymbol{Q}} \min_{k} \overline{\mathrm{SINR}}_k \tag{2.37}$$

with $\overline{\mathrm{SINR}}_k$ given in (2.30) on top of this page. The gradient w.r.t. the mth FBS is

$$\nabla_{\boldsymbol{q}_m} \overline{\mathrm{SINR}}_k = \frac{\nabla_{\boldsymbol{q}_m} \mathrm{Num}_m \, \mathrm{Den}_m - \mathrm{Num}_m \, \nabla_{\boldsymbol{q}_m} \mathrm{Den}_m}{\mathrm{Den}_m^2}, \tag{2.38}$$

where Num_m and Den_m are the numerator and denominator of (2.30), respectively, and $\nabla_{\boldsymbol{q}_m}\mathrm{Num}_m$ and $\nabla_{\boldsymbol{q}_m}\mathrm{Den}_m$ are the corresponding gradients. The computation of (2.38) w.r.t. x_m is sketched in Appendix A.3, and a similar process yields the derivatives w.r.t. y_m and thus the overall gradient in (2.38). All the expressions are simplified further if the LoS components have random phases as in Cor. 1, since then $\overline{g}_{k,m} = 0 \, \forall k, m$.

2.4 Partially Centralized Cell-Free Networks

In PC networks, some combining operations are performed locally at every FBS such that the C-RAN need not be privy to the channel estimates. Each FBS requires channel estimates for the GUs served by that FBS and does not forward the estimates to the C-RAN.

Once the mth FBS observes y_m, it is locally combined with $w_{k,m}$ to yield a signal estimate $\hat{x}_{k,m} = w_{k,m}^* y_m$. Next, $\hat{x}_{k,m}$ is sent through the fronthaul. Once $\{\hat{x}_{k,m} : m \in \mathcal{F}_k\}$ are at the

$$\mathrm{SINR}_k = \frac{|\boldsymbol{v}_k^* \mathbb{E}\{\boldsymbol{\mu}_{k,k}\}|^2 \, p}{\sum\limits_{i \neq k} \mathbb{E}\{|\boldsymbol{v}_k^* (\boldsymbol{m}_{k,i}^{(s)} \circ \boldsymbol{\mu}_{k,i})|^2\} \, p + \mathrm{var}\{\boldsymbol{v}_k^* \boldsymbol{\mu}_{k,k}\} \, p + \boldsymbol{v}_k^* \boldsymbol{\Sigma}_k^{'} \boldsymbol{v}_k}. \tag{2.43}$$

C-RAN, they are combined with $v_{k,m}$ to provide the final signal estimate

$$\hat{x}_k = \sum_{m \in \mathcal{F}_k} v_{k,m}^* \hat{x}_{k,m} \tag{2.39}$$

$$= \sum_{m \in \mathcal{F}_k} v_{k,m}^* w_{k,m}^* g_{k,m} x_k + \sum_{m \in \mathcal{F}_k} v_{k,m}^* \sum_{\substack{i \in \mathcal{U}_m \\ i \neq k}} w_{k,m}^* g_{i,m} x_i + \sum_{m \in \mathcal{F}_k} v_{k,m}^* n_{k,m}^{'}. \tag{2.40}$$

The equivalent noise $n_{k,m}^{'} = w_{k,m}^* (\sum_{i \notin \mathcal{U}_m} g_{i,m} x_i + n_m)$ has covariance $\boldsymbol{\Sigma}_k^{'} = \mathbb{E}\{\boldsymbol{n}_m^{'} \boldsymbol{n}_m^{'*}\}$, where $\boldsymbol{n}_m^{'}$ is the vectorized form of $n_{k,m}^{'}$. We also define the nonzero-mean vector $\boldsymbol{\mu}_{k,i} = \{\mu_{k,i,m} \, : \, m \in \mathcal{F}_k\} = \{w_{k,m}^* \times g_{i,m} \, : \, m \in \mathcal{F}_k\}$ whereby

$$\hat{x}_k = \boldsymbol{v}_k^* \boldsymbol{\mu}_{k,k} x_k + \sum_{i \neq k} \boldsymbol{v}_k^* \left(\boldsymbol{m}_{k,i}^{(s)} \circ \boldsymbol{\mu}_{k,i} \right) x_i + \boldsymbol{v}_k^* \boldsymbol{n}_m^{'}, \tag{2.41}$$

where $\boldsymbol{v}_k$ is the vectorized form of $v_{k,m}$.

Proposition 3. *In a PC network, the highest spectral efficiency achievable by GU k is*

$$\mathrm{SE}_k^{\mathrm{pc}} = \left(1 - \frac{\tau}{\tau_{\mathrm{c}}} \right) \log_2(1 + \mathrm{SINR}_k), \tag{2.42}$$

where SINR_k *is provided in* (2.43).

Proof. See Appendix A.4. □

The combiner required for (2.43) is MMSE-based, yet simpler large-scale-based solutions could be applied, specifically local MRC ($w_{k,m} = \hat{g}_{k,m}$) with equal gain combining (EGC) at the C-RAN ($\boldsymbol{v}_k = \mathbf{1}$).

$$\overline{\mathrm{SINR}}_k = \frac{\mathrm{tr}(\overline{\boldsymbol{g}}_k \overline{\boldsymbol{g}}_k^* + \boldsymbol{\Gamma}_k)^2}{\sum\limits_{i \neq k} \mathrm{tr}\big[(\overline{\boldsymbol{g}}_k \overline{\boldsymbol{g}}_k^* + \boldsymbol{\Gamma}_k)((\boldsymbol{m}_{k,i}^{(s)} \circ \overline{\boldsymbol{g}}_i)(\boldsymbol{m}_{k,i}^{(s)} \circ \overline{\boldsymbol{g}}_i)^* + \boldsymbol{R}_i)\big] + \mathrm{tr}\big[(\overline{\boldsymbol{g}}_k \overline{\boldsymbol{g}}_k^* + \boldsymbol{\Gamma}_k)\frac{\boldsymbol{\Sigma}_k''}{p}\big]}, \quad (2.44)$$

2.4.1 Large-Dimensional Analysis

Expressions can be derived for $|\mathcal{F}_k|, |\mathcal{U}_m| \to \infty$ with local MRC and ECG at the C-RAN under both deterministic and random phases in the LoS components.

Proposition 4. *For $|\mathcal{F}_k|, |\mathcal{U}_m| \to \infty \; \forall \, k, m$, local MRC, and subset ECG at the C-RAN with deterministic LoS phases,* $\mathrm{SINR}_k - \overline{\mathrm{SINR}}_k \overset{a.s.}{\to} 0$ *with* $\overline{\mathrm{SINR}}_k$ *defined in* (2.44) *on top of the next page, where* $\boldsymbol{\Sigma}_k'' = (\boldsymbol{M}_k^{(i)} \circ \overline{\boldsymbol{G}}_k)(\boldsymbol{M}^{(i)} \circ \overline{\boldsymbol{G}}_k)^* p + \boldsymbol{D}_2 + \sigma^2 \boldsymbol{I}$.

Proof. See Appendix A.5. □

Corollary 2. *For $|\mathcal{F}_k|, |\mathcal{U}_m| \to \infty \; \forall \, k, m$, local MRC, and subset ECG at the C-RAN with random LoS phases,* $\mathrm{SINR}_k - \overline{\mathrm{SINR}}_k \overset{a.s.}{\to} 0$ *with*

$$\overline{\mathrm{SINR}}_k = \frac{\big(\sum_{m \in \mathcal{F}_k} \gamma_{k,m}\big)^2}{\sum_{m \in \mathcal{F}_k} \gamma_{k,m} \big(\sum_{i \neq k} r_{i,m} + \frac{\sigma^2}{p}\big)}. \quad (2.45)$$

Proof. The proof follows the procedure in Appendix A.5, only with $\overline{g}_{k,m} = 0 \;\; \forall k, m$ and thus with $r_{k,m} = 2(\alpha_m + 1)\, \beta_0 \, \frac{H^{\alpha_m}}{d_{k,m}^{\alpha_m + \kappa}}$. Note that $\gamma_{k,m}$ depends on $r_{k,m}$ as derived in (2.10). □

From the continuous mapping theorem [89], the spectral efficiency in Prop. 3 satisfies

$$\mathrm{SE}_k^{\mathrm{pc}} - \left(1 - \frac{\tau}{\tau_c}\right) \log_2(1 + \overline{\mathrm{SINR}}_k) \overset{\mathrm{a.s.}}{\to} 0, \quad (2.46)$$

given the applicable form for $\overline{\mathrm{SINR}}_k$.

2.4.2 Problem Formulation

The maximization of the minimum spectral efficiency, hence the minimum SINR, entails

$$\max_{\boldsymbol{Q}} \quad \min_{k} \ \overline{\mathrm{SINR}}_k \tag{2.47}$$

with $\overline{\mathrm{SINR}}_k$ defined in (2.44) on top of this page and

$$\nabla_{\boldsymbol{q}_m} \overline{\mathrm{SINR}}_k = \frac{\nabla_{\boldsymbol{q}_m} \mathrm{Num}_m \mathrm{Den}_m - \mathrm{Num}_m \nabla_{\boldsymbol{q}_m} \mathrm{Den}_m}{\mathrm{Den}_m^2}, \tag{2.48}$$

where Num_m and Den_m are the numerator and denominator of (2.44), respectively, with $\nabla_{\boldsymbol{q}_m} \mathrm{Num}_m$ and $\nabla_{\boldsymbol{q}_m} \mathrm{Den}_m$ the corresponding gradients. The computation of (2.48) is akin to that of (2.38), which is detailed in Appendix A.3. If random phases are considered in the LoS components, a similar optimization problem can be formulated based on Cor. 2.

2.5 GB-GS Deployment

Equipped with the gradients for each of the scenarios, at iteration number j, the GB updates are

$$\boldsymbol{q}_m^{(j)} \leftarrow \boldsymbol{q}_m^{(j)} + \rho^{(j)} \nabla_{\boldsymbol{q}_m} \overline{\mathrm{SINR}}_k^{(j)} \big|_{\boldsymbol{q}_m = \boldsymbol{q}_m^{(j)}}, \tag{2.49}$$

where $\rho^{(j)}$ is a decreasing function of j for convergence reasons. Due to the lack of convexity, the updates in (2.49) may quickly converge to low-quality solutions. Furthermore, an exhaustive search on the 2D plane at altitude H would be computationally prohibitive. Therefore, affordable techniques should be investigated. A potential alternative is simulated annealing [10]. However, the exploration in simulated annealing is limited to only one possible direction. Although the results are promising, exploring other locations may result in

a better cost function. We thus resort to stochastic optimization, and more particularly to the GS technique [84], to improve the results. Given a discrete search space Θ, GS aims at solving

$$\max_{\boldsymbol{Q}\in\Theta} \min_{k} \overline{\mathrm{SINR}}_k. \tag{2.50}$$

The FBS locations are iteratively updated according to a certain probability distribution [84]. Precisely, at Iteration t, the coordinates of the FBSs serving the GU with the lowest $\overline{\mathrm{SINR}}^{(t)}$ are updated; the index of this GU is denoted by $k_{\min}^{(t)}$.

Let us define by $\boldsymbol{q}_m^{(t)}$ and $\eta^{(t)} = \min_k \overline{\mathrm{SINR}}_k^{(t)} = \overline{\mathrm{SINR}}_{k_{\min}^{(t)}}^{(t)}$ the location of FBS m and the cost function at Iteration t, respectively. By definition, $\eta^{(t)}$ is a function of the FBSs associated to GU $k_{\min}^{(t)}$, i.e., $\eta^{(t)}(\boldsymbol{q}_m^{(t)}, m \in \mathcal{F}_{k_{\min}^{(t)}})$. The coordinates of the $\mathcal{F}_{k_{\min}^{(t)}}$ FBSs are updated sequentially, starting at the FBSs with the lowest index. For a given $m \in \mathcal{F}_{k_{\min}^{(t)}}$, let us denote by $\boldsymbol{Q}_{-m,k_{\min}^{(t)}}^{(t)} = \{\boldsymbol{q}_j^{(t+1)}, \boldsymbol{q}_i^{(t)} : j < m, i > m, (j,i) \in \mathcal{F}_{k_{\min}^{(t)}}\}$ the set of FBS locations associated to GU $k_{\min}^{(t)}$ where

a) FBSs $j < m$ have already updated their locations to $t+1$,

b) the locations of FSBs $i > m$ still need to be updated, and

c) FBS m is removed.

The cost function $\eta^{(t)}(\boldsymbol{q}_m^{(t)}, m \in \mathcal{F}_{k_{\min}^{(t)}})$ can be written as a function of the set $\boldsymbol{Q}_{-m,k_{\min}^{(t)}}^{(t)}$, namely $\eta^{(t)}(\boldsymbol{q}_m^{(t)}, \boldsymbol{Q}_{-m,k}^{(t)})$. Then, the probability of FBS $m \in \mathcal{F}_{k_{\min}^{(t)}}$ updating its location to $\boldsymbol{q}_m^{(t+1)}$ can be calculated as [84]

$$\Pr\left\{\boldsymbol{q}_m^{(t+1)} | \boldsymbol{q}_m^{(t)}, \boldsymbol{Q}_{-m,k_{\min}^{(t)}}^{(t)}\right\} = \frac{\exp\left\{\gamma\, \eta^{(t)}\left(\boldsymbol{q}_m^{(t+1)}, \boldsymbol{Q}_{-m,k}^{(t)}\right)\right\}}{\sum\limits_{\hat{\boldsymbol{q}}_m^{(t+1)}\in\Theta} \exp\left\{\gamma\, \eta^{(t)}\left(\hat{\boldsymbol{q}}_m^{(t+1)}, \boldsymbol{Q}_{-m,k}^{(t)}\right)\right\}} \; \forall m \in \mathcal{F}_{k_{\min}^{(t)}}, \tag{2.51}$$

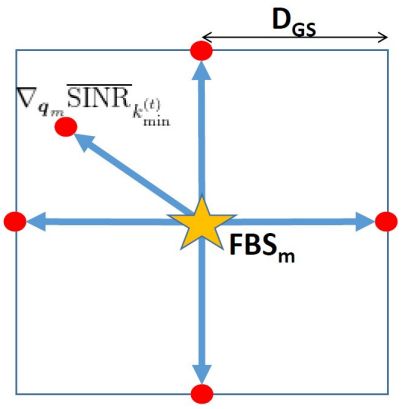


Figure 2.2: Two-dimensional search space

.

where γ is a fixed parameter and Θ^{t+1} represents the possible locations that FBS m can explore at iteration $t+1$. To reduce the search space, we limit the number of possible neighbors to six, i.e., $|\Theta^{t+1}| = 6$ (see Fig. 2.2). The six alternatives are to stay, move north, move south, move east, move west, and move in the direction of the gradient in (2.49). Note that the gradient can point at any location in the 2D plane at altitude H. Hence, the search space is continuous. Moreover, at each iteration, matrix $\boldsymbol{M}^{(s)}$ is updated. A summary of the iterative process is included in Alg. 1 where ϵ and $I_{\max}$ control when to stop.

Algorithm 1 GS-GB Algorithm

Require: At $t = 0$, initial FBS locations, $\boldsymbol{q}_m^{(0)}$, and cost function $\eta^{(0)}$.
  while $\frac{|\eta^{(t+1)} - \eta^{(t)}|}{\eta^{(t)}} > \epsilon$ or $t < I_{\max}$ **do**
    Find the user with lowest cost function, $k_{\min}^{(t)}$ and the FBS connected to it, $\mathcal{F}_{k_{\min}^{(t)}}$.
    for all $m \in \mathcal{F}_{k_{\min}^{(t)}}$ **do**
      Obtain $\boldsymbol{Q}^{(t)}_{-m,k_{\min}^{(t)}}$.
      The reduced search space Θ^{t+1} with six possible locations is created.
      The cost function at the six possible new locations is computed: $\eta^{(t)}\left(\hat{\boldsymbol{q}}_m^{(t+1)}, \boldsymbol{Q}^{(t)}_{-m,k}\right)$ for $\hat{q}_m^{(t+1)} \in \Theta^{t+1}$.
      Calculate (2.51) and choose one movement accordingly. Thus, $\boldsymbol{q}_m^{(t+1)}$ is obtained.
    end for
  end while

2.5.1 Complexity

The proposed algorithm combines the complexity of GB and GS. For the latter, the complexity of updating a given FBS location is proportional to the number of states in the search space, $\mathcal{O}\big(|\Theta|\big)$ with $|\Theta| = 6$ in our case. For the former, in a FC network, obtaining the $|\mathcal{F}_{k_{\min}^{(t)}}|$ FBS gradients for GU $k_{\min}^{(t)}$ has a complexity of $\mathcal{O}\big(|\mathcal{F}_{k_{\min}^{(t)}}|K\big)$ under MMSE and $\mathcal{O}\big(|\mathcal{F}_{k_{\min}^{(t)}}|K + K + |\mathcal{F}_{k_{\min}^{(t)}}|\big)$ under MRC in terms of complex multiplications. For PC networks, the complexity order coincides with that of FC MRC. The overall complexities, including GB and GS and given a maximum number of iterations $I_{\max}$, are, after keeping the dominant terms, $\mathcal{O}\big(\sum_{t=1}^{I_{\max}}|\mathcal{F}_{k_{\min}^{(t)}}|K\big)$ for MMSE and MRC.

2.6 Simulation Results

For the purpose of performance evaluation, we consider a 1 km^2 universe, wrapped around to avoid boundary effects. Unless otherwise specified, Table 2.1 lists the typical parameters used in the simulations, selected based on the cell-free and UAV literature [33, 90–93]. The channel coherence is larger than that of ground networks because aerial settings exhibit higher coherence bandwidths; this, in turn, enables channel estimation with low pilot contamination. Henceforth, contamination is thereby neglected, and readers interested in results that include it are referred to [10]. Unless otherwise specified, and, consistently with the neglect of contamination, we consider a higher value of τ; precisely, $\tau = 200$ for a 3.2% pilot overhead. The $[m, k]$ entry of $\boldsymbol{M}^{(s)}$ is 1 if $d_{k,m} \leq R_{\max}$ for $R_{\max} = 400$ m, which ensures connectivity to multiple FBSs per GU. Regarding the GB-GS implementation, $\rho^{(t)}$ must be a decreasing function of t for convergence reasons, in this case $\rho^{(t)} = 10 \cdot 1.005^{-t}$ though other choices would work as well. The value of D_{GS} is set to $\mathrm{D}_{\mathsf{GS}} = 0.6$ m (see Fig. 2.2) while $\gamma = 30$, $\epsilon = 0.01$ and the maximum number of iterations is $I_{\max} = 500$. Varying these parameters would shift the tradeoff between speed of convergence and final

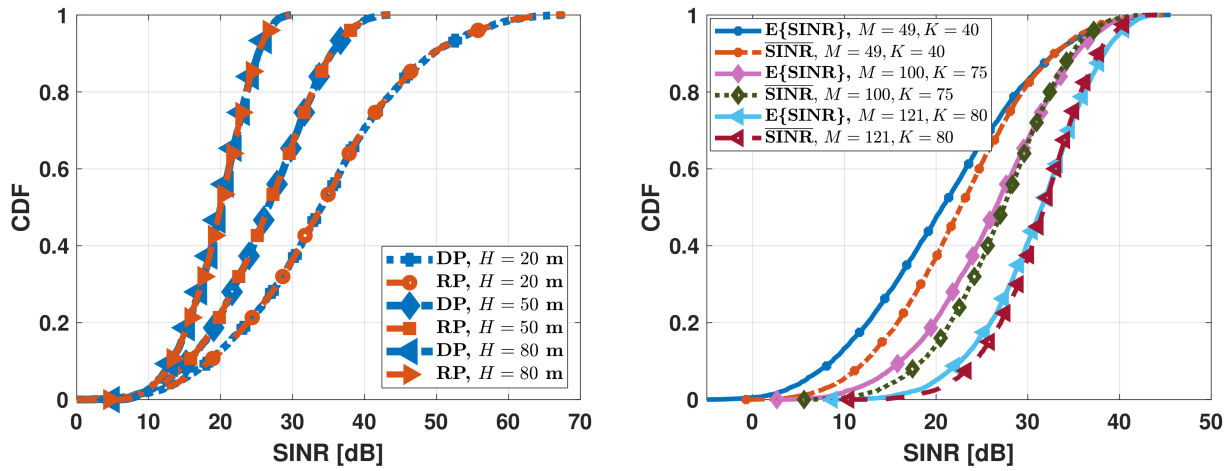
Table 2.1: Simulation Parameters

Description	Parameter	Value
GU transmit power	p	10 mW
Path loss at 1 m	β_0	-30 dB
Path loss exponent	κ	2.2
Dense urban parameters	A_1, A_2	0, 6.4 dB
Noise power	σ^2	-120 dBm
Antenna beamwidth	α_m	4
Operating frequency	f_{c}	2.4 GHz
Maximum FBS velocity	$v_{\max}$	10 m/s
Coherence bandwidth	B_{c}	1 MHz
Number of FBSs	M	100
Number of GUs under MMSE reception	K	75
Number of GUs under MRC reception	K	40

cost function; for example, diminishing D_{GS} increases the resolution of the search space and thus the number of iterations, while increasing γ improves the chances of the GS algorithm choosing the direction that yields the highest cost function, lowering the stochastic component of the method. Additionally, the entries of $\boldsymbol{M}^{(s)}$ are updated at every iteration of the GS-GB algorithm following the aforementioned distance-based rule. The GU locations abide by a Poisson Point Process (PPP) and we test our optimization algorithm over 100 deployments, sufficiently many to gauge the performance. The optimized deployment is denoted by a-opt (after optimization) while a square grid FBS deployment, denoted by b-opt (before optimization), serves as a benchmark.

2.6.1 FC Networks

Before proceeding to the deployment optimization in FC networks, we measure the gap in terms of average SINR between the deterministic phase (DP) and the random phase (RP) models for the LoS components. This is presented in Fig. 2.3a for different values of H, given $M = 100$, $K = 75$, and FC MMSE reception. The gap is only 0.1–0.5 dB, as there are enough pilot symbols to accurately estimate the channels. The high coherence of aerial



(a) (b)

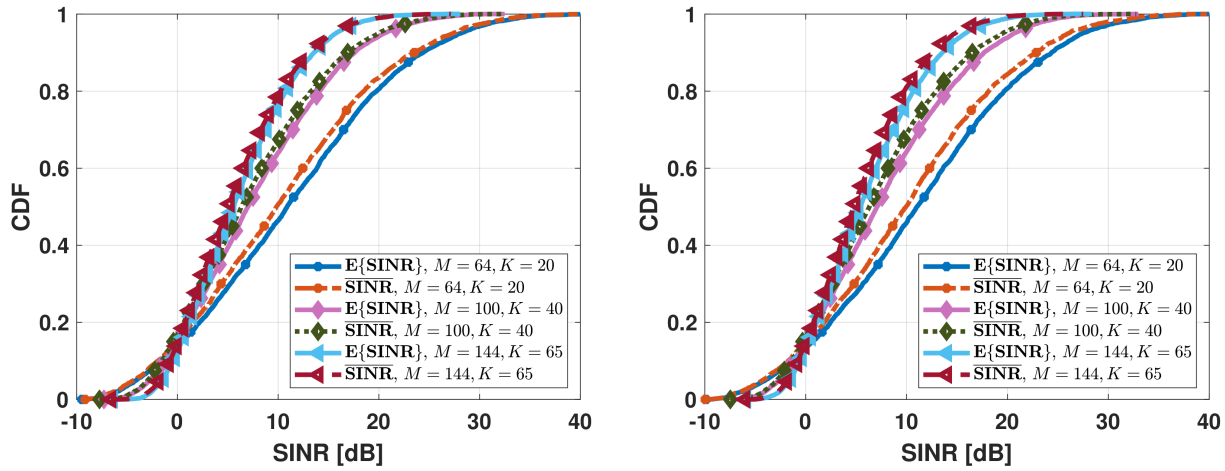
Figure 2.3: (a) $\mathbb{E}\{\text{SINR}_k\}$ under the DP and RP models, parameterized by H, with $M = 100$ and $K = 75$; (b) $\mathbb{E}\{\text{SINR}_k\}$ vs $\overline{\text{SINR}}_k$ for FC MMSE reception under the RP model for $H = 50$ m.

channels enables large values for τ, in contrast with [42, 43], where extremely low values are assumed.

Then, Fig. 2.3b shows that relatively small networks, with their correspondingly small subsets, suffice for the deterministic equivalent in Prop. 1 to be accurate.

The counterpart of Fig. 2.3b for MRC reception is presented in Fig. 2.4 for DP and RP models. Figs. 2.4a and 2.4b relate to the deterministic equivalents in Prop. 2 and Cor. 1, respectively. The network loads are kept at $|\bar{\mathcal{U}}_m|/|\bar{\mathcal{F}}_k| \approx M/K \approx 0.7$–$0.8$ with MMSE reception and $|\bar{\mathcal{U}}_m|/|\bar{\mathcal{F}}_k| \approx M/K \approx 0.3$–$0.4$ with MRC reception, as per the guidelines in [36]. Unless otherwise specified in the rest of simulations, we assume $M = 100$ and $K = 75$ in MMSE reception and $M = 100$ and $K = 40$ for the MRC case.

Next, in Figs. 2.5 and 2.6, we test our proposed GB-GS optimization in FC networks with those loads and under the DP model. In Fig. 2.5, different values of κ and H are considered while, in Fig. 2.6, the variation is in α_m and H. Similar results are presented in Figs. 2.7 and 2.8 for the RP model.



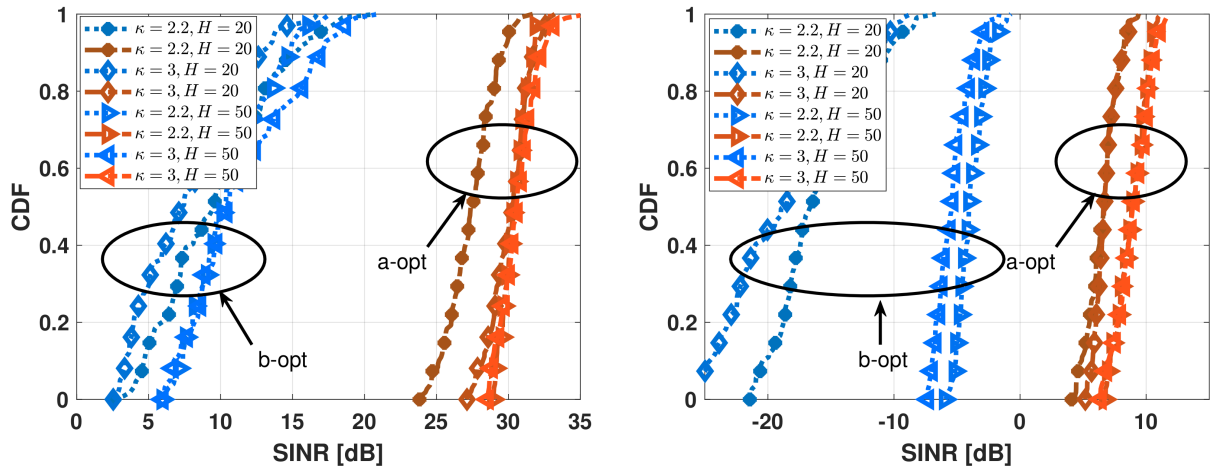
(a) (b)

Figure 2.4: $\mathbb{E}\{\text{SINR}_k\}$ vs $\overline{\text{SINR}}_k$ at $H = 50$ m for FC MRC reception under (a) the DP model; (b) the RP model.

Note that there is a tradeoff between κ, H and α_m in terms of coverage and performance. For example, at lower altitudes, higher values of α_m decrease fairness given the smaller coverage of the FBSs; then, deployment optimization can increase the minimum SINR by about 25 and 30 dB in the MMSE and MRC cases, respectively. In contrast, deployments at higher altitudes with wide beams, i.e., small α_m, result in fairer initial conditions and therefore the optimization gain is smaller, respectively around 15 and 10 dB. Similar conclusions apply when the variation is in κ and H. Altogether, the gain is in the range of 15-25 dB with MMSE and 15-30 dB with MRC reception.

The same trends can be observed for random LoS phases in Figs. 2.7 and 2.8 while Figs. 2.9 and 2.10 include results showing the increase in sum spectral efficiency. Given the space limitations, only the DP model is presented, although similar results are achieved for the RP model. For MMSE, the increase in sum spectral efficiency is between 30% and 45% while for MRC the range of improvement varies between 15% and 40%.

Fig. 2.11 examines whether augmenting the optimization to include also the altitude can further improve the deployments. Then, twelve additional locations, same six 2D spots with



(a) (b)

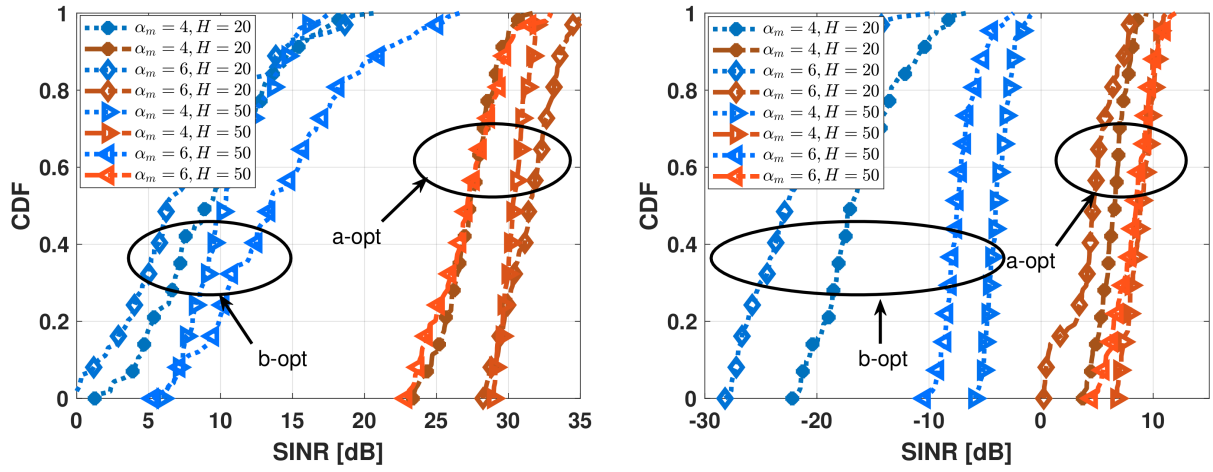
Figure 2.5: $\mathbb{E}\{\text{SINR}_k\}$ b-opt and a-opt for different κ and H under the DP model in (a) FC MMSE; (b) FC MRC.

a higher and lower heights, are explored at each iteration, with the altitude being constrained to $20 \leq H_m \leq 80$ m and the same D_{GS} applied vertically. Given the space limitations, only results for the RP MMSE case are presented, over different values of κ. The improvement in terms of minimum SINR remains in the range of 15 to 25 dB, hence adjusting the altitude does not seem to provide substantial additional gains once the horizontal position is already being optimized.

Finally, in Sec. 2.5 it is shown that the number of complex multiplications at each iteration is $\mathcal{O}\big(|\mathcal{F}_{k_{\min}^{(t)}}|K\big)$. In Fig. 2.12, the number of complex multiplications is plotted after averaging over the number of iterations varying K and for different M under MMSE reception. Note that M, and therefore $|\mathcal{F}_k|$, controls the slope while the growth is linear in K.

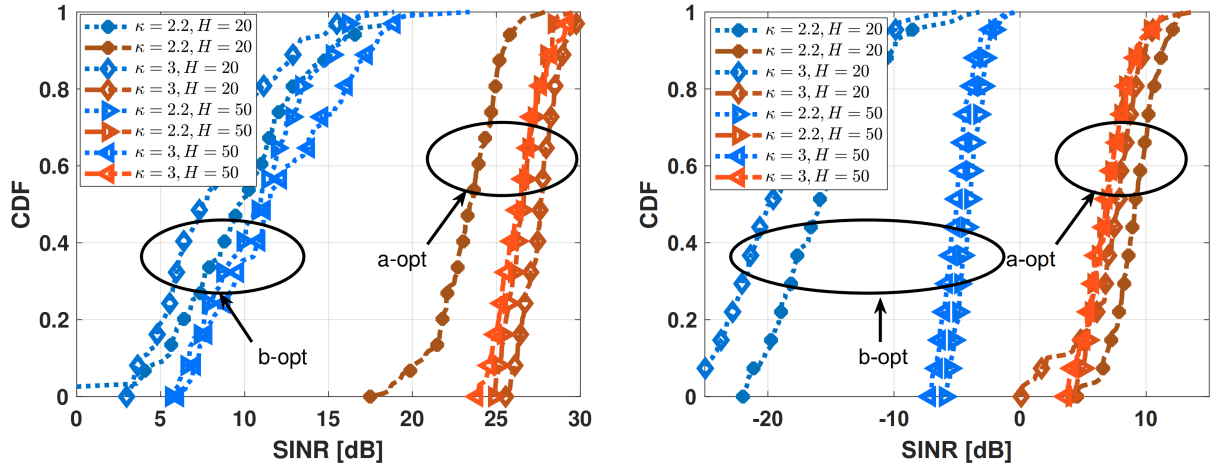
2.6.2 PC Networks

Prop. 4 and Cor. 2 are validated in Figs. 2.13a and 2.13b, respectively, with the same loads used for FC MRC. The deterministic equivalents are found to be representative even



(a) (b)

Figure 2.6: $\mathbb{E}\{\text{SINR}_k\}$ b-opt and a-opt for different α_m and H under the DP model in (a) FC MMSE; (b) FC MRC.



(a) (b)

Figure 2.7: $\mathbb{E}\{\text{SINR}_k\}$ b-opt and a-opt for different κ and H under the RP model in (a) FC MMSE; (b) FC MRC.

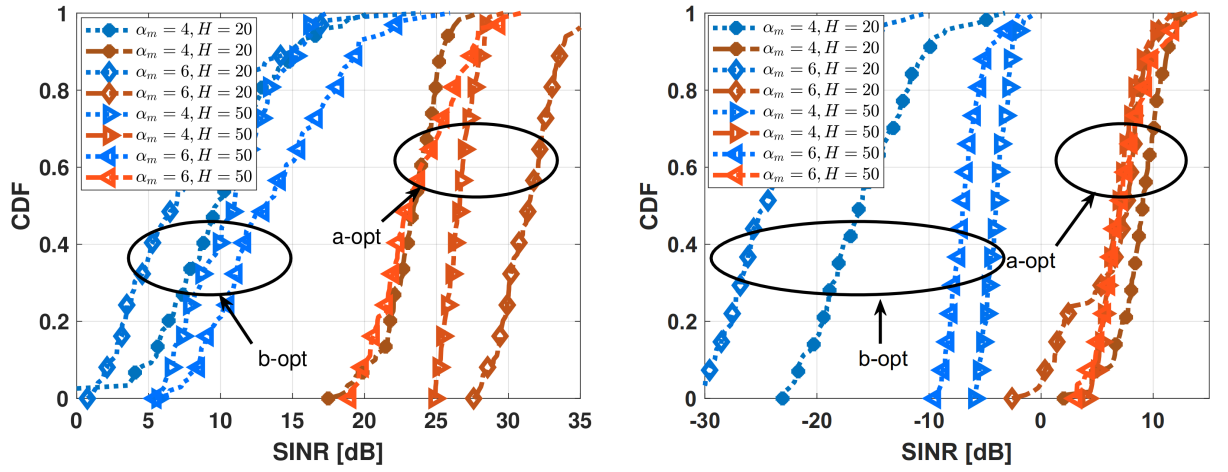


Figure 2.8: $\mathbb{E}\{\text{SINR}_k\}$ b-opt and a-opt for different α_m and H under the RP model in (a) FC MMSE; (b) FC MRC.

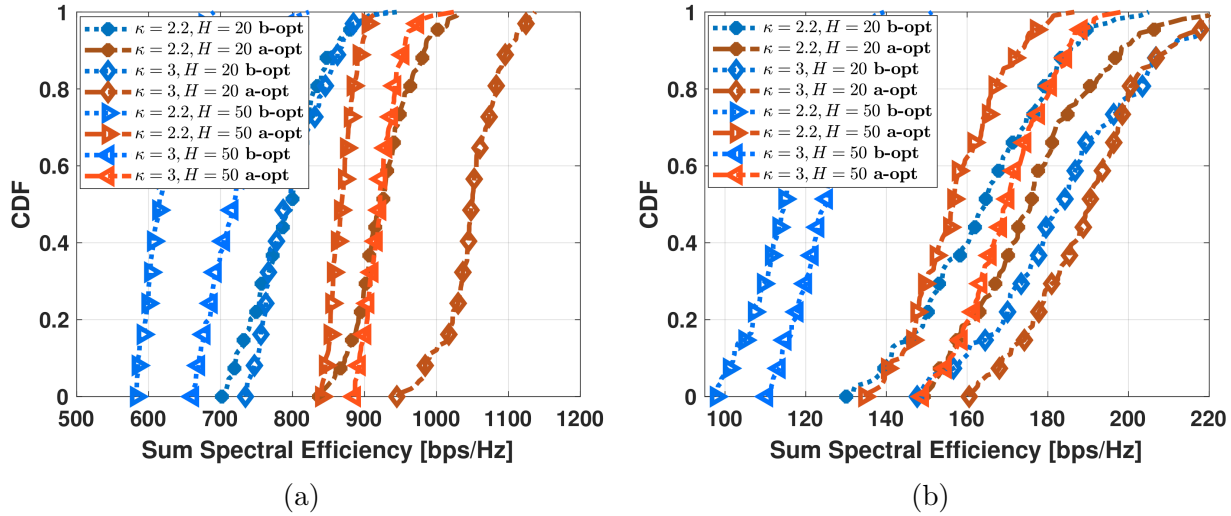
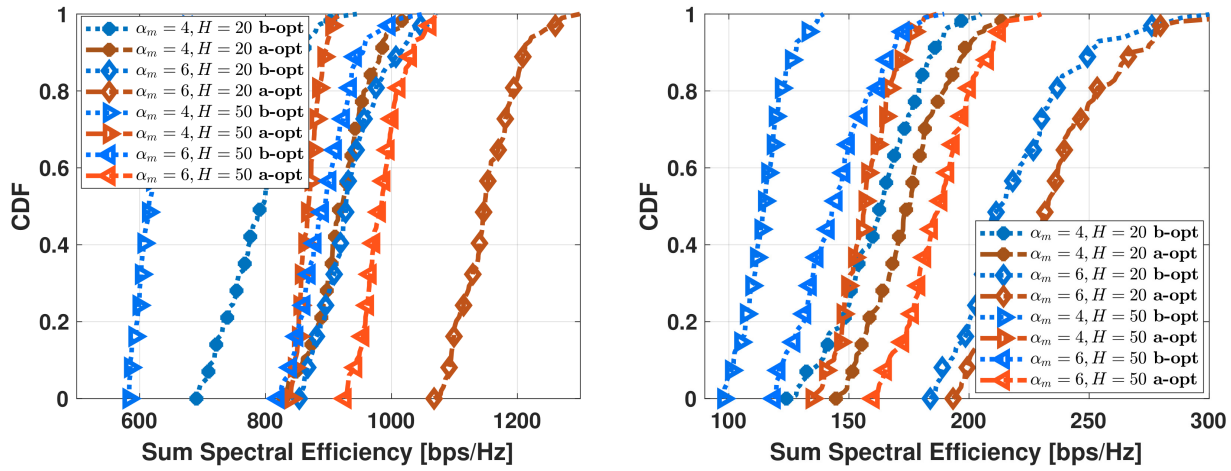


Figure 2.9: Sum spectral efficiency b-opt and a-opt for different κ and H under the DP model in (a) FC MMSE; (b) FC MRC.



(a) (b)

Figure 2.10: Sum spectral efficiency b-opt and a-opt for different α_m and H under the DP model in (a) FC MMSE; (b) FC MRC.

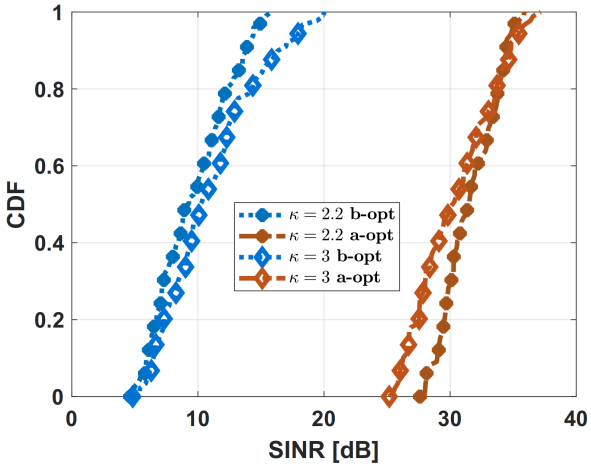


Figure 2.11: $\mathbb{E}\{\text{SINR}_k\}$ b-opt and a-opt for different α_m optimizing altitude.

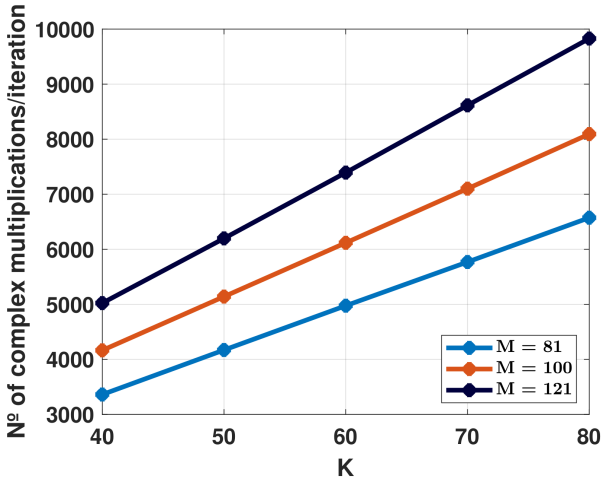


Figure 2.12: Average number of complex multiplications per iteration for different M.

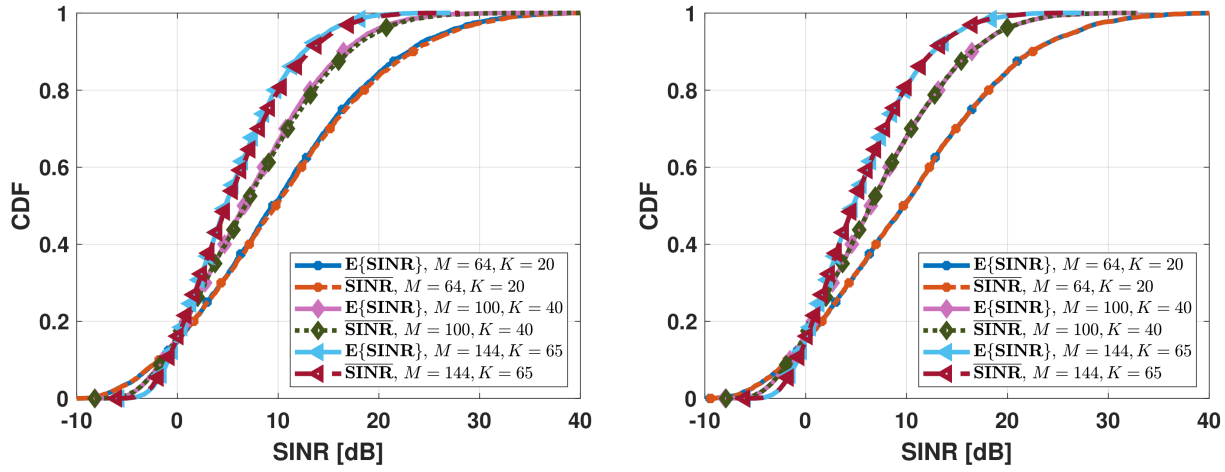
for small networks.

Figs. 2.14a and 2.14b depict deployment results for different values of κ, H and α_m under the DP model. The minimum SINR improvement is 15-30 dB in both figures. Similar results are provided for the RP LoS phase model in Figs. 2.15a and 2.15b. Fig. 2.16 shows the increase in sum spectral efficiency for the DP model, which ranges between 15% and 40%.

2.7 Conclusions

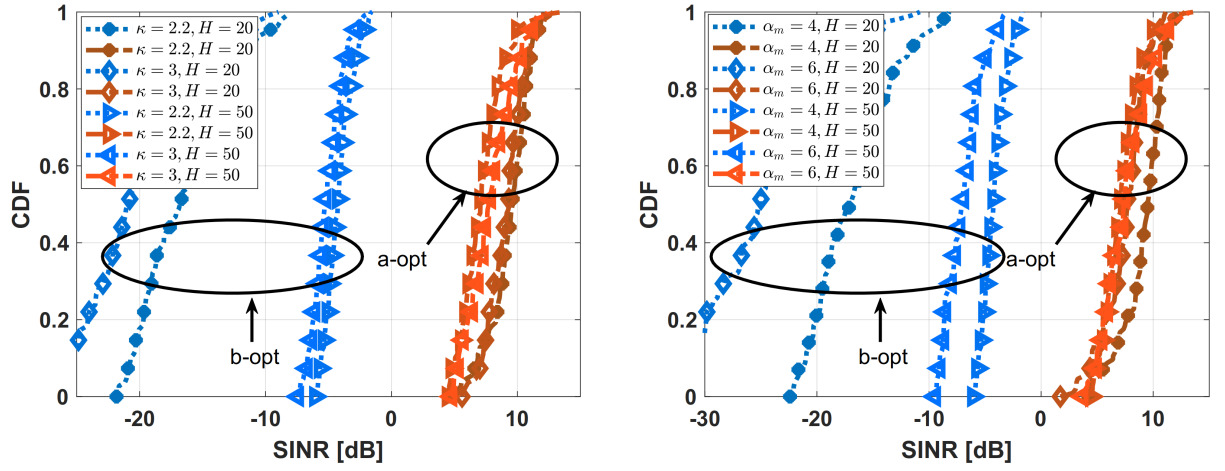
This chapter has considered fully and partially centralized architectures for the uplink of aerial cell-free networks. Two different Rician channel models have been considered: (i) with known phase and (ii) with random phase in the LoS component. For each model, deterministic equivalents have been derived for the SINR and the corresponding spectral efficiency.

Capitalizing on these deterministic equivalents, a deployment optimization problem has been proposed for each of the architectures. The analytical gradients have been obtained and, given the lack of convexity, a combined GB-GS approach has been followed. The resulting



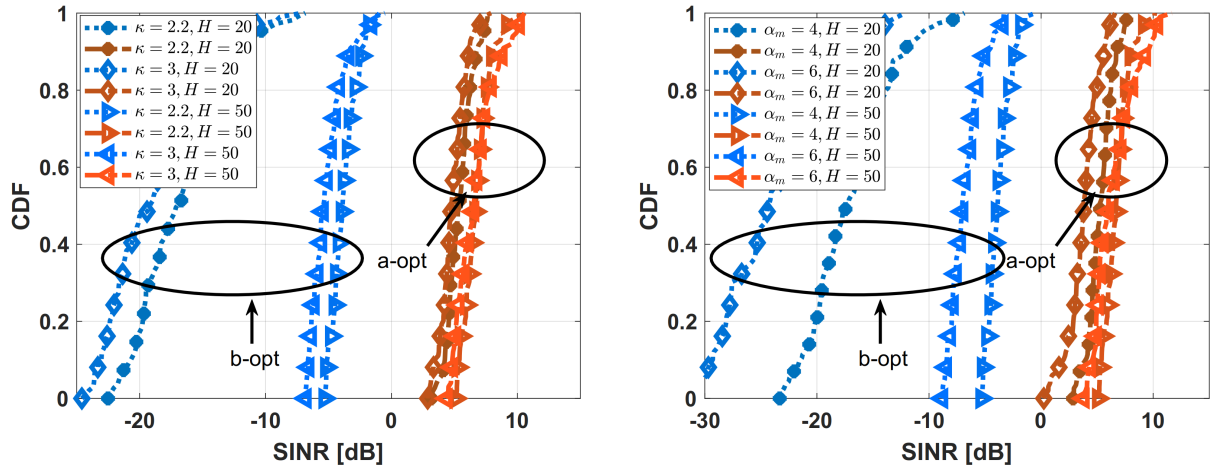
(a) (b)

Figure 2.13: $\mathbb{E}\{\text{SINR}_k\}$ vs $\overline{\text{SINR}}_k$ for PC MRC reception at $H = 50$ m under (a) the DP model; (b) the RP model.



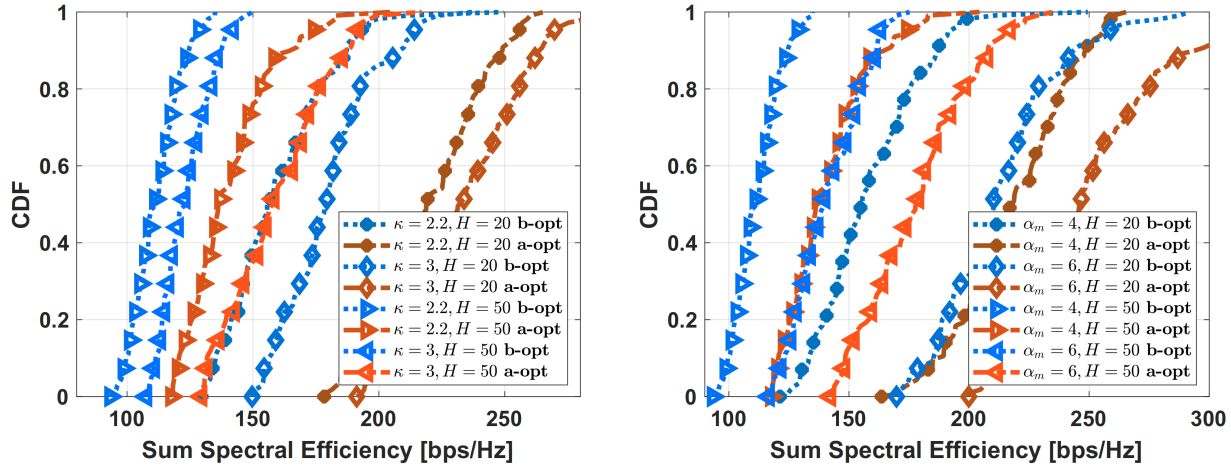
(a) (b)

Figure 2.14: $\mathbb{E}\{\text{SINR}_k\}$ b-opt and a-opt for PC MRC under the DP model for (a) different κ and H; (b) different α_m and H.



(a) (b)

Figure 2.15: $\mathbb{E}\{\text{SINR}_k\}$ b-opt and a-opt for PC MRC under the RP model for (a) different κ and H; (b) different α_m and H.



(a) (b)

Figure 2.16: Sum spectral efficiency b-opt and a-opt for (a) different κ and H; (b) different α_m and H.

improvements in minimum SINR and in sum spectral efficiency have been thoroughly quantified. Most remarkably, 15-25 and 15-30 dB increases in minimum SINR are attained with MMSE and MRC reception, respectively, for both fully and partially centralized networks.

Avenues for follow-up work include incorporating power control [94], accounting for the residual effects of pilot contamination, considering multiantenna transceivers, or studying the impact of the user location distributions and how such distributions could be estimated dynamically.

Chapter 3

Cell-Free UAV Networks with Wireless Fronthaul: Analysis and Optimization

3.1 Introduction

The race towards 6G wireless networks has begun and many ideas are under investigation [95], with UAVs as a potential game changer. Indeed, the inclusion of UAVs in wireless networks, and in particular their deployment as flying APs in cellular-based systems, is a research problem of growing interest [14–23, 25, 27]. Such flying APs are an attractive alternative to their terrestrial counterparts in terms of coverage, cost, and deployment flexibility. In particular, their flexibility makes flying APs enticing whenever the fixed infrastructure is disrupted. With respect to terrestrial APs, UAVs serving as flying APs pose two distinct challenges: (*i*) the ground-to-air and air-to-ground character of the radio access links (uplink and downlink, respectively) and (*ii*) the necessarily wireless nature of the fronthaul connect-

ing the UAVs to the rest of the network. The bulk of the research on this topic has thus far been on the former challenge, including UAV deployment, trajectory optimization, power control, or interference management [61–66, 72, 96], always assuming an ideal fronthaul.

Concurrently with the integration of UAVs, wireless systems are progressing towards software-defined architectures [97–99] under the umbrella of centralized radio access networks (C-RANs). This goes hand in hand with transcending the time-honored cellular paradigm and moving to cell-free network structures. In such networks, each user can potentially communicate with multiple APs by joint processing of the signals at the APs [11,12,31–33,35–43,100]. A cell-free framework is especially suitable for UAV networks since UAVs can create strong interference to adjacent cells because of the LoS nature of their channels. In a cell-free network, not only can this potential interference be mitigated, but it can actually be turned into useful signals. Initial results confirm the efficacy and benefits of organizing networks where UAVs serve as APs in a cell-free fashion [9, 10]. Again, these early results focus on the radio access, under the premise of ideal fronthauling.

The present chapter broadens the scope to encompass both the radio access and the wireless fronthaul, in recognition that an isolated study of one aspect may be deceiving because of potential bottlenecks in the other. With this broader view, UAVs go from being ideal conduits to having to face a constrained wireless fronthaul. This, in turn, brings to the fore issues such as the multiple access in that fronthaul, with alternatives that range from simple frequency-division multiple access (FDMA) to more sophisticated space-division multiple access (SDMA). While this work remains application-agnostic, the performance of the different fronthaul alternatives, and combinations thereof, are tackled. Particularly, the simplicity of FDMA, where the signal isolation reduces the interference, comes at the expense of a higher demand for bandwidth and therefore a reduction in the spectral efficiency. Alternatively, in SDMA, co-channel interference is the price of a multiplexing gain that enables parallelizing transmissions, thereby increasing the spectral efficiency. Finally, the hybrid FDMA-SDMA

fronthaul alternative provides more flexibility and can combine the best of both methods. For the sake of specificity, the chapter concentrates on the radio access uplink, with the equally important downlink relegated to future work.

While, motivated by massive MIMO principles, much of the cell-free literature considers matched-filter reception for the radio access uplink [32,33,35], the present work posits MMSE reception [35–37], which is optimum from a SINR perspective and reverts to matched filtering in some limiting regimes. This endows the results with broader generality.

With MMSE reception on the radio access and various alternatives for the wireless fronthaul, the analysis then takes place in the asymptotic regime in which the number of UAVs, users, and antennas at the C-RAN gateway, all grow large. This enables leveraging random matrix theory results [79–83] to derive deterministic equivalents (finite-dimensional approximations that become exact asymptotically) to the spectral efficiency; importantly, the analysis allows to flexibly define finite subsets of users being served by each UAV, and vice versa, whereby the signal processing complexity remains bounded even as the aforementioned quantities are scaled up. While the aforementioned references study the large-dimensional regime of one-hop cellular networks, to the best of our knowledge this is the first UAV work that provides an asymptotic analysis for two-hop networks, either cellular or cell-free. Two-hop channels are much more difficult to deal with as their overall distribution may not have a closed-form, and in fact the information-theoretic capacity of a multi-hop channel is not yet known. Algorithms that handle point-to-point two-hop settings have been proposed [101]; however, there are still many open problems in a multi-hop network setup. The addition of a wireless fronthaul therefore poses new challenges to UAV networks, especially under Rician fading, where new asymptotic results under zero-forcing reception are derived that might be of independent interest.

Armed with the deterministic equivalents for the spectral efficiency, three key problems are addressed, namely the optimization of (i) the UAV deployment, including altitude, (ii)

the user transmit powers, and (*iii*) the UAV transmit powers. These problems are studied separately given their lack of convexity and, for the deployment problem specifically, a combination of gradient-based (GB) and Gibbs sampling (GS) methods is invoked [84]. The joint optimization of the UAV deployment and the user and UAV transmit powers drastically improves the spectral efficiency, with the lion's share of the benefits being associated with the deployment given that larger feasible sets, i.e., the 3D space, can be explored compared to traditional performance optimization schemes, such as power or rate control. Altogether, the main contributions of the chapter can be summarized as follows:

- An analytical framework is set forth for the uplink of a cell-free UAV network with Rician fading, channel estimation, realistic antenna patterns, and MMSE reception on the radio access, as well as a wireless fronthaul.
- Deterministic equivalents are derived for the spectral efficiency in the above framework, under various fronthaul alternatives.
- For each of the fronthaul alternatives, and with the maximization of the minimum spectral efficiency as objective, the UAV deployment and the user and UAV transmit power problems are confronted.
- The impact on the optimization gains of network parameters such as the pathloss exponent or the antenna directivity is established.

The remainder of the chapter is organized as follows. Sec. 3.2 lays down the system and communication models. In Sec. 3.3, the transmission schemes are unveiled, including the cell-free aspects and the different fronthaul alternatives. Then, in Secs. 3.4–3.6, these alternatives are successively studied. Sec. 3.7 subsequently focuses on the deployment optimization problem while numerical results are presented and discussed in Sec. 3.8. Concluding remarks are provided in Sec. 3.9.

3.2 System Model

Consider the uplink of a cell-free network featuring M UAVs, located at $\boldsymbol{q}_m = (x_m, y_m)$ and altitude H_m, serving K cochannel single-antenna users at $\mathbf{w}_k = (\mathrm{x}_k, \mathrm{y}_k)$. The channel coefficient between user k and the single-antenna UAV m is denoted by $g_{k,m}$, drawn from a Rician distribution such that [86, Sec. 3.4.1]

$$g_{k,m} = \sqrt{\frac{\beta_0\, g_m(d_{k,m})}{d_{k,m}^{\kappa}} \frac{1}{K_{k,m}+1}} \left[\sqrt{K_{k,m}} e^{j\psi_{k,m}} + a_{k,m} \right], \tag{3.1}$$

where β_0 and κ are the pathloss at a reference distance of 1 m and the pathloss exponent, respectively, while $d_{k,m}$ is the distance. The Rician factor is $K_{k,m} = A_1 e^{A_2 \arcsin(\frac{H_m}{d_{k,m}})}$ for environment-dependent parameters A_1 and A_2 [73]. The phase of the LoS component, $\psi_{k,m}$, is uniformly random to reflect drifting [41–43] whereas $a_{k,m} \sim \mathcal{N}_{\mathbb{C}}(0,1)$ to account for the small-scale fading. Finally, [78]

$$g_m(d_{k,m}) = 2\,(\alpha_m + 1)\, \frac{H_m^{\alpha_m}}{d_{k,m}^{\alpha_m}} \tag{3.2}$$

models the antenna gain at the mth UAV, with α_m regulating the trade-off between coverage and directivity.[1] Hence, the channel correlation coefficient is

$$r_{k,m} = \mathbb{E}\{|g_{k,m}|^2\} \tag{3.3}$$

$$= 2\,(\alpha_m + 1)\, \beta_0\, \frac{H_m^{\alpha_m}}{d_{k,m}^{\alpha_m+\kappa}}. \tag{3.4}$$

Upon reception by the UAVs, the collected data is forwarded to the C-RAN gateway, whose coordinates are $\boldsymbol{q} = (x, y)$ with altitude H. Given its air-to-ground nature, a Rician model is invoked for the fronthaul as well. The channel vector connecting the mth UAV with the

[1] If multiantenna UAVs were considered, the generalization would be straightforward for IID fading while a spatial correlation matrix would have to be incorporated otherwise.

N-antenna C-RAN gateway is

$$\boldsymbol{h}_m = \sqrt{\frac{\beta_0}{d_m^\kappa}}\left[\sqrt{\frac{K_m}{K_m+1}}e^{j\psi_m}\boldsymbol{s}_m + \sqrt{\frac{1}{K_m+1}}\boldsymbol{a}_m\right] \tag{3.5}$$

where d_m and K_m are the distance and Rician factor between UAV m and the C-RAN, respectively. Additionally, ψ_m accounts for the drifting, again modelled as uniformly random. Moreover, $\boldsymbol{s}_m \in \mathbb{C}^{N\times 1}$ is the array response to the mth UAV. For an N-antenna uniform linear array (ULA), the array response satisfies

$$[\boldsymbol{s}_m]_n = e^{j\frac{2\pi f_c}{c}d(n-1)\sin(\theta_m)\cos(\phi_m)} \tag{3.6}$$

given the azimuth θ_m, elevation ϕ_m, and antenna spacing d. The small scale fading is $\boldsymbol{a}_m \sim \mathcal{N}_{\mathbb{C}}(\boldsymbol{0}, \boldsymbol{\Omega_m})$ for some spatial correlation matrix $\boldsymbol{\Omega_m}$ among the gateway antennas. Therefore, the overall covariance matrix for a given fronthaul link is

$$\boldsymbol{R}_m = \mathbb{E}\{\boldsymbol{h}_m\boldsymbol{h}_m^*\} \tag{3.7}$$

$$= \frac{\beta_0}{(K_m+1)d_m^\kappa}\left[K_m\boldsymbol{s}_m\boldsymbol{s}_m^* + \boldsymbol{\Omega_m}\right]. \tag{3.8}$$

A toy example of this two-hop structure containing wireless access and fronthaul is depicted in Fig. 1. While the access links are cell-free-based, the fronthaul allows for FDMA, SDMA or the combination FDMA-SDMA.

3.2.1 Channel Acquisition

The number of orthogonal pilot dimensions, denoted by τ, is constrained by the coherence bandwidth B_c and the coherence time T_c. The latter depends on the maximum UAV velocity, $v_{\max}$, and the carrier frequency, f_c, with the worst-case dependence being $T_c = \frac{c/f_c}{2v_{\max}}$ for isotropic scattering [86, Sec. 3.4]. The number of resource units within a fading block

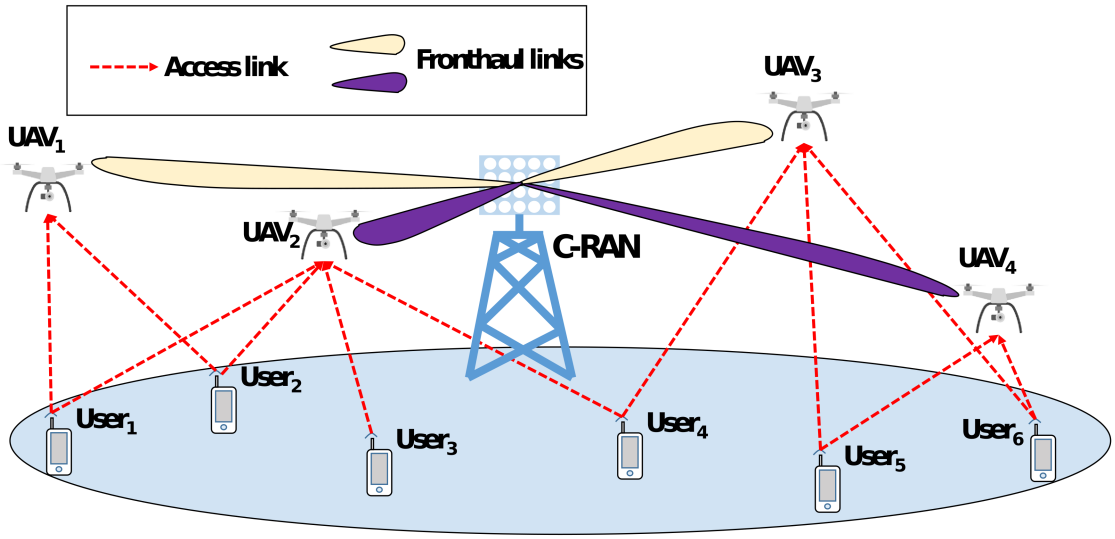


Figure 3.1: Cell-free UAV network with wireless access and fronthaul links. In this example, the multiple access employed in the fronthaul is FDMA-SDMA (see Sec. 3.6).

is $\tau_{\mathrm{c}} \approx T_{\mathrm{c}} B_{\mathrm{c}}$, typically a large number that enables τ to be itself large enough for pilot contamination to be negligible [40, 102]; it also allows for the use of techniques such as random pilots [103]. For instance, at $f_{\mathrm{c}} = 2.4$ GHz, and with conservative values $v_{\max} = 10$ m/s and $B_{\mathrm{c}} = 1$ MHz, we have $\tau_{\mathrm{c}} = 6250$. Upon observation of the pilot transmitted by user k at the mth UAV, the MMSE channel estimate $\hat{g}_{k,m}$ satisfies $g_{k,m} = \hat{g}_{k,m} + \tilde{g}_{k,m}$, where $\hat{g}_{k,m}$ is zero-mean with [88]

$$\gamma_{k,m} = \mathbb{E}\{|\hat{g}_{k,m}|^2\} \tag{3.9}$$

$$= \frac{r_{k,m}^2}{r_{k,m} + \frac{\sigma^2}{p^{\mathrm{t}}\tau}}, \tag{3.10}$$

for given τ and p^{t}, the latter denoting the pilot transmit power, while σ^2 is the noise power at the receiver. In addition, $\tilde{g}_{k,m}$ is zero-mean with variance $c_{k,m} = r_{k,m} - \gamma_{k,m}$. A similar approach is applied to the fronthaul, operating at a frequency different from those of the

access links, such that pilot contamination between the two stages is avoided. Concretely, the channel estimates between the mth UAV and the C-RAN gateway satisfy $\boldsymbol{h}_m = \hat{\boldsymbol{h}}_m + \tilde{\boldsymbol{h}}_m$ where $\hat{\boldsymbol{h}}_m$ is zero-mean with covariance

$$\boldsymbol{\Phi}_m = \mathbb{E}\{\hat{\boldsymbol{h}}_m \hat{\boldsymbol{h}}_m^*\} \tag{3.11}$$

$$= \boldsymbol{R}_m \boldsymbol{\Psi}_m^{-1} \boldsymbol{R}_m, \tag{3.12}$$

for $\boldsymbol{\Psi}_m = \boldsymbol{R}_m + \frac{\sigma^2}{p^t \tau} \boldsymbol{I}$. The error, $\tilde{\boldsymbol{h}}_m$, is zero-mean with covariance $\boldsymbol{C}_m = \boldsymbol{R}_m - \boldsymbol{\Phi}_m$.

3.3 Data Transmission Schemes

This section describes the two-stage data transmission, namely the cell-free radio access and the wireless fronthaul. For the latter, several alternatives are entertained: FDMA, SDMA, and combinations thereof.

3.3.1 Cell-Free Radio Access

On a given time-frequency resource unit, the uplink channel matrix is

$$\boldsymbol{G} = \left(\boldsymbol{g}_1, \ldots, \boldsymbol{g}_K \right), \tag{3.13}$$

where $\boldsymbol{g}_k \in \mathbb{C}^{M \times 1}$ is the channel between user k and all UAVs, satisfying $\boldsymbol{G} = \hat{\boldsymbol{G}} + \tilde{\boldsymbol{G}}$, with $\hat{\boldsymbol{G}}$ and $\tilde{\boldsymbol{G}}$ being the channel estimation and error matrices, respectively. The subset of UAVs participating in the reception of each user is determined by the binary matrix

$\boldsymbol{M}^{(s)} = (\boldsymbol{m}_1^{(\mathrm{s})}, \ldots, \boldsymbol{m}_K^{(\mathrm{s})}) \in \mathbb{Z}_2^{M \times K}$ with entries

$$\left[\boldsymbol{M}^{(s)}\right]_{m,k} = \begin{cases} 1 & \text{if } k \in \mathcal{U}_m \\ 0 & \text{otherwise} \end{cases},$$

where $\mathcal{U}_m$ is the set of users regarded as signal by the mth UAV. Its complementary matrix is $\boldsymbol{M}^{(i)} = \mathbf{1} - \boldsymbol{M}^{(s)}$, with nonzero entries indicating the users that each UAV regards as noise. Pooling the observations from the M UAVs,

$$\boldsymbol{y} = \boldsymbol{M}^{(s)} \circ \boldsymbol{G}\boldsymbol{x} + \boldsymbol{M}^{(i)} \circ \boldsymbol{G}\boldsymbol{x} + \boldsymbol{n} \tag{3.14}$$

$$= \boldsymbol{M}^{(s)} \circ \hat{\boldsymbol{G}}\boldsymbol{x} + \left(\boldsymbol{M}^{(s)} \circ \tilde{\boldsymbol{G}} + \boldsymbol{M}^{(i)} \circ \boldsymbol{G}\right)\boldsymbol{x} + \boldsymbol{n}, \tag{3.15}$$

where $\circ$ denotes Hadamard product, $\boldsymbol{y} = (y_1, \ldots, y_M)^{\mathrm{T}}$, $\boldsymbol{x} = (\sqrt{p_1}s_1, \ldots, \sqrt{p_K}s_K)^{\mathrm{T}}$ with symbols s_k having unit power, p_k being the transmit power, and $\boldsymbol{n} \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \sigma^2\boldsymbol{I})$.

3.3.2 Wireless Fronthaul

At the fronthaul stage, the mth UAV transmits $t_m = \sqrt{\rho_m}y_m$, where

$$\rho_m = \frac{\mathrm{p}_m}{\mathbb{E}\{|y_m|^2\}} \tag{3.16}$$

$$= \frac{\mathrm{p}_m}{\sum_{k=1}^{K} r_{k,m}p_k + \sigma^2} \tag{3.17}$$

ensures an average transmit power of p_m. Within this general framework, the various fronthaul alternatives can be modeled.

FDMA

The bandwidth availability at mmWave and sub-THz frequencies makes FDMA an enticing solution, in which signals are perfectly isolated. Here, single-antenna reception at the C-RAN gateway suffices—this is a special case of the FDMA-SDMA strategy with $N = 1$ receive antennas presented later in this section. As a consequence, the observed signal at the C-RAN gateway over the band allocated to the mth UAV is then

$$z_m = h_m t_m + n_m, \tag{3.18}$$

where $n_m \sim \mathcal{N}_{\mathbb{C}}(0, \sigma^2)$.

SDMA

Systems suffering from bandwidth limitations for the fronthaul might consider SDMA, where UAVs transmit concurrently. Their signals are untangled at the C-RAN gateway by the fronthaul combiner $\boldsymbol{u}_m \in \mathbb{C}^{N\times 1}$, with $N \geq M$. At that combiner's output, the signal corresponding to the mth UAV is

$$z_m = \boldsymbol{u}_m^* \left(\sum_{j=1}^{M} \boldsymbol{h}_j t_j + \boldsymbol{n} \right), \tag{3.19}$$

with $\boldsymbol{n} \sim \mathcal{N}_{\mathbb{C}}(\boldsymbol{0}, \sigma^2 \boldsymbol{I})$. The structure of $\boldsymbol{u}_m$ is discussed in the next section.

FDMA-SDMA

FDMA and SDMA can be combined. Let the system have $1 \leq L \leq M$ frequency bands, with $L = M$ being FDMA and $L = 1$ being pure SDMA. Over band f_ℓ, a subset of UAVs, denoted by $\mathcal{M}_\ell$, conveys data to the C-RAN, which separates the $|\mathcal{M}_\ell|$ streams through an

N-dimensional fronthaul combiner, $\boldsymbol{u}_m \in \mathbb{C}^{N\times 1}$. For $L > 1$, the observed signal for $m \in \mathcal{M}_\ell$ is

$$z_m = \boldsymbol{u}_m^* \Bigg(\sum_{j\in\mathcal{M}_\ell} \boldsymbol{h}_j t_j + \boldsymbol{n} \Bigg), \tag{3.20}$$

whose terms respectively correspond to the signals from the $|\mathcal{M}_\ell|$ UAVs sharing the ℓth bands and noise.

3.4 FDMA Fronthaul

Let us now proceed to analyze the performance under FDMA fronthauling. After collecting the M fronthaul transmissions over different bands, the C-RAN receives

$$\boldsymbol{z} = \boldsymbol{c} \circ \boldsymbol{M}^{(\boldsymbol{s})} \circ \hat{\boldsymbol{G}}\boldsymbol{x} + \boldsymbol{n}'. \tag{3.21}$$

With the C-RAN treating the fronthaul channel estimate as the true channel, the effective fronthaul gain for the mth UAV is $c_m = \hat{h}_m\sqrt{\rho_m}$; the gains for the M UAVs are assembled into $\boldsymbol{c} = (c_1, \ldots, c_M)$. In turn, $\boldsymbol{n}'$ is the effective noise, zero-mean and with covariance $\boldsymbol{\Sigma} = \mathbb{E}\{\boldsymbol{n}'\boldsymbol{n}'^*\}$. It can be verified that $\boldsymbol{\Sigma}$ is diagonal, with entries

$$[\boldsymbol{\Sigma}]_{m,m} = r_m\rho_m \left(\sum_{\forall i} r_{i,m}p_i + \sigma^2 \right) - \phi_m\rho_m \sum_{i\in\mathcal{U}_m} \gamma_{i,m}p_i + \sigma^2. \tag{3.22}$$

Let $\mathcal{F}_k = \left\{ m : \left[\boldsymbol{M}^{(\boldsymbol{s})}\right]_{m,k} = 1 \ , \ m = 1, \ldots, M \right\}$ be the subset of UAVs that regard what is received from user k as signal. From the rows of $\boldsymbol{z}$ whose indices are in $\mathcal{F}_k$, we obtain the

$$\mathrm{SINR}_k = \frac{|\boldsymbol{w}_k^*(\boldsymbol{c}_k \circ \hat{\boldsymbol{g}}_k)|^2 p_k}{\boldsymbol{w}_k^* \left(\sum_{i \neq k} (\boldsymbol{c}_k \circ \boldsymbol{m}_{k,i}^{(s)} \circ \hat{\boldsymbol{g}}_i)(\boldsymbol{c}_k \circ \boldsymbol{m}_{k,i}^{(s)} \circ \hat{\boldsymbol{g}}_i)^* p_i + \boldsymbol{\Sigma}_k \right) \boldsymbol{w}_k^*}, \tag{3.24}$$

$$\mathrm{SINR}_k = (\boldsymbol{c}_k \circ \hat{\boldsymbol{g}}_k)^* \left(\sum_{i \neq k} (\boldsymbol{c}_k \circ \boldsymbol{m}_{k,i}^{(s)} \circ \hat{\boldsymbol{g}}_i)(\boldsymbol{c}_k \circ \boldsymbol{m}_{k,i}^{(s)} \circ \hat{\boldsymbol{g}}_i)^* p_i + \boldsymbol{\Sigma}_k \right)^{-1} (\boldsymbol{c}_k \circ \hat{\boldsymbol{g}}_k)\, p_k. \tag{3.26}$$

$|\mathcal{F}_k| \times 1$ vector

$$\boldsymbol{z}_k = \boldsymbol{c}_k \circ \boldsymbol{M}_k^{(s)} \circ \hat{\boldsymbol{G}}_k \boldsymbol{x} + \boldsymbol{n}'_k, \tag{3.23}$$

where $\boldsymbol{M}_k^{(s)} = (\boldsymbol{m}_{k,1}^{(s)}, \ldots, \boldsymbol{m}_{k,K}^{(s)}) \in \mathbb{Z}_2^{|\mathcal{F}_k| \times K}$, $\boldsymbol{c}_k \in \mathbb{C}^{|\mathcal{F}_k| \times |\mathcal{F}_k|}$, $\hat{\boldsymbol{G}}_k \in \mathbb{C}^{|\mathcal{F}_k| \times K}$ and $\boldsymbol{n}'_k \in \mathbb{C}^{|\mathcal{F}_k| \times 1}$ contain the $\mathcal{F}_k$ rows of the original matrices and vectors. For a generic combiner, $\boldsymbol{w}_k \in \mathbb{C}^{|\mathcal{F}_k| \times 1}$, the instantaneous SINR experienced by user k is given in Eq. (3.24) achieving a spectral efficiency of

$$\mathrm{SE}_k = \frac{1}{L} \left(1 - \frac{\tau}{\tau_c} \right) \mathbb{E}\{\log_2(1 + \mathrm{SINR}_k)\}, \tag{3.25}$$

where $\frac{\tau}{\tau_c}$ accounts for the pilot overhead and L represents the number of fronthaul frequency bands; in this case $L = M$. Consequently, although an increase in M yields higher SINR values, the pre-log factor dominates (3.25) and therefore the overall spectral efficiency decreases. With the optimum MMSE combiner, the above specializes to [36] the expression presented in Eq. (3.26).

$$\overline{\mathrm{SINR}}_{k,m} = \frac{\gamma_{k,m} p_k}{\sum\limits_{\substack{i \in \mathcal{U}_m \\ i \neq k}} \frac{\gamma_{i,m}}{1+e_i} p_i + \frac{r_m}{\phi_m}\big(\sum\limits_{\forall i} r_{i,m} p_i + \sigma^2\big) - \sum\limits_{i \in \mathcal{U}_m} \gamma_{i,m} p_i + \frac{\sigma^2}{\phi_m \rho_m}}. \tag{3.31}$$

3.4.1 Large-Dimensional Analysis

The evaluation of (3.25) takes place in the asymptotic regime, $|\mathcal{F}_k|, |\mathcal{U}_m| \to \infty \; \forall \, k, m$, where convergence to nonrandom limits is assured provided that

$$\boldsymbol{\Gamma}_k = \mathbb{E}\Big\{ \big(\boldsymbol{m}_k^{(\mathrm{s})} \circ \hat{\boldsymbol{g}}_k\big) \big(\boldsymbol{m}_k^{(\mathrm{s})} \circ \hat{\boldsymbol{g}}_k\big)^* \Big\} \tag{3.27}$$

$$= \mathrm{diag}\big\{ \gamma_{k,m} m_{k,m}^{(s)} \; \forall m \big\}, \tag{3.28}$$

and

$$\boldsymbol{\Phi} = \mathbb{E}\Big\{ \hat{\boldsymbol{h}} \hat{\boldsymbol{h}}^* \Big\} \tag{3.29}$$

$$= \mathrm{diag}\big\{ \phi_m \; \forall m \big\}, \tag{3.30}$$

with $\hat{\boldsymbol{h}} = (\hat{h}_1, \ldots, \hat{h}_M)^{\mathrm{T}}$ satisfying some technical conditions. Specifically, the inverse of the resolvent matrix in (3.26) must exist, which is ensured by the presence of $\boldsymbol{\Sigma}_k$, while $\boldsymbol{\Gamma}_k$ and $\boldsymbol{\Phi}$ must have uniformly bounded spectral norms. In other words, the received power should not concentrate on a subset of dimensions as the network grows large.

Theorem 1. *With an FDMA fronthaul, $|\mathcal{F}_k|, |\mathcal{U}_m| \to \infty \; \forall \, k, m$ and MMSE subset combining,* $\mathrm{SINR}_k - \overline{\mathrm{SINR}}_k \xrightarrow{\text{a.s.}} 0$ *almost surely (a.s.) with* $\overline{\mathrm{SINR}}_k = \sum\limits_{m \in \mathcal{F}_k} \overline{\mathrm{SINR}}_{k,m}$ *and* $\overline{\mathrm{SINR}}_{k,m}$ *given in Eq.* (3.31)*. The coefficients e_j are obtained iteratively by $e_j = \lim_{n \to \infty} e_j^{(n)}$, $e_j^{(0)} = |\mathcal{F}_j|$, and*

$$e_j^{(n)} = p_j \, \mathrm{tr}\left[\boldsymbol{\Phi} \boldsymbol{\Gamma}_j \Big(\sum_{i \neq j}^{K} \frac{\boldsymbol{\Phi} \boldsymbol{\Gamma}_i}{1 + e_i^{(n-1)}} \, p_i + \boldsymbol{\Sigma}_j \Big)^{-1} \right]. \tag{3.32}$$

Proof. Details on how (3.31) emanates from [82, 83] can be found in Appendix A.6. □

Interestingly, note that in the asymptotic regime, the value of $\overline{\mathrm{SINR}}_k$ is a linear combination of the SINRs that the kth user experiences over the $\mathcal{F}_k$ UAVs weighted by the fronthaul channel. Finally, from the continuous mapping theorem [89], $\mathrm{SE}_k - \frac{1}{M}\left(1 - \frac{\tau}{\tau_\mathrm{c}}\right)\log_2(1 + \overline{\mathrm{SINR}}_k) \overset{\text{a.s.}}{\to} 0$.

3.4.2 Problem Formulation

Let us now turn to optimizing the UAV deployment and transmit powers. With the aim of ensuring fairness in the network, this is formulated as the max-min problem

$$\begin{aligned} \max_{\boldsymbol{q}_m, H_m, p_k, \mathrm{p}_m} \ & \min_k \ \frac{1}{M}\left(1 - \frac{\tau}{\tau_\mathrm{c}}\right)\mathbb{E}\{\log_2(1 + \mathrm{SINR}_k)\} \\ \text{s.t.} \ & H_{\min} \le H_m \le H_{\max},\ p_k \le p_{\max},\ \mathrm{p}_m \le \mathrm{p}_{\max}, \end{aligned} \tag{3.33}$$

which is nonconvex. Invoking Thm. 1, and with the constraints not reiterated for the sake of compactness, the above leads to

$$\max_{\boldsymbol{q}_m, H_m, p_k, \mathrm{p}_m} \ \min_k \ \sum_{m \in \mathcal{F}_k} \overline{\mathrm{SINR}}_{k,m}, \tag{3.34}$$

where $\overline{\mathrm{SINR}}_{k,m}$ is provided in (3.31). The optimizations of UAV deployment and transmit powers are tacked separately as follows.

Deployment Optimization

The analytical 2D-gradients w.r.t. (3.34) for a given altitude are

$$\nabla_{\boldsymbol{q}_j} \overline{\mathrm{SINR}}_k = \frac{\nabla_{\boldsymbol{q}_j}\gamma_{k,j}\,\mathrm{Den}_j - \gamma_{k,j}\nabla_{\boldsymbol{q}_j}\mathrm{Den}_j}{\mathrm{Den}_j^2} p_k \ \text{ for } j \in \mathcal{F}_k, \tag{3.35}$$

where Den_j is the denominator of (3.31). The optimization of H_m is studied separately, as it is common to every fronthaul alternative.

User Transmit Power

The following result is a stepping stone to the user transmit power optimization.

Proposition 5. *The objective function* $\min_k \overline{\text{SINR}}_k$ *in* (3.33) *satisfies the definition of competitive utility function while the constraints* $p_k \leq p_{\max}$ *follow the definition of monotonic constraints.*

Proof. See Appendix A.7. □

Capitalizing on Prop. 5, the algorithm in [104, Alg. 1] can be applied with sure converge to the optimum user transmit power in the max-min SINR sense.

UAV Transmit Power

From (3.31), it can be shown that $\overline{\text{SINR}}_{k,m}$ is an increasing function of p_m. Consequently, $\overline{\text{SINR}}_k$ increases with p_m as well. Therefore, the optimal UAV transmit power that maximizes the $\min_k \overline{\text{SINR}}_k$ is $\text{p}_\text{m} = \text{p}_\text{max}$.

3.5 SDMA Fronthaul

Let us now turn to the SDMA fronthaul alternative. The C-RAN received signals still follow (3.21) after applying the N-dimensional combiner $\boldsymbol{u}_m$ in (3.19) and replacing $c_m =$

$\boldsymbol{u}_m^* \hat{\boldsymbol{h}}_m \sqrt{\rho_m}$ and the equivalent noise

$$n'_m = \sum_{j=1}^{M} \boldsymbol{u}_m^* \boldsymbol{h}_j \sqrt{\rho_j} y_j - \boldsymbol{u}_m^* \hat{\boldsymbol{h}}_m \sqrt{\rho_m} \Bigg(\sum_{k \in \mathcal{U}_m} \hat{g}_{k,m} x_k \Bigg) + \boldsymbol{u}_m^* \boldsymbol{n}. \tag{3.36}$$

The SINR and spectral efficiency expressions in (3.25)–(3.26), corresponding to an MMSE access combiner, also hold with the aforementioned modifications. In particular, the pre-log factor only depends on the pilot overhead when $L = 1$.

A zero-forcing (ZF) structure is adopted for the fronthaul, whereby $\boldsymbol{U} = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_M) \in \mathbb{C}^{N \times M}$ is given by $\boldsymbol{U} = \hat{\boldsymbol{H}} (\hat{\boldsymbol{H}}^* \hat{\boldsymbol{H}})^{-1}$ with $\hat{\boldsymbol{H}} = (\hat{\boldsymbol{h}}_1, \ldots, \hat{\boldsymbol{h}}_M)$. Then, $\boldsymbol{u}_m^* \hat{\boldsymbol{h}}_j = \delta_{m,j}$ with $\delta_{m,j} = 1$ if $m = j$ and 0 otherwise. The ensuing SINR involves the equivalent noise power $\mathbb{E}\{n'_m n'^*_j\}$ under Rician fading, for which no expressions are available in the literature. One of the contributions in the sequel is an asymptotic expression for this power.

3.5.1 Large-Dimensional Analysis

As in Sec. 4.3.1, the spectral efficiency is evaluated for $|\mathcal{F}_k|, |\mathcal{U}_m| \to \infty \; \forall \, k, m$ and $N \to \infty$ with $N \geq M$. Convergence to deterministic limits is assured provided that $\boldsymbol{R}_m$ satisfies the same conditions as $\boldsymbol{\Phi}$ and $\boldsymbol{\Gamma}_m$. As the equivalent noise n'_m in (3.36) satisfies $\mathbb{E}\{n'_m n'^*_j\} \propto \mathbb{E}\{\boldsymbol{u}_m^* \boldsymbol{Q} \boldsymbol{u}_j\}$, we first proceed to characterize such quadratic form asymptotically with a result that might be of independent interest.

Theorem 2. *Let $\boldsymbol{Q} \in \mathbb{C}^{N \times N}$ be a deterministic Hermitian matrix while $\boldsymbol{U} = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_M) \in \mathbb{C}^{N \times M}$ is a ZF matrix combiner, $\boldsymbol{U} = \lim_{\epsilon \to 0} \hat{\boldsymbol{H}} (\hat{\boldsymbol{H}}^* \hat{\boldsymbol{H}} + \epsilon \boldsymbol{I})^{-1}$. For M,$N \to \infty$,*

$$\mathbb{E}\{\boldsymbol{u}_m^* \boldsymbol{Q} \boldsymbol{u}_m\} - \lim_{\epsilon \to 0} \frac{\frac{1}{N^2} \mathrm{tr}\big(\boldsymbol{\Phi}_m \boldsymbol{T}'(\epsilon, \boldsymbol{Q})\big)}{\big(1 + \frac{1}{N} \mathrm{tr}(\boldsymbol{\Phi}_m \boldsymbol{T})\big)^2} \xrightarrow{\text{a.s.}} 0 \tag{3.37}$$

for $\boldsymbol{T}$ and $\boldsymbol{T}'(\epsilon, \boldsymbol{Q})$ defined in (A.1) *and* (A.3)*, respectively.*

$$\overline{\mathrm{SINR}}_{k,m} = \frac{\gamma_{k,m} p_k}{\sum\limits_{\substack{i \in \mathcal{U}_m \\ i \neq k}} \frac{\gamma_{i,m}}{1+e_i} p_i + \sum\limits_{\forall i} r_{i,m} p_i - \sum\limits_{i \in \mathcal{U}_m} \gamma_{i,m} p_i + \sigma^2 + \frac{\xi_m^{\mathrm{SDMA}}}{\rho_m}}. \tag{3.39}$$

Proof. See Appendix A.8. □

The convergence of (3.37), in terms of relative error, is illustrated in Fig. 3.2.

Corollary 3. *Let $\boldsymbol{Q} \in \mathbb{C}^{N \times N}$ be a deterministic Hermitian matrix while $\boldsymbol{U} = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_M) \in \mathbb{C}^{N \times M}$ is a ZF matrix combiner, $\boldsymbol{U} = \lim_{\epsilon \to 0} \hat{\boldsymbol{H}} (\hat{\boldsymbol{H}}^* \hat{\boldsymbol{H}} + \epsilon \boldsymbol{I})^{-1}$. For M,$N \to \infty$, and $m \neq j$*

$$\mathbb{E}\{\boldsymbol{u}_m^* \boldsymbol{Q} \boldsymbol{u}_j\} \overset{\text{a.s.}}{\to} 0. \tag{3.38}$$

Proof. The proof follows similar steps as the ones included in Appendix A.8 and exploits the fact that $\hat{\boldsymbol{h}}_m$ and $\hat{\boldsymbol{h}}_j$ are uncorrelated. □

The combination of Thm. 2 and Corollary 3 results in an asymptotically diagonal noise covariance matrix $\boldsymbol{\Sigma}_k$.

Theorem 3. *With an SDMA fronthaul, ZF fronthaul combining, $|\mathcal{F}_k|, |\mathcal{U}_m| \to \infty \; \forall \, k, m$, $N \to \infty$ with $N \geq M$ and MMSE subset combining, $\mathrm{SINR}_k - \overline{\mathrm{SINR}}_k \overset{\text{a.s.}}{\to} 0$ with $\overline{\mathrm{SINR}}_k = \sum\limits_{m \in \mathcal{F}_k} \overline{\mathrm{SINR}}_{k,m}$ and $\overline{\mathrm{SINR}}_{k,m}$ given in Eq. (3.39). The application of Thm. 2 to $\mathbb{E}\{n'_m n'^*_m\}$ results in*

$$\xi_m^{SDMA} = \lim_{\epsilon \to 0} \sum_{n=1}^{M} \mathrm{p}_n \frac{\frac{1}{N^2} \mathrm{tr}(\boldsymbol{\Phi}_m \boldsymbol{T}'(\epsilon, \boldsymbol{C}_n))}{\left(1 + \frac{1}{N} \mathrm{tr}(\boldsymbol{\Phi}_m \boldsymbol{T})\right)^2} + \sigma^2 \frac{\frac{1}{N^2} \mathrm{tr}(\boldsymbol{\Phi}_m \boldsymbol{T}'(\epsilon, \boldsymbol{I}))}{\left(1 + \frac{1}{N} \mathrm{tr}(\boldsymbol{\Phi}_m \boldsymbol{T})\right)^2}$$

while the coefficients e_j are obtained iteratively by $e_j = \lim_{n \to \infty} e_j^{(n)}$, $e_j^{(0)} = |\mathcal{F}_j|$, and $e_j^{(n)}$ included in Eq. (3.40).

Proof. Proceed as in Appendix A.6. □

$$e_j^{(n)} = \sum_{m\in\mathcal{F}_j} \frac{\gamma_{j,m}p_j}{\sum\limits_{\substack{i\in\mathcal{U}_m\\ i\neq k}} \frac{\gamma_{i,m}}{1+e_i^{(n-1)}}p_i + \sum\limits_{\forall i} r_{i,m}p_i - \sum\limits_{i\in\mathcal{U}_m} \gamma_{i,m}p_i + \sigma^2 + \frac{\xi_m^{\mathrm{SDMA}}}{\rho_m}}. \tag{3.40}$$

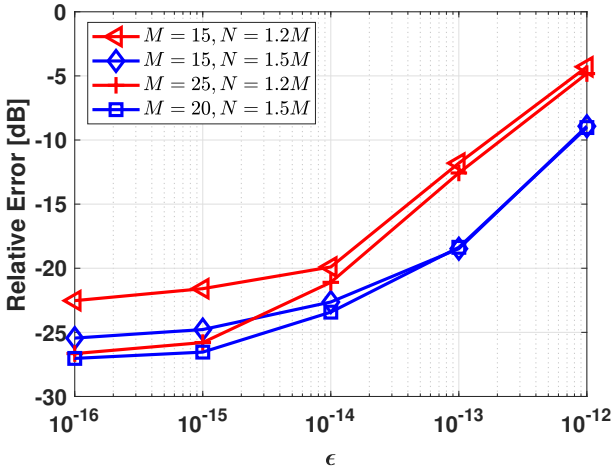


Figure 3.2: Relative error between the two terms in Thm. 2 as a function of ϵ for various M and N.

Similarly to the FDMA case, $\overline{\mathrm{SINR}}_k$ can be decomposed as the sum of SINRs over the $\mathcal{F}_k$ UAVs with two main differences: (i) the fronthaul channel is compensated by the ZF combiner and (ii) the noise is increased after the ZF stage, as per $\frac{\xi_m^{\mathrm{SDMA}}}{\rho_m}$. Finally, from the continuous mapping theorem, $\mathrm{SE}_k - \left(1 - \frac{\tau}{\tau_c}\right)\log_2(1+\overline{\mathrm{SINR}}_k) \overset{\text{a.s.}}{\to} 0$.

3.5.2 Problem Formulation

We now turn to optimizing the UAV deployment and transmit powers by maximizing the minimum SINR under SDMA fronthauling. Capitalizing on Thm. 3, that amounts to

$$\begin{aligned} &\max_{\boldsymbol{q}_m, H_m, p_k, \mathrm{p}_m} \min_k \sum_{m\in\mathcal{F}_k} \overline{\mathrm{SINR}}_{k,m} \\ &\quad \text{s.t.} \quad H_{\min} \le H_m \le H_{\max},\ p_k \le p_{\max},\ \mathrm{p}_m \le \mathrm{p}_{\max}, \end{aligned} \tag{3.41}$$

for $\overline{\mathrm{SINR}}_{k,m}$ in (3.39). The above problem is nonconvex.

Deployment Optimization

The presence of ξ_m^{SDMA} in the denominator of (3.39) makes the gradients analytically unwieldy. However, as shown in Fig. 3.3a, the signal terms within ξ_m^{SDMA} satisfy

$$\lim_{\epsilon \to 0} \frac{\frac{1}{N^2}\mathrm{tr}(\boldsymbol{\Phi}_m \boldsymbol{T}'(\epsilon, \boldsymbol{C}_n))}{\left(1 + \frac{1}{N}\mathrm{tr}(\boldsymbol{\Phi}_m \boldsymbol{T})\right)^2} \approx c_m d_m^{\kappa}, \tag{3.42}$$

where c_m is a regression parameter and d_m, recall, is the distance between UAV m and the C-RAN. Referring to Fig. 3.3a, c_m can be obtained by fitting every data point (solid regression curve) or only the maximum at each distance (dashed regression curve). Similarly for the noise term within ξ_m^{SDMA}, as shown in Fig. 3.3b,

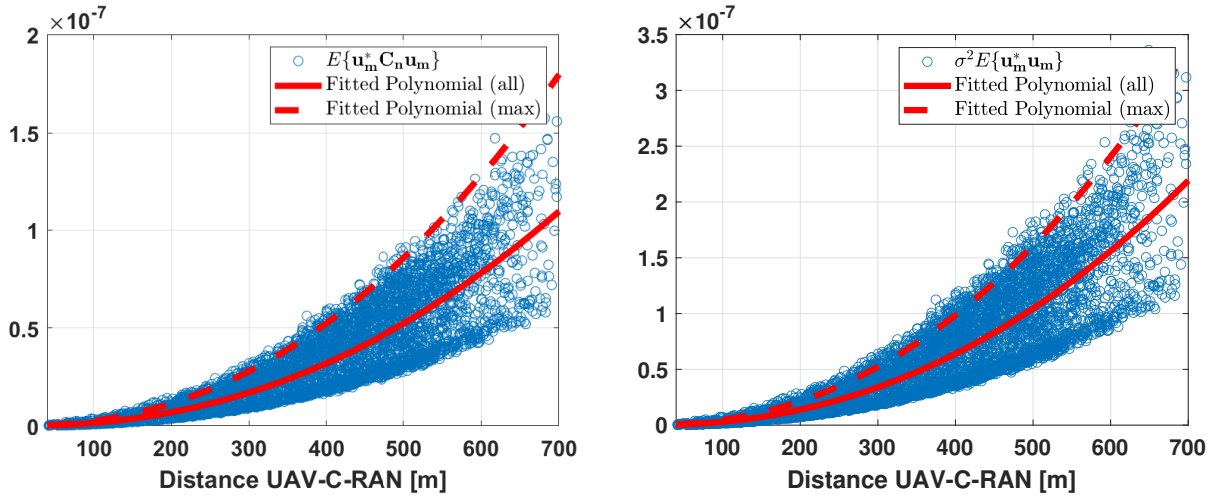
$$\lim_{\epsilon \to 0} \frac{\frac{1}{N^2}\mathrm{tr}(\boldsymbol{\Phi}_m \boldsymbol{T}'(\epsilon, \boldsymbol{I}))}{\left(1 + \frac{1}{N}\mathrm{tr}(\boldsymbol{\Phi}_m \boldsymbol{I})\right)^2} \approx c_m^{(\mathrm{n})} d_m^{\kappa}, \tag{3.43}$$

with a corresponding regression parameter $c_m^{(\mathrm{n})}$. After comparing the respective performances, the solid regression curves are chosen and the gradient satisfies

$$\nabla_{\boldsymbol{q}_m} \overline{\mathrm{SINR}}_k \approx \frac{\nabla_{\boldsymbol{q}_m} \gamma_{k,m} \, \mathrm{Den}_m - \gamma_{k,m} \nabla_{\boldsymbol{q}_m} \mathrm{Den}_m}{\mathrm{Den}_m^2} p_k, \tag{3.44}$$

for $m \in \mathcal{F}_k$ with

$$\mathrm{Den}_m = \sum_{\substack{i \in \mathcal{U}_m \\ i \neq k}} \frac{\gamma_{i,m}}{1 + e_i} p_i + \sum_{\forall i} r_{i,m} p_i - \sum_{i \in \mathcal{U}_m} \gamma_{i,m} p_i + \sigma^2 + \frac{d_m^{\kappa}}{\rho_m} \left(\sum_{n=1}^{M} \mathrm{p}_n c_m + \sigma^2 c_m^{(n)} \right).$$



(a) (b)

Figure 3.3: Curve fitting with every data point (solid) or only the maximum at each distance (dashed) for (a) $\mathbb{E}\{\boldsymbol{u}_m^* \boldsymbol{C}_n \boldsymbol{u}_m\}$ and (b) $\sigma^2 \mathbb{E}\{\boldsymbol{u}_m^* \boldsymbol{u}_m\}$.

User Transmit Power

It can be verified that $\min_k \overline{\mathrm{SINR}}_k$ in (3.41) satisfies the definition of competitive utility function and the constraints are monotonic. Thus, [104, Alg. 1] converges to the optimal user transmit powers.

UAV Transmit Power

To tackle this subproblem, it is convenient to reformulate (3.41) so as to capitalize on the fact that for any set of functions $f_k(\boldsymbol{x})$, the problem $\max_{\boldsymbol{x}} \min_k f_k(\boldsymbol{x})$ is equivalent to

$$\begin{aligned} &\max_{\boldsymbol{x},t} \ t \\ &\text{s.t.} f_k(\boldsymbol{x}) \geq t \ \ \forall k. \end{aligned} \tag{3.45}$$

It follows that the optimization in (3.41) w.r.t. p_m is equivalent to

$$\begin{aligned} &\max_{\mathrm{p}_m,\, t,\, y_{k,m}} \quad t \\ \text{s.t.} &\sum_{m\in\mathcal{F}_k} y_{k,m}^2 \geq t \;\forall k \\ &\qquad y_{k,m}^2 \leq \overline{\mathrm{SINR}}_{k,m} \;\forall k, m \end{aligned} \tag{3.46}$$

where $y_{k,m}$ is a slack variable satisfying $y_{k,m}^2 = \overline{\mathrm{SINR}}_{k,m}$ when the optimum solution is attained; elsewhere, the value of $y_{k,m}^2$ can be increased for a higher cost function.

While equivalent to the original problem, (3.46) is neither convex nor concave. To tackle it, we leverage the successive convex approximation method (SCA) [105]. First, given that $y_{k,m}^2$ is convex, it accepts a lower bound of the type $y_{k,m}^2 \geq y_{k,m}^{2\,(\mathrm{lb})}$ with

$$y_{k,m}^{2\,(\mathrm{lb})} = y_{k,m}^{2\,(\mathrm{p})} + 2y_{k,m}^{(\mathrm{p})}(y_{k,m} - y_{k,m}^{(\mathrm{p})}) \tag{3.47}$$

where $y_{k,m}^{2\,(\mathrm{p})}$ is the value of $y_{k,m}^2$ at approximation point p. Then, defining for the sake of brevity

$$\lambda_{m,n} = \lim_{\epsilon\to 0} \frac{\frac{1}{N^2}\mathrm{tr}(\boldsymbol{\Phi}_m \boldsymbol{T}'(\epsilon, \boldsymbol{C}_n))}{\left(1 + \frac{1}{N}\mathrm{tr}(\boldsymbol{\Phi}_m \boldsymbol{T})\right)^2} \tag{3.48}$$

and

$$\lambda_m^{'} = \lim_{\epsilon\to 0} \frac{\frac{1}{N^2}\mathrm{tr}(\boldsymbol{\Phi}_m \boldsymbol{T}'(\epsilon, \boldsymbol{I}))}{\left(1 + \frac{1}{N}\mathrm{tr}(\boldsymbol{\Phi}_m \boldsymbol{I})\right)^2}, \tag{3.49}$$

a similar procedure is followed to derive a lower bound for $\overline{\mathrm{SINR}}_{k,m}$, which is convex w.r.t. $\frac{1}{\mathrm{p}_m}\left(\sum_{n\neq m}^{M} \lambda_{m,n} r_n \mathrm{p}_n + \sigma^2 \lambda_m^{'}\right)$ and therefore satisfies $\overline{\mathrm{SINR}}_{k,m} \geq \overline{\mathrm{SINR}}_{k,m}^{(\mathrm{lb})}$, the latter term included in Eq. (3.50) and with

$$\overline{\mathrm{SINR}}_{k,m}^{(\mathrm{lb})} = \overline{\mathrm{SINR}}_{k,m}^{(\mathrm{p})} - \zeta_{k,m}^{\mathrm{SD}} \left[\frac{1}{\mathrm{p}_m} \left(\sum_{n=1}^{M} \lambda_{m,n} r_n \mathrm{p}_n + \sigma^2 \lambda_m^{'} \right) - \frac{1}{p_m^{(\mathrm{p})}} \left(\sum_{n=1}^{M} \lambda_{m,n} r_n \mathrm{p}_n^{(\mathrm{p})} + \sigma^2 \lambda_m^{'} \right) \right] \tag{3.50}$$

$$\frac{y_{k,m}^2}{\sum_{n=1}^{M} \lambda_{m,n} r_n \mathrm{p}_n + \sigma^2 \lambda_m^{'}} \leq -\frac{\zeta_{k,m}^{\mathrm{SD}}}{\mathrm{p}_m} + \frac{\overline{\mathrm{SINR}}_{k,m}^{(\mathrm{p})} + \frac{\zeta_{k,m}^{\mathrm{SD}}}{p_m^{(\mathrm{p})}} \left(\sum_{n=1}^{M} \lambda_{m,n} r_n \mathrm{p}_n^{(\mathrm{p})} + \sigma^2 \lambda_m^{'} \right)}{\sum_{n=1}^{M} \lambda_{m,n} r_n \mathrm{p}_n + \sigma^2 \lambda_m^{'}}, \tag{3.52}$$

$$\zeta_{k,m}^{\mathrm{SD}} = -\frac{\partial \overline{\mathrm{SINR}}_{k,m}}{\partial \frac{1}{\mathrm{p}_m} \left(\sum_{n=1}^{M} \lambda_{m,n} r_n \mathrm{p}_n + \sigma^2 \lambda_m^{'} \right)} \Bigg|_{\mathrm{p}_i = \mathrm{p}_i^{(\mathrm{p})}\ i=1,\ldots,M}. \tag{3.51}$$

Still, $y_{k,m}^2 \leq \overline{\mathrm{SINR}}_{k,m}^{(\mathrm{lb})}$ is not convex because of the quotients $\frac{\mathrm{p}_n}{\mathrm{p}_m}$ in (3.50). Division of both sides of the inequality by $\sum_{n=1}^{M} \lambda_{m,n} r_n \mathrm{p}_n + \sigma^2 \lambda_m^{'}$ results in (3.52) where the only nonconvex term is the second in the right-hand side, which itself accepts a lower bound w.r.t. $\sum_{n=1}^{M} \lambda_{m,n} r_n \mathrm{p}_n + \sigma^2$. As a consequence, further application of the SCA technique results in the convex set of constraints derived in (3.53). Altogether then, an approximate convex reformulation of (3.46) is

$$\begin{aligned} \max_{\mathrm{p}_m,\, t,\, y_{k,m}} \quad & t \\ \text{s.t.} \quad & \sum_{m \in \mathcal{F}_k} y_{k,m}^{2\,(\mathrm{lb})} \geq t \ \forall k \end{aligned} \tag{3.54}$$

$$\begin{aligned} \frac{y_{k,m}^2}{\sum_{n=1}^{M} \lambda_{m,n} r_n \mathrm{p}_n + \sigma^2 \lambda_m^{'}} \leq & -\frac{\zeta_{k,m}^{\mathrm{SD}}}{\mathrm{p}_m} + \left[\overline{\mathrm{SINR}}_{k,m} + \frac{\zeta_{k,m}^{\mathrm{SD}}}{p_m^{(\mathrm{p})}} \left(\sum_{n=1}^{M} \lambda_{m,n} r_n \mathrm{p}_n^{(\mathrm{p})} + \sigma^2 \lambda_m^{'} \right) \right] \times \\ & \left[\frac{1}{\sum_{n=1}^{M} \lambda_{m,n} r_n \mathrm{p}_n^{(\mathrm{p})} + \sigma^2 \lambda_m^{'}} - \frac{1}{\left(\sum_{n=1}^{M} \lambda_{m,n} r_n \mathrm{p}_n^{(\mathrm{p})} + \sigma^2 \lambda_m^{'} \right)^2} \left(\sum_{n=1}^{M} \lambda_{m,n} r_n \mathrm{p}_n - \sum_{n=1}^{M} \lambda_{m,n} r_n \mathrm{p}_n^{(\mathrm{p})} \right) \right]. \end{aligned} \tag{3.53}$$

and further subject to (3.53). This problem can be efficiently solved with standard optimization tools [106]. In addition, it can be shown that, given the tightness of the local approximations, the sequence of objective values generated by the SCA applied to (3.54) is monotonically non-decreasing with an upper bound, and therefore converges.

3.6 FDMA-SDMA

Finally, under a combined FDMA and SDMA fronthaul, the application of the N-dimensional fronthaul combiner $\boldsymbol{u}_m$ in (3.20) yields the same model of (3.21) with $c_m = \boldsymbol{u}_m^* \hat{\boldsymbol{h}}_m \sqrt{\rho_m}$ and an equivalent noise $\boldsymbol{n}' = (n'_1, \ldots, n'_M) \in \mathbb{C}^{M \times 1}$ with

$$n'_m = \sum_{j \in \mathcal{M}_\ell}^{M} \boldsymbol{u}_m^* \boldsymbol{h}_j \sqrt{\rho_j} y_j - \boldsymbol{u}_m^* \hat{\boldsymbol{h}}_m \sqrt{\rho_m} \Big(\sum_{k \in \mathcal{U}_m} \hat{g}_{k,m} x_k \Big) + \boldsymbol{u}_m^* \boldsymbol{n}_c.$$

The SDMA component requires $N \geq \max\{|\mathcal{M}_\ell|\,,\, \ell = 1, \ldots, L\}$ and, with the fronthaul combiner $\boldsymbol{u}_m$ set to be ZF, $\boldsymbol{u}_m^* \hat{\boldsymbol{h}}_j = \delta_{m,j}$ for $m, j \in \mathcal{M}_\ell$.

3.6.1 Large-Dimensional Analysis

Under the same assumptions as for pure FDMA or SDMA and given the ZF nature of $\boldsymbol{u}_m$, Thm. 2 is applied to characterize the asymptotic equivalent noise terms.

Theorem 4. *With a combined FDMA and ZF-SDMA fronthaul, $|\mathcal{F}_k|, |\mathcal{U}_m| \to \infty \; \forall\, k, m$, $N \to \infty$ with $N \geq \max\{|\mathcal{M}_\ell|\,,\, \ell = 1, \ldots, L\}$, and MMSE subset combining, $\mathrm{SINR}_k - \overline{\mathrm{SINR}}_k \overset{\text{a.s.}}{\to} 0$ with $\overline{\mathrm{SINR}}_k = \sum_{m \in \mathcal{F}_k} \overline{\mathrm{SINR}}_{k,m}$ and $\overline{\mathrm{SINR}}_{k,m}$ given in Eq. (3.55). The application of Thm. 2 to $\mathbb{E}\{n'_m n'^*_m\}$ results in*

$$\xi_m^{FS} = \lim_{\epsilon \to 0} \sum_{j \in \mathcal{M}_\ell} p_j \frac{\frac{1}{N^2} \mathrm{tr}(\boldsymbol{\Phi}_m \boldsymbol{T}'(\epsilon, \boldsymbol{C}_j))}{\big(1 + \frac{1}{N} \mathrm{tr}(\boldsymbol{\Phi}_m \boldsymbol{T})\big)^2} + \sigma^2 \frac{\frac{1}{N^2} \mathrm{tr}(\boldsymbol{\Phi}_m \boldsymbol{T}'(\epsilon, \boldsymbol{I}))}{\big(1 + \frac{1}{N} \mathrm{tr}(\boldsymbol{\Phi}_m \boldsymbol{I})\big)^2}.$$

$$\overline{\mathrm{SINR}}_{k,m} = \frac{\gamma_{k,m} p_k}{\sum\limits_{\substack{i \in \mathcal{U}_m \\ i \neq k}} \frac{\gamma_{i,m}}{1+e_{i,k}} p_i + \sum\limits_{\forall i} r_{i,m} p_i - \sum\limits_{i \in \mathcal{U}_m} \gamma_{i,m} p_i + \sigma^2 + \frac{\xi_m^{\mathrm{FS}}}{\rho_m}}. \tag{3.55}$$

The coefficients e_j are obtained iteratively by $e_j = \lim_{n\to\infty} e_j^{(n)}$, $e_j^{(0)} = |\mathcal{F}_j|$, and are obtained similarly to the ones presented in Eq. (3.40).

Proof. Proceed as in Appendix A.6. □

From the continuous mapping theorem, $\mathrm{SE}_k - \frac{1}{L}\left(1 - \frac{\tau}{\tau_\mathrm{c}}\right)\log_2(1 + \overline{\mathrm{SINR}}_k) \xrightarrow{\text{a.s.}} 0$.

3.6.2 Problem Formulation

The max-min SINR optimization problem in this case boils down to

$$\begin{aligned} \max_{\boldsymbol{q}_m, H_m, p_k, \mathrm{p}_m} \; & \min_k \sum_{m \in \mathcal{F}_k} \overline{\mathrm{SINR}}_{k,m} \\ \text{s.t.} \quad & H_{\min} \leq H_m \leq H_{\max},\; p_k \leq p_{\max},\; \mathrm{p}_m \leq p_{\max}, \end{aligned} \tag{3.56}$$

for $\overline{\mathrm{SINR}}_{k,m}$ given in (3.55).

Deployment Optimization

As in pure SDMA, the terms in ξ_m^{FS} can be approximated by a linear combination of polynomials whose variable is the distance between the UAV and the C-RAN. Therefore,

$$\nabla_{\boldsymbol{q}_m} \overline{\mathrm{SINR}}_k \approx \frac{\nabla_{\boldsymbol{q}_m} \gamma_{k,m} \, \mathrm{Den}_m - \gamma_{k,m} \nabla_{\boldsymbol{q}_m} \mathrm{Den}_m}{\mathrm{Den}_m^2} p_k, \tag{3.57}$$

for $m \in \mathcal{F}_k$ where

$$\mathrm{Den}_m = \sum_{\substack{i \in \mathcal{U}_m \\ i \neq k}} \frac{\gamma_{i,m}}{1 + e_{i,k}} p_i + \sum_{\forall i} r_{i,m} p_i - \sum_{i \in \mathcal{U}_m} \gamma_{i,m} p_i + \sigma^2 + \frac{d_m^{\kappa}}{\rho_m} \Bigg(\sum_{j \in \mathcal{M}_\ell} \mathrm{p}_j c_m + \sigma^2 c_m^{(n)} \Bigg).$$

User Power Allocation

Again, [104, Alg. 1] converges to the user transmit power that maximizes $\min_k \overline{\mathrm{SINR}}_k$ in (3.56).

UAV Power Allocation

Because of space limitations, the derivation of the UAV transmit power optimization is not included. Similar steps as in Sec. 3.4.2 should be followed.

3.7 GB-GS Deployment Algorithm

Equipped with the 2D gradients derived in the previous section, the UAV locations could be updated iteratively as

$$\boldsymbol{q}_m^{(t)} \leftarrow \boldsymbol{q}_m^{(t)} + \rho^{(t)} \nabla_{\boldsymbol{q}_m} \overline{\mathrm{SINR}}_k^{(t)} \big|_{\boldsymbol{q}_m = \boldsymbol{q}_m^{(t)}}, \tag{3.58}$$

where t is the iteration counter and $\rho^{(t)}$ a decreasing function of t for convergence reasons. However, the nonconvex nature of the problem may result in low-quality solutions. Moreover, the altitudes should be part of the optimization as well. For such a complex optimization, an attractive approach is that of stochastic optimization. This work leverages the well-known GS technique in conjunction with (3.58). Concretely, for a set of possible states Θ, GS aims

at solving

$$\max_{\{\boldsymbol{\ell}_m\ \forall m\}\in\Theta}\ \min_{k}\ \overline{\mathrm{SINR}}_k, \tag{3.59}$$

where $\boldsymbol{\ell}_m = \big(\boldsymbol{q}_m, H_m\big)$ corresponds to the 3D locations that are iteratively updated according to a certain probability distribution [84].

Let $\eta^{(t)} = \min\limits_{k} \overline{\mathrm{SINR}}_k^{(t)} \equiv \overline{\mathrm{SINR}}_{k_{\min}^{(t)}}^{(t)}$ be the cost function at Iteration t whereas $k_{\min}^{(t)}$ represents the index of the user with lowest $\overline{\mathrm{SINR}}^{(t)}$. In SDMA, such $\eta^{(t)}$ is a function of all UAVs since those within subset $\mathcal{F}_k$ provide service while the rest create fronthaul interference. For the other two fronthaul strategies, only a subset of UAVs are relevant. To maintain a generic formulation, we derive the algorithm under SDMA fronthauling; minor changes apply for FDMA and FDMA-SDMA. The cost function can be expressed as $\eta^{(t)}(\boldsymbol{\ell}_m^{(t)},\ \forall m)$ and the 3D locations of the M UAVs are updated sequentially, starting with the lowest index.

Denote by $\mathcal{L}_{-m}^{(t)} = \{\boldsymbol{\ell}_1^{(t+1)}, \ldots, \boldsymbol{\ell}_{m-1}^{(t+1)}, \boldsymbol{\ell}_{m+1}^{(t)}, \ldots, \boldsymbol{\ell}_M^{(t)}\}$ the set of UAVs satisfying:

a) UAVs $1, \ldots, m-1$ have already updated their locations to $t+1$;

b) the locations of UAVs $m+1, \ldots, M$ still need to be updated; and

c) UAV m is excluded.

The cost function allows an alternative expression as a function of $\mathcal{L}_{-m}^{(t)}$, namely $\eta^{(t)}(\boldsymbol{\ell}_m^{(t)}, \mathcal{L}_{-m}^{(t)})$. From [84], the probability of the mth UAV updating its 3D location to $\boldsymbol{\ell}_m^{(t+1)}$ is

$$\Pr\Big\{\boldsymbol{\ell}_m^{(t+1)} | \boldsymbol{\ell}_m^{(t)}, \mathcal{L}_{-m}^{(t)}\Big\} = \frac{\exp\big\{\gamma\, \eta^{(t)}\big(\boldsymbol{\ell}_m^{(t+1)}, \mathcal{L}_{-m}^{(t)}\big)\big\}}{\sum\limits_{\hat{\boldsymbol{\ell}}_m^{(t+1)}\in\Theta^{t+1}} \exp\big\{\gamma\, \eta^{(t)}\big(\hat{\boldsymbol{\ell}}_m^{(t+1)}, \mathcal{L}_{-m}^{(t)}\big)\big\}}, \tag{3.60}$$

where γ is a fixed parameter and Θ^{t+1} represents the possible locations that UAV m can explore at Iteration $t+1$. To reduce the search space, the number of such locations is

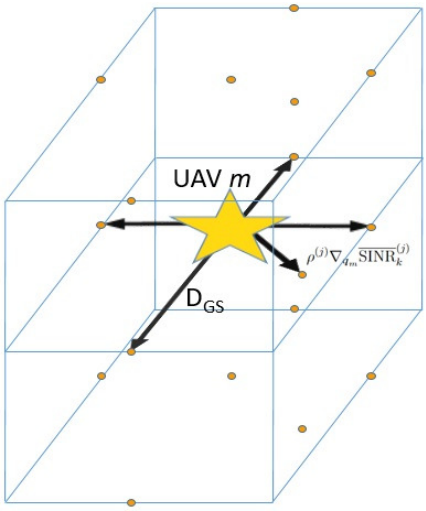


Figure 3.4: 3D search space for UAV m with the dots representing Θ^{t+1}.

limited to $|\Theta^{t+1}| = 18$ (see Fig. 3.4). The options are to stay, move north, move south, move east, move west, and move in the direction of the gradient in (3.58), as well as the corresponding twelve locations at a higher and lower altitude. The search space is the set of 3D positions confined between some minimum and maximum altitudes, respectively $H_{\min}$ and $H_{\max}$. And, after each iteration, matrix $\boldsymbol{M}^{(s)}$ is updated. A summary of the process is included in Algorithm 2 where ϵ is a stopping parameter.

Algorithm 2 GS-GB Algorithm

Require: at $t = 0$, initialize UAV locations, $\boldsymbol{\ell}_m^{(0)}$, and cost function, $\eta^{(0)}$
  while $\frac{|\eta^{(t+1)} - \eta^{(t)}|}{\eta^{(t)}} > \varepsilon$ **do**
    find the user with lowest cost function, $k_{\min}^{(t)}$
    for all $m = 1, \ldots, M$ **do**
      obtain $\mathcal{L}_{-m,k_{\min}^{(t)}}^{(t)}$.
      create the reduced search space with eighteen possible locations, Θ^{t+1}.
      compute the cost function at the possible new locations, $\eta^{(t)}\big(\hat{\boldsymbol{q}}_m^{(t+1)}, \mathcal{L}_{-m,k_{\min}^{(t)}}^{(t)}\big)$
        for $\hat{\boldsymbol{\ell}}_m^{(t+1)} \in \Theta^{t+1}$.
      calculate (3.60) and choose one movement accordingly, obtaining $\boldsymbol{\ell}_m^{(t+1)}$.
    end for
  end while

It is proved in [85] that, for large enough γ and $t \to \infty$, the solution for (3.60) converges to the optimal solution with probability 1.

3.8 Numerical Results

To evaluate the performance, we consider a 1 km^2 universe, wrapped around to avoid boundary effects. The simulation parameters are listed in Table 5.1, selected based on the cell-free and UAV literature [33, 90–93]. Consistent with the neglect of pilot contamination, we consider $\tau = 200$ for a 3.2% pilot overhead. To ensure connectivity to multiple UAVs, the $[m, k]$ entry of $\boldsymbol{M}^{(s)}$ is 1 if $d_{k,m} \leq R_{\max}$ for $R_{\max} = 400$ m. The fading is IID, such that $\boldsymbol{\Omega_m} = \boldsymbol{I}$. Moreover, the noise arising in the fronthaul is scaled by a factor of M and L in SDMA and FDMA-SDMA, respectively, to account for the bandwidth difference among the schemes. As for the GB-GS algorithm, and noting that other choices may be as effective, $\rho^{(t)} = T_{\text{GS}} \cdot 1.005^{-t}$ where T_{GS} depends on the fronthaul scheme and is set to $T_{\text{GS}} = 80$ for FDMA and to $T_{\text{GS}} = 40$ for SDMA and FDMA-SDMA. In addition, $\mathsf{D_{GS}} = 1$ m (see Fig. 3.4) while $\gamma = 10$ and $\epsilon = 0.01$. The entries of $\boldsymbol{M}^{(s)}$ are updated at every iteration of the GS-GB algorithm following the aforementioned distance-based rule with the frequency band allocations drawn at random. Finally, the user locations abide by a Poisson Point Process and the optimization algorithm that combines deployment and power optimization is tested over 100 deployments. When presenting results, the optimized deployment is denoted by A-O (after optimization) while a square grid UAV deployment, denoted by B-O (before optimization), serves as a benchmark.

We first evaluate the performance with FDMA fronthauling under a variety of parameters while validating the asymptotic derivations. Concretely, Fig. 3.5a plots the average per user spectral efficiency for different M and K. Additionally, Fig. 3.5b verifies Thm. 1 for different K/M. From Fig. 3.5b, a smaller K/M, i.e., more UAVs per ground user, provides better SINRs while allowing more UAVs to participate in the decoding of each user. Conversely, by looking at Fig. 3.5a, for fixed K, increasing M is not helpful in terms of spectral efficiency given the $1/M$ pre-log factor in (3.25). Finally, Fig. 3.5b shows that the derived results are indeed tight for finite-dimensional systems given the small gap between the $\mathbb{E}\{\text{SINR}\}$ and

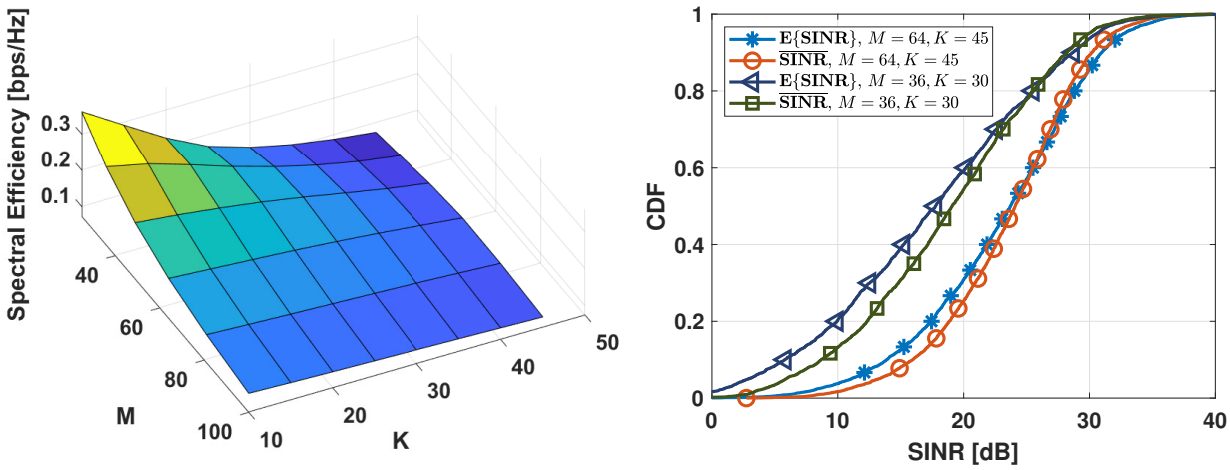
Table 3.1: Simulation parameters

Description	Parameter	Value
Maximum transmit power	$p_{\max}$, $\mathrm{p}_{\max}$	100 mW
Pathloss at 1 m	β_0	-30 dB
Pathloss exponent	κ	2.2
Dense urban parameters	A_1, A_2	0, 6.4 dB
Noise power for access and FDMA	σ^2	-96 dBm
Antenna beamwidth	α_m	4
Operating frequency	f_{c}	2.4 GHz
Maximum UAV velocity	$v_{\max}$	10 m/s
Coherence bandwidth	B_{c}	1 MHz
Number of UAVs	M	64
Number of users	K	45
Maximum and minimum altitude	$H_{\max}$, $H_{\min}$	25, 100 m

$\overline{\mathrm{SINR}}$ curves, with the advantage of only depending on large-scale parameters. A similar assessment is conducted for SDMA fronthauling in Fig. 3.6. The number of antennas is set to $N = 1.2N$. Interestingly, although SDMA provides lower SINRs compared to FDMA, an increase in M results in an improved spectral efficiency provided that the pre-log factor in Eq. (3.25) is one. This is because of the multiplexing gain in SDMA. Finally, Fig. 3.7 presents the results for FDMA-SDMA. We consider $L = \frac{M}{5}$ and $N = 1.2L$, and the observations are consistent with those of FDMA and SDMA both in terms of (a) the tendency when varying the network load, and (b) the match between real and asymptotic SINR derivations.

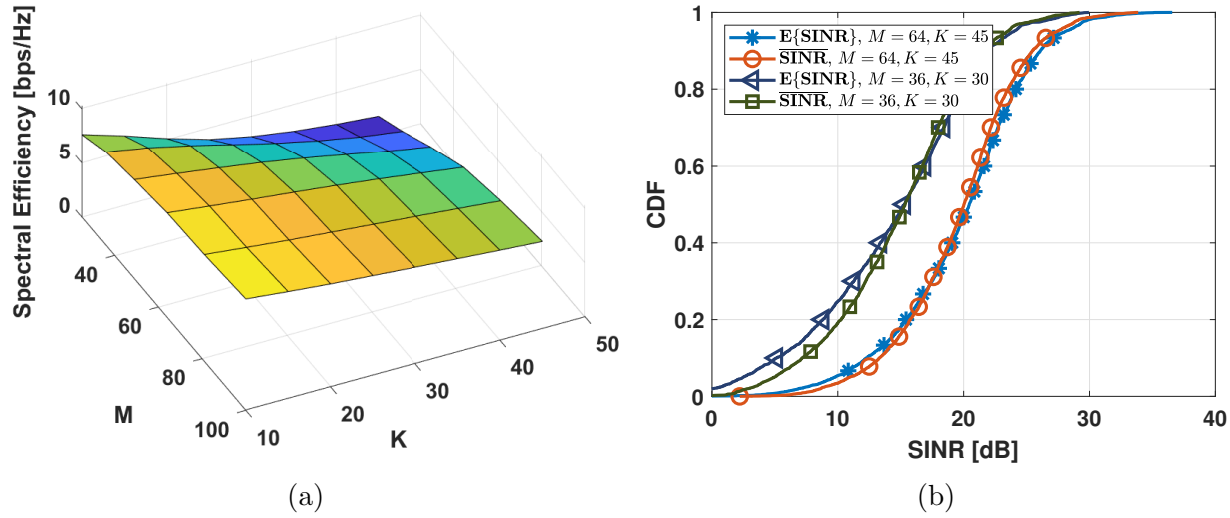
As one would expect, the SINRs achieved with FDMA are decidedly higher because of the orthogonal nature of the transmisions and reduced noise bandwidth. In contrast (see Fig. 3.8), when measuring the sum spectral efficiency, SDMA vastly outperforms FDMA thanks to its spatial multiplexing gain. The hybrid FDMA-SDMA scheme balances the two.

Turning now to the deployment optimization, Fig. 3.9 presents results under FDMA fronthauling with different κ and α_m. Particularly, with the aim of keeping a small legend, the values shown in such figures are (κ, α_m) where B-O and A-O, recall, stand for before and after optimization, respectively. Specifically, Fig. 3.9a plots the B-O and A-O distributions;



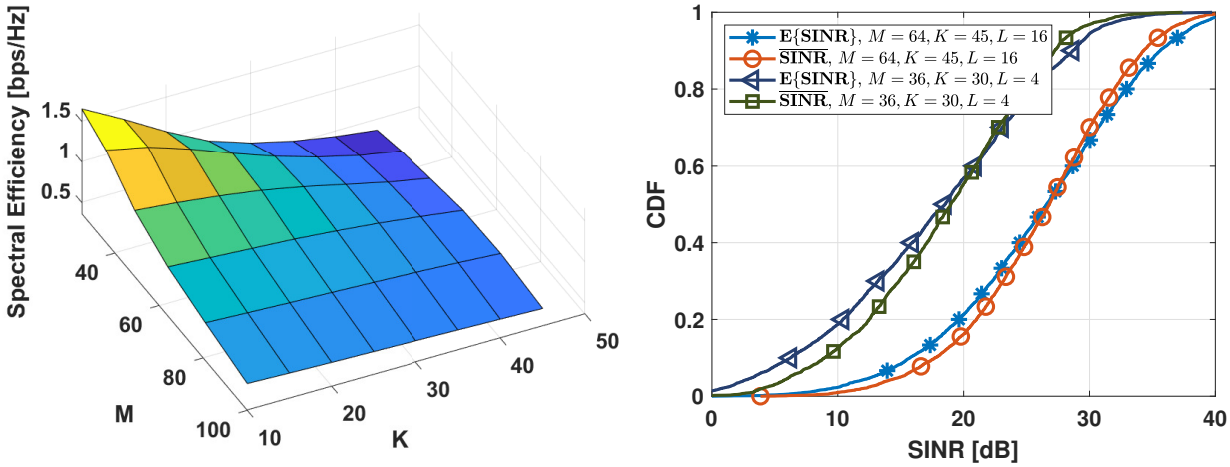
(a) (b)

Figure 3.5: (a) FDMA performance for different K, M; (b) validation of Thm. 1.



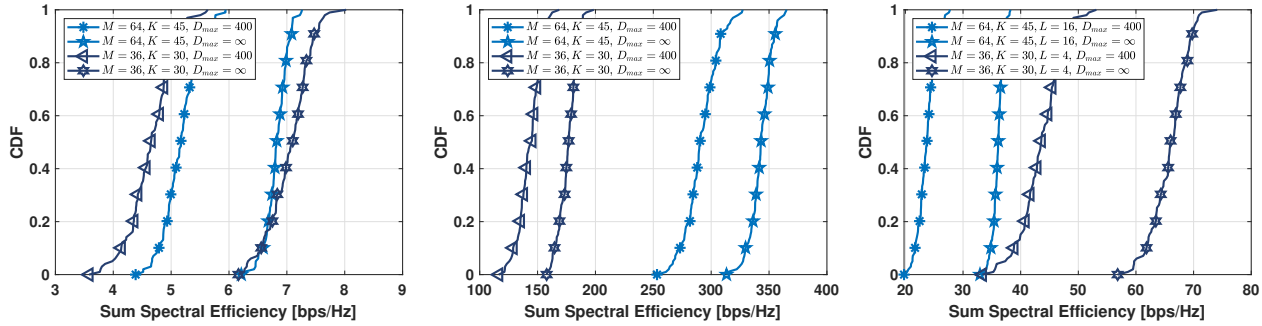
(a) (b)

Figure 3.6: (a) SDMA under different K, M; (b) validation of Thm. 3.



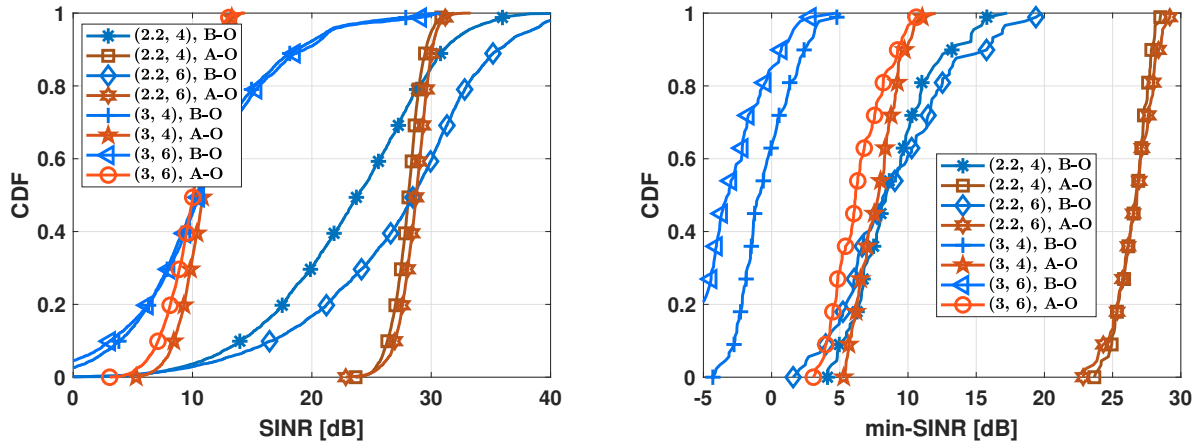
(a) (b)

Figure 3.7: (a) FDMA-SDMA for different K, M, $L = \frac{M}{5}$; (b) validation of Thm. 4.



(a) (b) (c)

Figure 3.8: Sum spectral efficiency under different network loads K/M for (a) FDMA; (b) SDMA; (c) FDMA-SDMA.



(a) (b)

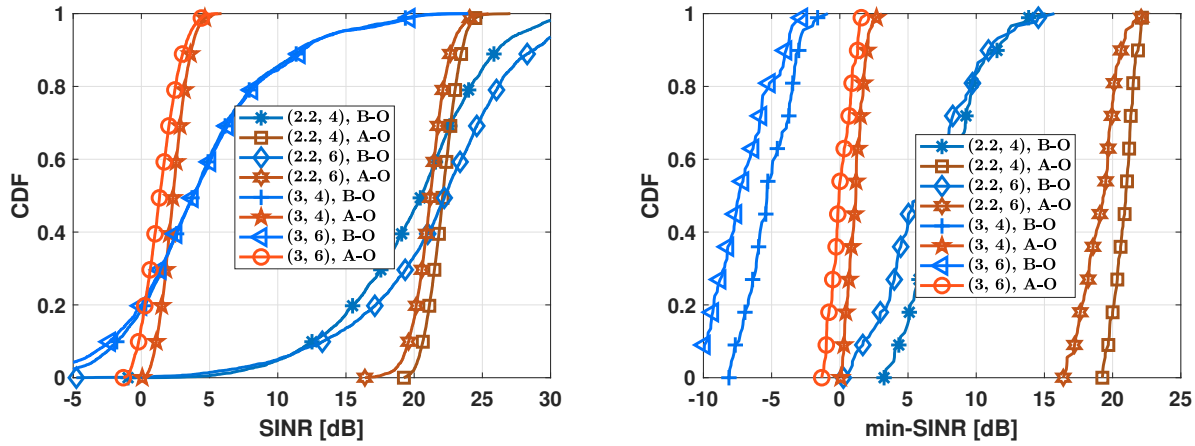
Figure 3.9: CDFs B-O and A-O for different κ and α_m under an FDMA fronthaul (a) $\mathbb{E}\{\text{SINR}_k\}$; (b) min-$\mathbb{E}\{\text{SINR}_k\}$.

the optimization is highly effective, with at least 45% of users improving their SINR as a result. Then, Fig. 3.9b presents the CDF of the minimum SINR, where the optimization yields a 8-18 dB gain.

Fig. 3.10 presents results for the SDMA fronthaul parameterized by (κ, α_m). For $\kappa = 2.2$, 40-60% of users enjoy an improved SINR after the optimization. For a higher pathloss exponent, $\kappa = 3$, it is 20-40%. The minimum SINR improves by 5-17 dBs for a variety of κ and α_m.

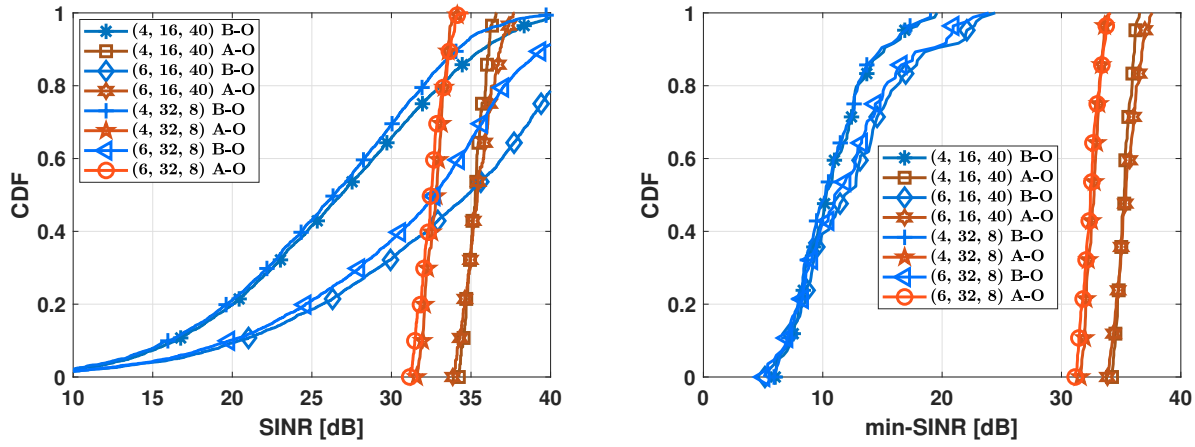
Results for the third fronthaul option, which combines FDMA and SDMA, are included in Fig. 3.11 for different (α_m, L, N). Again, the combination of deployment and power optimization highly increases the SINR experienced by those users with unfavorable initial conditions. Concretely, at least 50% of the SINRs are increased depending on the network parameters while the gains in terms of minimum SINR are 12-27 dB.

In Figs. 3.12a and 3.12b, respectively for FDMA and SDMA fronthauls, we provide insight on the contributions to the optimization gain. Precisely, we present the CDFs B-O, A-O, only optimizing the deployment (DEPLOY-O) and only optimizing the transmit pow-



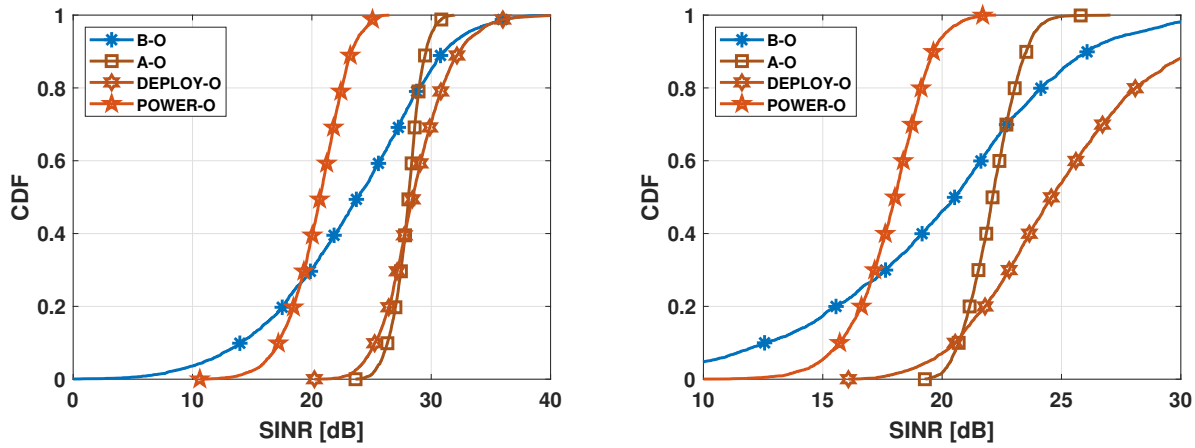
(a) (b)

Figure 3.10: CDF B-O and A-O parameterized by (κ, α_m) under SDMA fronthaul with $N = 80$ (a) $\mathbb{E}\{\text{SINR}_k\}$; (b) $\min -\mathbb{E}\{\text{SINR}_k\}$.



(a) (b)

Figure 3.11: CDF B-O and A-O parameterized by (α_m, L) and N under FDMA-SDMA fronthaul (a) $\mathbb{E}\{\text{SINR}_k\}$; (b) $\min -\mathbb{E}\{\text{SINR}_k\}$.



(a) (b)

Figure 3.12: $\mathbb{E}\{\text{SINR}_k\}$ CDF B-O, A-O, DEPLOY-O and POWER-O for $\kappa = 2.2$ and $\alpha_m = 4$ for (a) FDMA fronthaul; (b) SDMA fronthaul.

ers (POWER-O) (POWER-O) for (a) FDMA ; (b) SDMA. Power optimization helps to increase the lowest SINRs for 20-30% of users. However, the main source of gain is from the deployment optimization, improving 90-100% of the user SINRs.

3.9 Conclusions

This chapter has considered a cell-free network with wireless access and fronthaul links. For the latter, a variety of schemes have been considered, namely FDMA, SDMA, and FDMA-SDMA. Under Rician fading for the access and fronthaul links, deterministic equivalents for the SINR with MMSE reception have been provided for the three fronthaul schemes. Based on these deterministic expressions, the minimum SINR has been maximized with respect to (a) the 3D UAV locations, (b) user transmit power, and (c) UAV transmit power. A combination of gradient-based and Gibbs sampling algorithms has been employed for the former, and classic optimization techniques for the latter two.

Extensive results have shown how the optimization of the minimum SINR provides superior

and fairer conditions in the network. Gains of 5-27 dB are achieved depending on the fronthaul techniques and network parameters. Further results have uncovered that the lion's share of the improvements can be attributed to the deployment optimization, with marginal additional gains associated with the optimization to the transmit powers.

Chapter 4

A Reinforcement Learning Approach for Wildfire Tracking with UAV Swarms

4.1 Introduction

Wildfires are a growing concern worldwide and cause tremendous environmental and economic damage. It is therefore crucial to predict, track, and monitor them to facilitate the actions of firefighters and emergency responders [44–46]. Additionally, the study of wildfires can help to identify areas at risk and to understand the nature of fire propagation, along with possible causes and factors that contribute to their spread [107–110].

An attractive means to monitor, track, and sense events are UAVs given the (*i*) availability of cameras and sensors, (*ii*) wireless connectivity to ground APs, (*iii*) long-lasting batteries, and (*iv*) easy control and maneuverability [4–8]. UAVs can fly at low altitudes and collect data ranging from high-resolution images to heat signatures [111–116]. However, the transmission

of such data is contingent on the connectivity. Regardless of whether APs are damaged by fire, the flow of information should be guaranteed. This relates to the network's resiliency and, in that respect, wireless systems are evolving from traditional cellular structures towards cell-free arrangements. These are especially appealing when a high degree of reliability is needed, given that users can then communicate with multiple APs [31–33, 35–37, 39, 41, 42, 100]. Initial results have demonstrated the efficacy and benefits of a cell-free architecture for UAVs [9–13] and hence this is the structure adopted here.

The design of deployments and trajectories for UAVs in cellular and wireless sensor networks is a problem of growing interest [14–25, 27–29], yet, to the best of our knowledge, this is the first work that considers a multi-UAV trajectory optimization to track an event with cell-free connectivity. This gives rise to various challenges, mainly related to the complexity and time-varying nature of the problem, which precludes the use of classic optimization techniques. Compounding this complexity, the lifespan of UAV batteries is limited and thus, once a UAV is low on energy, a recharge is needed. Authors in [117] provide an exhaustive review of different techniques to recharge or replace UAV batteries, along with the required time. Altogether, the problem at hand breaks down into two stages of *tracking* and *charging*. Two decidedly nonconvex optimization problems arise, one for each stage, with the switching from tracking to charging based on the remaining energy of each UAV.

Reinforcement learning (RL) provides a framework where an agent learns optimal policies based on interactions with the environment and on feedback in the form of rewards [47]. For settings with a large number of states and/or actions, strategies combining RL with neural networks have been devised, converging to the so-called deep Q-learning (DQL) [118–123]. These methods have already been considered for UAV trajectory optimizations [124–128]. A recent actor-critic algorithm named twin delayed deep deterministic policy gradient (TD3) has been shown to perform better than its predecessors [48] in terms of stability, exploration capabilities, handling of continuous actions, and sampling efficiency. The field of

UAV trajectory optimization has greatly benefited from this new algorithm [57, 59, 129, 130], which is applied in this work as well. The main contributions of the chapter are as follows:

- An analytical framework is set forth to describe the tracking of a wildfire by a swarm of UAVs equipped with cameras. The connectivity between UAVs and APs is through a cell-free network, with Rician fading, channel estimation, and minimum mean square error (MMSE) reception explicitly modeled.
- Based on communication and mechanical constraints, the tracking and charging multi-UAV optimization problems are formulated.
- For each of the two problems, the UAV trajectory and the transmit power optimizations are confronted using the TD3 approach. While, for the tracking stage, the objective is to monitor the wildfire perimeter ensuring a correct reception of the video/images in a timely fashion, during the charging stage the goal is to reach a charging point with the minimum energy expenditure.
- The impact on the wildfire coverage of parameters such as the number of UAVs, the allowed flying altitudes, and the UAVs' energy is established. A similar analysis is conducted for the charging problem, with the remaining energy as the main constraint.

The remainder of the manuscript is organized as follows. Sec. 4.2 presents the camera and communication models. In Sec. 4.3, the cell-free connectivity is introduced while Sec. 4.4 unveils the two optimization problems. Sec. 4.5 subsequently focuses on the solution of those problems while numerical results are discussed in Sec. 4.6. Concluding remarks are provided in Sec. 4.7.

4.2 System Model

The system under consideration features M UAVs, each equipped with a video camera. The mth UAV is located at $\boldsymbol{q}_m^{(n)} = \left(x_m^{(n)}, y_m^{(n)}, h_m^{(n)}\right)$, where the height $h_m^{(n)}$ is the distance from the focal point of its camera's lens to the ground, n is the time index, and the duration of each time slot is δ. The UAVs are served by L APs with locations $\boldsymbol{q}_\ell = (x_\ell, y_\ell, h_\ell)$. Additionally, the system contains C UAV charging stations located at $\boldsymbol{c}_c = (\mathrm{x}_c, \mathrm{y}_c, \mathrm{h}_c)$.

4.2.1 Camera Model

Consider a planar environment $\mathcal{F} \in \mathbb{R}^2$ and define the field of view (FoV) as the area projected over $\mathcal{F}$ that a camera captures. For a UAV with a downward-facing camera, a rectangular FoV $\mathcal{B}_m^{(n)}$ defined as

$$\mathcal{B}_m^{(n)} = \left\{\boldsymbol{v} = (v_x, v_y) : |x_m^{(n)} - v_x| \leq h_m^{(n)} \tan(\alpha_1) \text{ and } |y_m^{(n)} - v_y| \leq h_m^{(n)} \tan(\alpha_2)\right\}, \quad (4.1)$$

where α_1 and α_2 represent the two halfview angles associated with the perpendicular edges of a rectangle (see Fig. 4.1). The notion of *area per pixel* expresses the tradeoff between the quality of the image and the dimension of the FoV: higher-resolution images correspond to smaller FoVs, and vice versa. From classical optics, the area per pixel is [131]

$$f\left(\boldsymbol{q}_m^{(n)}, \boldsymbol{v}\right) = \begin{cases} a\left(b - h_m^{(n)}\right)^2 & \boldsymbol{v} \in \mathcal{B}_m^{(n)} \\ \infty & \text{otherwise} \end{cases}, \quad (4.2)$$

where b and a depend on the camera capabilities, representing the focal length of the lens and the area of a pixel on the lens divided by b^2, respectively, while $\boldsymbol{v}$ represents the position on $\mathcal{F}$. At higher altitudes, $f(\cdot)$ is larger because each pixel covers a bigger area on the FoV, and the resolution is lower. At lower altitudes, it is the other way around. Outside the FoV,

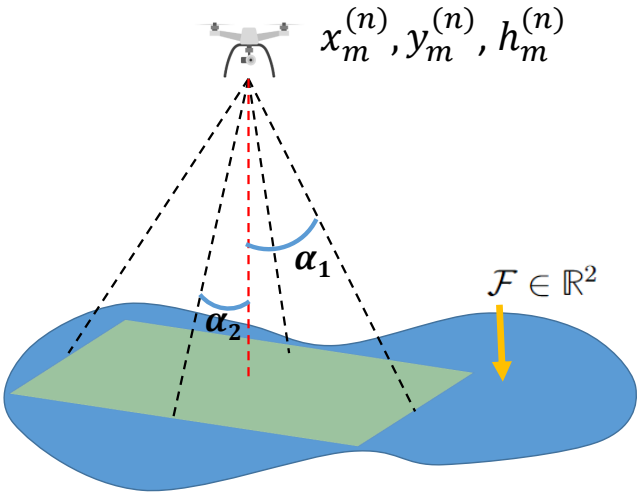


Figure 4.1: Rectangular FoV of the mth UAV for given α_1 and α_2 over a planar region $\mathcal{F}$.

by definition, there are no pixels and $f(\cdot)$ is therefore infinite.

The camera's sensor consists of I equispaced pixels. From sheer geometry,

$$I = \frac{4\left(b - h_m^{(n)}\right)^2 \tan(\alpha_1)\tan(\alpha_2)}{f\left(\boldsymbol{q}_m^{(n)}, \boldsymbol{v}\right)}. \tag{4.3}$$

For a 24-bit RGB color system, the number of bits to represent a picture is $24I$ and, compressing the image with a ratio of ρ, the number of bits to be transmitted per image is

$$B = 24I\rho. \tag{4.4}$$

4.2.2 Channel Model

Upon imaging the environment, UAVs establish wireless connections with the APs to convey the captured images. The air-to-ground channel coefficient between the mth UAV and the

ℓth AP is denoted by $g_{m,\ell}^{(n)}$, drawn from a Rician distribution such that [86, Sec. 3.4.1]

$$g_{m,\ell}^{(n)} = \sqrt{\frac{\beta_0}{\left(d_{m,\ell}^{(n)}\right)^{\kappa}\left(K_{m,\ell}^{(n)}+1\right)}}\left(\sqrt{K_{m,\ell}^{(n)}}e^{j\psi_{m,\ell}^{(n)}} + a_{m,\ell}^{(n)}\right), \tag{4.5}$$

where β_0 and κ are, respectively, the pathloss at a reference distance of 1 m and the pathloss exponent; in turn, the distance is $d_{m,\ell}^{(n)}$ and the Rician factor is

$$K_{m,\ell}^{(n)} = A_1 \exp\left(A_2 \arcsin\left(\frac{h_\ell - h_m^{(n)}}{d_{m,\ell}^{(n)}}\right)\right) \tag{4.6}$$

for environment-dependent parameters A_1 and A_2 [73]. The phase of the LoS component, $\psi_{m,\ell}^{(n)}$, is uniformly random to reflect drifting [41, 42] whereas the small-scale fading is

$$a_{m,\ell}^{(n)} \sim \mathcal{N}_{\mathbb{C}}(0,1). \tag{4.7}$$

Hence, the channel power gain is given by

$$r_{m,\ell}^{(n)} = \frac{\beta_0}{\left(d_{m,\ell}^{(n)}\right)^{\kappa}}. \tag{4.8}$$

4.2.3 Channel Acquisition

Unlike most of the UAV-related literature, this chapter incorporates imperfect channel estimates to enhance the model's realism. Given that the system under consideration features a small number of UAVs, orthogonal pilots can be allocated to each of those UAVs while keeping the overhead at bay; pilot contamination is thus not an issue. After observing the length-τ pilot sequence transmitted by the mth UAV, the ℓth AP can compute the MMSE channel estimate $\hat{g}_{m,\ell}^{(n)}$, which satisfies

$$g_{m,\ell}^{(n)} = \hat{g}_{m,\ell}^{(n)} + \tilde{g}_{m,\ell}^{(n)} \tag{4.9}$$

such that $\hat{g}_{m,\ell}^{(n)}$ is zero-mean with [88, 132]

$$\gamma_{m,\ell}^{(n)} = \mathbb{E}\{|\hat{g}_{m,\ell}^{(n)}|^2\} \tag{4.10}$$

$$= \frac{\left(r_{m,\ell}^{(n)}\right)^2}{r_{m,\ell}^{(n)} + \frac{\sigma^2}{p^{\mathrm{t}}\tau}}, \tag{4.11}$$

given p^{t} as the pilot transmit power and σ^2 as the noise power at the receiver. Finally, the estimation error $\tilde{g}_{m,\ell}^{(n)}$ is zero-mean with variance

$$c_{m,\ell}^{(n)} = r_{m,\ell}^{(n)} - \gamma_{m,\ell}^{(n)}. \tag{4.12}$$

4.2.4 Energy Consumption Model

UAVs are governed by certain dynamics relating location, speed, and acceleration. A first-order Taylor expansion yields [45],[46]

$$\boldsymbol{q}_m^{(n+1)} = \boldsymbol{q}_m^{(n)} + \boldsymbol{v}_m^{(n)}\delta + \frac{1}{2}\boldsymbol{a}_m^{(n)}\delta^2, \tag{4.13}$$

and

$$\boldsymbol{v}_m^{(n+1)} = \boldsymbol{v}_m^{(n)} + \boldsymbol{a}_m^{(n)}\delta, \tag{4.14}$$

where $\boldsymbol{v}_m^{(n)}$ and $\boldsymbol{a}_m^{(n)}$ denote the speed and acceleration vectors. For a quad-rotor UAV moving in 3D with a given climb angle $\tau_m^{(n)}$, and denoting the magnitude of the speed vector by $v_m^{(n)}$, i.e., $v_m^{(n)} = \|\boldsymbol{v}_m^{(n)}\|$, the power consumption can be modeled as [27, 126]

$$P_m^{(n)} = P_0\left(1 + \frac{3v_m^{(n)2}}{U_{\mathrm{tip}}^2}\right) + P_i\left(\sqrt{1 + \frac{v_m^{(n)4}}{4v_0^4}} - \frac{v_m^{(n)2}}{2v_0^2}\right)^{1/2}$$
$$+ \frac{1}{2}d_0\iota s A v_m^{(n)3} + mgv_m^{(n)}\sin\tau_m^{(n)}, \tag{4.15}$$

where P_0 and P_i represent the blade profile and induced powers, respectively, U_{tip} is the tip speed of the rotor blade, and v_0 is the mean rotor induced velocity. In addition, d_0 and s are the fuselage drag ratio and rotor solidity, respectively, while ι denotes the air density and A the rotor disc area.

As advanced, the UAV's mission is divided into tracking and charging stages. The switching occurs at $n = n_m$, once the UAV's energy falls below a threshold that is a function of the distance between the UAV and the closest charging point (see Sec. 4.6). Consequently, the UAV needs to head for a charging station and the remaining energy, denoted by $E_m^{(n_m)}$, must satisfy

$$E_m^{(n_m)} - \sum_{n=n_m}^{n_m+N_{c_m}} P_m^{(n)}\delta \geq 0, \tag{4.16}$$

to avoid running out of energy, where N_{c_m} is the number of slots it takes for the mth UAV to reach charging station $\boldsymbol{c}_c$.

Other constraints include the maximum and minimum UAV altitudes, a maximum velocity and acceleration, and the avoidance of collisions:

$$h_{\min} \leq h_m^{(n)} \leq h_{\max} \tag{4.17}$$

$$v_m^{(n+1)} \leq V_{\max}\delta \tag{4.18}$$

$$\|\boldsymbol{v}_m^{(n+1)} - \boldsymbol{v}_m^{(n)}\| \leq A_{\max}\delta \tag{4.19}$$

$$\mathrm{d}_{m,j}^{(n)} \geq D_{\text{safe}}, \tag{4.20}$$

where $V_{\max}$ and D_{safe} are the maximum UAV velocity and minimum safety distance among UAVs, respectively, $\mathrm{d}_{m,j}^{(n)}$ is the distance between the mth and jth UAVs, and $A_{\max}$ is the maximum acceleration.

4.3 Cell-free Connectivity

On a given time-frequency resource unit, the number of active APs is $L^{(n)}$, which is a function of time. The uplink channel matrix is then

$$\boldsymbol{G}^{(n)} = \left(\boldsymbol{g}_1^{(n)}, \ldots, \boldsymbol{g}_M^{(n)} \right), \tag{4.21}$$

where $\boldsymbol{g}_m^{(n)} \in \mathbb{C}^{L^{(n)} \times 1}$ is the channel between the mth UAV and the active APs, satisfying

$$\boldsymbol{G}^{(n)} = \hat{\boldsymbol{G}}^{(n)} + \tilde{\boldsymbol{G}}^{(n)}, \tag{4.22}$$

where $\hat{\boldsymbol{G}}^{(n)}$ and $\tilde{\boldsymbol{G}}^{(n)}$ are the channel estimate and error matrices, respectively. In general, subsets of APs can provide service to subsets of UAVs [9,13,36], yet this work considers full connectivity, for the sake of the simplicity of notation, pooling the observations from the $L^{(n)}$ APs into

$$\boldsymbol{y}^{(n)} = \boldsymbol{G}^{(n)} \boldsymbol{x}^{(n)} + \boldsymbol{n}^{(n)} \tag{4.23}$$

$$= \underbrace{\hat{\boldsymbol{G}}^{(n)} \boldsymbol{x}^{(n)}}_{\text{signal}} + \underbrace{\tilde{\boldsymbol{G}}^{(n)} \boldsymbol{x}^{(n)} + \boldsymbol{n}^{(n)}}_{\text{effective noise: } \boldsymbol{n}_{\text{e}}^{(n)}}, \tag{4.24}$$

where

$$\boldsymbol{x}^{(n)} = \left(\sqrt{p_1^{(n)}} s_1^{(n)}, \ldots, \sqrt{p_M^{(n)}} s_M^{(n)} \right)^{\mathrm{T}}, \tag{4.25}$$

with unit power symbols $s_m^{(n)}$ while $p_m^{(n)}$ denotes the transmit powers and the noise is $\boldsymbol{n}^{(n)} \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \sigma^2 \boldsymbol{I})$. The effective noise $\boldsymbol{n}_{\mathrm{e}}^{(n)}$ has covariance $\boldsymbol{\Sigma}^{(n)} = \boldsymbol{D}^{(n)} + \sigma^2 \boldsymbol{I}$ given

$$\boldsymbol{D}^{(n)} = \mathbb{E}\Big\{ \big(\tilde{\boldsymbol{G}}^{(n)} \boldsymbol{x}^{(n)}\big) \big(\tilde{\boldsymbol{G}}^{(n)} \boldsymbol{x}^{(n)}\big)^* \Big\} \tag{4.26}$$

$$= \mathrm{diag}\left\{ \sum_{m=1}^{M} c_{m,1}^{(n)} p_m^{(n)}, \ldots, \sum_{m=1}^{M} c_{m,L}^{(n)} p_m^{(n)} \right\}. \tag{4.27}$$

Upon observing $\boldsymbol{y}^{(n)}$, the combiner that maximizes the signal-to-interference-plus-noise ratio (SINR) is the MMSE filter, achieving a value of [36]

$$\mathrm{SINR}_m^{(n)} = \hat{\boldsymbol{g}}_m^{(n)*} \left(\sum_{n \neq m} \hat{\boldsymbol{g}}_n^{(n)} \hat{\boldsymbol{g}}_n^{(n)*} p_n^{(n)} + \boldsymbol{\Sigma}^{(n)} \right)^{-1} \hat{\boldsymbol{g}}_m^{(n)} \, p_m^{(n)}, \tag{4.28}$$

giving a spectral efficiency of

$$\mathrm{SE}_m^{(n)} = \left(1 - \frac{\tau}{\tau_{\mathrm{c}}} \right) \mathbb{E}\{ \log_2(1 + \mathrm{SINR}_m^{(n)}) \}, \tag{4.29}$$

where τ / τ_{c} accounts for the pilot overhead and τ_c denotes the total number of resource units within a coherence block. Random matrix theory offers a way to circumvent the numerical evaluations of the above expectation, providing stable spectral efficiency forms that depend solely on large-scale parameters.

4.3.1 Large-Dimensional Analysis

When evaluating (4.29) for $M, L^{(n)} \to \infty$, convergence to nonrandom limits is assured provided that

$$\boldsymbol{\Gamma}_m^{(n)} = \mathbb{E}\Big\{ \hat{\boldsymbol{g}}_m^{(n)} \hat{\boldsymbol{g}}_m^{(n)*} \Big\} \tag{4.30}$$

$$= \mathrm{diag}\big\{ \gamma_{m,\ell}^{(n)} \; \forall \ell \big\} \tag{4.31}$$

satisfies some technical conditions. Specifically, the inverse of the resolvent matrix in (4.28) must exist, which is ensured by the presence of $\mathbf{\Sigma}^{(n)}$, while $\mathbf{\Gamma}_m^{(n)}$ must have uniformly bounded spectral norm, meaning that the received power does not concentrate on a subset of dimensions as the network grows large.

Theorem 5. *For $M, L^{(n)} \to \infty$ with MMSE reception, $\mathrm{SINR}_m^{(n)} - \overline{\mathrm{SINR}}_m^{(n)} \to 0$ almost surely (a.s.) with*

$$\overline{\mathrm{SINR}}_m^{(n)} = \sum_{\ell=1}^{L^{(n)}} \frac{\gamma_{m,\ell}^{(n)} p_m^{(n)}}{\sum\limits_{i \neq m} \frac{\gamma_{i,\ell}^{(n)}}{1+e_i} p_i^{(n)} + \sum\limits_{\forall i} c_{i,\ell}^{(n)} p_i^{(n)} + \sigma^2}. \tag{4.32}$$

The coefficients e_j are obtained iteratively with $e_j = \lim_{s\to\infty} e_j^{(s)}$, $e_j^{(0)} = L^{(n)}$, and

$$e_j^{(s)} = p_j^{(n)} \operatorname{tr}\left[\mathbf{\Gamma}_j^{(n)} \left(\sum_{i \neq j}^{M} \frac{\mathbf{\Gamma}_i^{(n)}}{1 + e_i^{(s-1)}} \, p_i^{(n)} + \mathbf{\Sigma}^{(n)} \right)^{-1} \right]. \tag{4.33}$$

Proof. The proof follows a similar procedure as the one described in Appendix A.1. □

As anticipated, (4.32) is a deterministic quantity that depends only on large-scale parameters. From the continuous mapping theorem [89], it follows that

$$\mathrm{SE}_m^{(n)} - \overline{\mathrm{SE}}_m^{(n)} \xrightarrow{a.s.} 0, \tag{4.34}$$

where

$$\overline{\mathrm{SE}}_m^{(n)} = \left(1 - \frac{\tau}{\tau_\mathrm{c}}\right) \log_2\left(1 + \overline{\mathrm{SINR}}_m^{(n)}\right) \tag{4.35}$$

is the asymptotic spectral efficiency.

4.4 Problem Formulation

Let us now turn to optimizing the UAV trajectories and the transmit powers. In the tracking stage, the UAVs in the set $\mathcal{S}_1^{(n)}$ have sufficient energy to track while, in the charging stage, the UAVs in the set $\mathcal{S}_2^{(n)}$ attempt to reach one of the C charging points as expeditiously as possible. Given the different nature of the two stages, separate optimizations are devised.

4.4.1 Tracking Stage

During the tracking stage, UAVs periodically capture images, to be forwarded to the APs every N slots. To keep the information fresh on average, the received image bit content must satisfy

$$\delta W \overline{\mathrm{SE}}_m^{(n)} \geq \frac{B}{N} \tag{4.36}$$

where B is given in (4.4) and W is the transmission bandwidth. Note that this conservatively assumes that successive transmissions involve completely new images.

With (4.36) as a constraint that adds to those on altitude, speed, and collision-avoidance, the optimization of trajectories and powers can be cast as the minimization of (a weighted version of) the area per pixel. As discussed in Sec. 4.2.1, the area per pixel reflects the tradeoff between image quality and FoV. Before proceeding, though, the definition of area per pixel must be extended from a single UAV to multiple UAVs, as their FoV may intersect. If multiple cameras cover a certain point $\boldsymbol{v}$, the area per pixel is [131]

$$\tilde{f}\Big(\boldsymbol{q}_m^{(n)} : m \in \mathcal{S}_1^{(n)}, \boldsymbol{v}\Big) = \frac{1}{\sum_{i \in \mathcal{N}_{\boldsymbol{v}}} f\big(\boldsymbol{q}_i^{(n)}, \boldsymbol{v}\big)^{-1} + \Delta}, \tag{4.37}$$

where $\mathcal{N}_{\boldsymbol{v}}$ is the set of UAVs whose cameras cover $\boldsymbol{v}$ while Δ is a regularization factor that

prevents the function from diverging if none of the UAVs covers $\boldsymbol{v}$. Therefore, the joint optimization of the UAVs in tracking mode emerges as

$$\begin{aligned} &\min_{\boldsymbol{q}_m^{(n)}, p_m^{(n)}} \int_{\mathcal{F}} \tilde{f}\Big(\boldsymbol{q}_m^{(n)} : m \in \mathcal{S}_1^{(n)}, \boldsymbol{v}\Big)\psi(\boldsymbol{v}, n)d\boldsymbol{v}, \\ &\text{s.t.} \quad p_m^{(n)} \leq p_{\max} \\ &\qquad (4.13), (4.14), (4.17)\text{-}(4.20), (4.36) \end{aligned} \tag{4.38}$$

where $\psi : \mathbb{R}^2 \times \mathbb{N} \to [0, \infty)$ is a density that weights the points on the FoV, assigning them relative importance (see Sec. 4.4.3). As reported in Sec. 4.6, the solution to the above problem also maximizes the coverage, defined as the ratio between the points with positive density that lie inside the UAVs' FoV and the total number of points with positive density:

$$\text{Coverage}^{(n)} = \frac{\int_{\mathcal{F}} \mathbb{I}\{\psi(\boldsymbol{v}, n) > 0\}\, \mathbb{I}\Big\{\exists \mathcal{B}_m^{(n)} : \boldsymbol{v} \in \mathcal{B}_m^{(n)}\Big\}\, d\boldsymbol{v}}{\int_{\mathcal{F}} \mathbb{I}\{\psi(\boldsymbol{v}, n) > 0\}\, d\boldsymbol{v}}, \tag{4.39}$$

where $\mathbb{I}\{\cdot\}$ is an indicator function whose value is one when the argument's condition is true and zero otherwise.

4.4.2 Charging Stage

During the charging stage, a UAV tries to reach one of the C charging stations. For the mth UAV, the closest one is

$$c_m = \arg\min_c \|\boldsymbol{q}_m^{(n_m)} - \boldsymbol{c}_c\|, \tag{4.40}$$

where n_m, recall, is the time slot in which the mth UAV switches from tracking to charging. In an unconstrained setup, trivial straight-line trajectories would be optimal, but the actual solution is subject to constraints. In particular, choosing to minimize the energy

consumption, the UAV trajectories are optimized via

$$\min_{\boldsymbol{q}_m^{(n)}, N_{c_m}} \sum_{n=n_m}^{n_m+N_{c_m}} P_m^{(n)}\delta, \tag{4.41}$$
$$\text{s.t.} \quad (4.13), (4.14), (4.16)\text{-}(4.20)$$

where N_{c_m} is introduced in Sec. 4.2.4, $P_m^{(n)}$ is defined in (4.15), and the constraints ensure sufficient energy to reach a charging point, the avoidance of collisions, and the respect of altitude and velocity limits. The switching between tracking and charging, recall, is controlled by the remaining energy at the UAV, as furthered in Sec. 4.6.

4.4.3 Fire Dynamics

The density function $\psi(\boldsymbol{v}, n)$ depends on the event being tracked and can be generated either synthetically through a model or constructed from actual data. While the solution presented in the next section is valid for any generic density function, for wildfires this function should encode certain features that model the spread of a fire. Wildfire simulation is an active field of research in itself [109]. FARSITE, an established model employed by government agencies, is adopted here [110]. It dictates that the spread of every ignition originally follows an ellipse. Subsequently, the points on the ellipse serve as new fronts as per Huygens principle [110], each such front point growing as a new ellipse. The set of new front points is defined by the combination of the local ellipses, i.e., their convex hull [110, Fig. 1]. Because of the different local weather conditions, fuels, or terrain, the dimensions of the local ellipses can be different and result in a new front that is no longer elliptic. Let us define $U^{(n)}$ [m/s] and $\theta_{\text{wind}}^{(n)}$ as the midflame wind's speed and direction, respectively, modelled as

$$U^{(n)} \sim |\mathcal{N}(U_0, \sigma_U)| \tag{4.42}$$
$$\theta_{\text{wind}}^{(n)} \sim \mathcal{N}(\overline{\theta}_{\text{wind}}, \sigma_{\text{wind}}). \tag{4.43}$$

Generating the ellipse corresponding to each front point requires its minor and major axes, $2a^{(n)}$ and $2b^{(n)}$, respectively, with

$$a^{(n)} = \frac{1}{2\,\mathsf{LB}^{(n)}}\left(\mathsf{R} + \frac{\mathsf{R}}{\mathsf{HB}^{(n)}}\right) \tag{4.44}$$

$$b^{(n)} = \frac{1}{2}\left(\mathsf{R} + \frac{\mathsf{R}}{\mathsf{HB}^{(n)}}\right), \tag{4.45}$$

where R [m/min] is the fire's steady-state spreading rate and

$$\mathsf{LB}^{(n)} = 0.936\, e^{0.2566U^{(n)}} + 0.461\, e^{-0.1548U^{(n)}} - 0.397 \tag{4.46}$$

$$\mathsf{HB}^{(n)} = \frac{\mathsf{LB}^{(n)} + \sqrt{[\mathsf{LB}^{(n)}]^2 - 1}}{\mathsf{LB}^{(n)} - \sqrt{[\mathsf{LB}^{(n)}]^2 - 1}}. \tag{4.47}$$

Hence, at time $n+1$, the ith front point, denoted by $\boldsymbol{z}_i^{(n)} \in \mathbb{R}^2$, generates the ellipse

$$\boldsymbol{z}_i^{(n)} + \delta \begin{bmatrix} c_x^{(n)} \sin\theta_{\text{wind}}^{(n)} + a^{(n)} \cos\omega \\ c_y^{(n)} \cos\theta_{\text{wind}}^{(n)} + b^{(n)} \sin\omega \end{bmatrix}, \tag{4.48}$$

where $0 \leq \omega \leq 2\pi$ whereas $c_x^{(n)}$ and $c_y^{(n)}$ represent the fire spreading gradients, respectively. As the exact FARSITE implementation requires detailed information about the environment, and with a view to a general formulation, we consider a simplified set of FARSITE parameters that has been widely used in the literature [111, 112]. In this simplified model [110]

$$c_x^{(n)} = c_y^{(n)} = \frac{\mathsf{R}}{2}\left(1 - \frac{1}{\mathsf{HB}^{(n)}}\right). \tag{4.49}$$

With that, a 2D histogram can be produced to serve as the density function $\psi(\boldsymbol{v}, n)$, assigning a nonzero weight to the points of the fire's perimeter being tracked. Fig. 4.2 illustrates the expansion of a wildfire from an ignition point (black dot) using FARSITE. Assuming wind in the direction of the red arrow, an ellipse with minor and major axes, $2a^{(n)}$ and $2b^{(n)}$,

respectively, represents the next fire front as shown in Fig. 4.2b. Then, each blue point in the new fire front acts like a new ignition point and the convex hull of the corresponding ellipses generates the new fire front in Fig. 4.2d. Here, the density function $\psi(\boldsymbol{v}, n)$ corresponds to the perimeter, represented by the red curve.

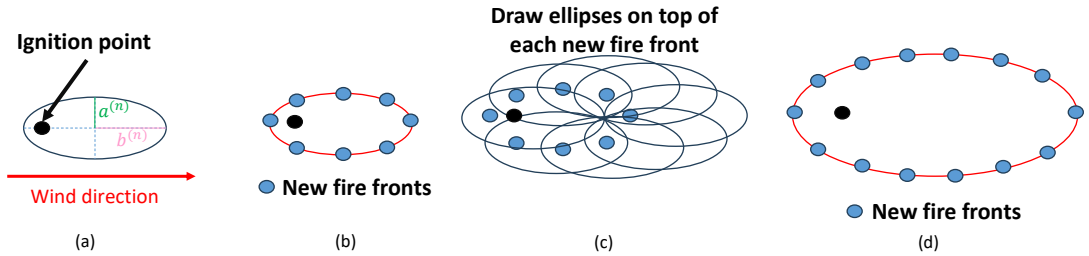


Figure 4.2: Summary of FARSITE fire propagation model.

4.5 Proposed Solution

4.5.1 DQL Fundamentals

Solving (4.38) and (4.41) is challenging due to (*i*) the variety of constraints, (*ii*) the lack of convexity with respect to the optimization variables, and (*iii*) the time-varying nature of the problem. However, the optimization can be formulated as a Markov decision process (MDP) problem, for which RL is an appropriate solution. In general, an MDP is defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, R, \gamma)$, where $\mathcal{S}$ and $\mathcal{A}$ are the state and action spaces, respectively. In turn, $\mathcal{P}$ denotes the state transition probability with $P\big(s^{(n+1)}|s^{(n)}, a^{(n)}\big)$ being the probability of transitioning from state $s^{(n)}$ to $s^{(n+1)}$ after action $a^{(n)}$. Finally, R is the reward function and $\gamma \in [0, 1]$ denotes the discount factor in such function. According to the MDP,

$$P\big(s^{(n+1)}|s^{(0)}, a^{(0)}, \ldots, s^{(n)}, a^{(n)}\big) = P\big(s^{(n+1)}|s^{(n)}, a^{(n)}\big)$$

and

$$R\big(s^{(n+1)}|s^{(0)},a^{(0)},\ldots,s^{(n)},a^{(n)}\big) = R\big(s^{(n+1)}|s^{(n)},a^{(n)}\big) \\ = r\big(s^{(n)},a^{(n)}\big)\,, \tag{4.50}$$

where $r(\cdot)$ is introduced as shorthand notation. The goal in RL is to find a policy $\pi : \mathcal{S} \to \mathcal{A}$ that maximizes the average discounted reward,

$$\pi^* = \arg\max_{\pi} \mathbb{E}\left\{\sum_{j=0}^{\infty} \gamma^j r\big(s^{(j)},a^{(j)}\big)\,\middle|\,\pi\right\}, \tag{4.51}$$

with expectation over $s^{(0)} \sim P\big(s^{(0)}\big)$, $a^{(n)} \sim \pi\big(\cdot|s^{(n)}\big)$ and $s^{(n+1)} \sim P\big(s^{(n+1)}|s^{(n)},a^{(n)}\big)$. In RL, the Q-function measures the expected future reward when the system is in state s, performs action a, and follows policy π, namely

$$Q^{\pi}(s,a) = \mathbb{E}_{\pi}\left\{\sum_{j=0}^{\infty} \gamma^j r\left(s^{(j')},a^{(j')}\right)\,\middle|\,s^{(n)}=s, a^{(n)}=a\right\}. \tag{4.52}$$

where $j' = j+n+1$. An interesting property of the Q-function is that every optimum policy π^* achieves the optimum Q-function, i.e.,

$$Q^{\pi^*}(s,a) = Q^*(s,a), \tag{4.53}$$

where $*$ denotes optimality. Hence, one can identify the optimum policy through the Q-function. In fact, through dynamic programming and the Bellman equation, learning the Q-function is possible. Under the temporal difference (TD) learning solution, which combines the current estimate of the Q-function with samples obtained from the environment, the estimates can be iteratively updated following [47]

$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha\left[r(s,a) + \gamma \max_{a'} Q(s',a')\right]. \tag{4.54}$$

Although (4.54) provides a straightforward rule to learn the Q-function, the corresponding learning process is highly dependent on the state and action space dimensions, as every pair (s, a) needs to be explored many times. Multi-UAV communication problems, with their complex and dynamic environments, and high-dimensional state and action spaces, motivate the use of DQL. This leverages deep neural networks to approximate the Q-value function, with a slight abuse of the notation, as $Q(s, a) \approx Q(s, a; \boldsymbol{\theta})$ where $\boldsymbol{\theta}$ are the neural network parameters tuned to minimize

$$L(\boldsymbol{\theta}) = \mathbb{E}\left\{|y(s, a) - Q(s, a; \boldsymbol{\theta})|^2\right\}, \tag{4.55}$$

where $y(s, a)$ is the target value defined as

$$y(s, a) = r(s, a) + \gamma \max_{a'} Q(s', a'; \boldsymbol{\theta}). \tag{4.56}$$

Still, DQL suffers from a major limitation as it considers a discrete action space. To consider continuous actions, we adopt TD3, a policy-based model-free algorithm [48]. TD3 leverages an actor-critic architecture, with both modules composed by neural networks. On the one hand, the critic aims at learning the Q-function, which is exploited by the actor to know how beneficial an action is. On the other hand, the actor's role is to learn a policy that, as opposed to traditional Q-learning, outputs continuous action values. Such policy is parameterized by $\boldsymbol{\phi}$, the actor's neural network parameters, that are updated by the gradient computed as

$$\nabla J(\boldsymbol{\phi}) = \mathbb{E}\left\{\nabla_{\boldsymbol{\phi}} \pi_{\boldsymbol{\phi}}(s)\ \nabla_a Q(s, a; \boldsymbol{\theta})|_{a=\pi_{\boldsymbol{\phi}}(s)}\right\}, \tag{4.57}$$

where

$$J(\boldsymbol{\phi}) = \mathbb{E}_{\pi}\left\{\sum_{j=0}^{\infty} \gamma^j r\left(s^{(j)}, a^{(j)}\right)\right\}. \tag{4.58}$$

In addition, DQL algorithms typically include the so-called target networks. These are copies of the original networks whose parameters remain frozen over a number of iterations. Then, a soft-update rule is applied. Concretely, we denote by $\boldsymbol{\theta}'$ and $\boldsymbol{\phi}'$ the critic and actor neural network target parameters, respectively, with update rules

$$\boldsymbol{\theta}^{'} \leftarrow (1-\tau_{\mathrm{T}})\boldsymbol{\theta}^{'} + \tau_{\mathrm{T}}\boldsymbol{\theta}, \tag{4.59}$$

$$\boldsymbol{\phi}^{'} \leftarrow (1-\tau_{\mathrm{T}})\boldsymbol{\phi}^{'} + \tau_{\mathrm{T}}\boldsymbol{\phi}, \tag{4.60}$$

where τ_{T} controls the memory of the updates. The inclusion of target networks addresses the moving target issue that arises when Q-values and target values are obtained using the same neural network and are constantly changing, creating a moving target for the learning; the algorithm's goal then keeps varying as the agent learns, hampering the finding of a solution.

Relative to previous actor-critic methods, TD3 introduces improvements that enhance the stability during learning:

1. It features two critics, and respective target critic networks. This addresses the overestimation of Q-values caused by the use in previous methods of the maximum action value to approximate the maximum expected action value, as per (4.56). In TD3, the agent selects the Q-target network with smaller Q-value to construct the target value as

$$y(s,a) = r(s,a) + \gamma \min_{i=1,2} Q(s', a'; \boldsymbol{\theta}_i'). \tag{4.61}$$

2. To reduce the accumulation of residual errors, the policy network parameters are updated less frequently than the Q-function parameters. Precisely, [48] suggests a policy update for every two Q-function updates.

3. To prevent overfitting to specific estimated values, TD3 adds clipped random noise to the target actions,

$$\hat{a} = \pi_{\phi'}(s) + \hat{\epsilon}, \tag{4.62}$$

where $\hat{\epsilon} \sim \text{clip}\big(\mathcal{N}(0, \hat{\sigma}_a), -\hat{\epsilon}_{\max}, \hat{\epsilon}_{\max}\big)$, i.e., the values outside the interval $[-\hat{\epsilon}_{\max}, \hat{\epsilon}_{\max}]$ are clipped to the interval edges.

TD3 is applicable to (4.38) and (4.41). Next, we define the states, actions, and rewards for each problem with one agent coordinating a swarm of M UAVs.

4.5.2 TD3: Tracking Stage

To solve (4.38), single-agent or multi-agent algorithms can be employed. A single-agent solution would coordinate the entire swarm and present a significant challenge due to the exponential growth of state and action spaces with the number of UAVs. Alternatively, large MDPs can be factored into simpler ones, leading to simpler agents and distributed solutions [133, 134]. Accordingly, we use the multi-agent solution, where each UAV is controlled by a different agent. The following is needed:

- The state space, which includes information about (*i*) the UAV locations, (*ii*) the portion of the image bit content that has not yet been transmitted, and (*iii*) the state of the fire coverage. To quantify the latter, let us compute the center of mass of the area per pixel as per (4.63). If important points remain uncovered, the agent learns to move in that direction. The untransmitted portion of the image bit content is included in the state space as

$$i_m^{(n)} = \max\left\{0, \left(1 - \frac{\delta W \overline{\text{SE}}_m^{(n)}}{B/N}\right)\right\}. \tag{4.64}$$

$$\left(\mathrm{x}_{\mathrm{c}}^{(n)}, \mathrm{y}_{\mathrm{c}}^{(n)}\right) = \frac{\int_{\mathcal{F}} \boldsymbol{v} \tilde{f}\left(\boldsymbol{q}_m^{(n)} : m \in \mathcal{S}_1^{(n)}, \boldsymbol{v}\right) \psi(\boldsymbol{v}, n) d\boldsymbol{v}}{\int_{\mathcal{F}} \tilde{f}\left(\boldsymbol{q}_m^{(n)} : m \in \mathcal{S}_1^{(n)}, \boldsymbol{v}\right) \psi(\boldsymbol{v}, n) d\boldsymbol{v}} \tag{4.63}$$

Altogether, the state of the mth UAV at time n is

$$s_m^{(n)} = \left\{ \frac{x_m^{(n)}}{\mathrm{S}}, \frac{y_m^{(n)}}{\mathrm{S}}, \frac{h_m^{(n)}}{h_{\max}}, \frac{\boldsymbol{v}_m^{(n)}}{v_{\max}}, \frac{\mathrm{x}_{\mathrm{c}}^{(n)}}{\mathrm{S}}, \frac{\mathrm{y}_{\mathrm{c}}^{(n)}}{\mathrm{S}}, \frac{\mathrm{d}_{m,j}^{(n)}}{C_{\mathrm{d}}}, i_m^{(n)} \right\}, \tag{4.65}$$

where S and C_{d} are positive normalization constants, $\boldsymbol{v}_m^{(n)}$ is the UAV's speed and $\mathrm{d}_{m,j}^{(n)}$ is the distance between the mth and the jth UAVs.

- The action space defines the set of actions that the UAVs can take and consists of variations in the 3D location and transmit power, such that $a_m^{(n)} \in \mathbb{R}^{4\times 1}$. Given the relationships established by (4.13) and (4.14), the problem is solved with respect to the acceleration. The first three actions relate to the acceleration variations in 3D space, each constrained by a maximum and minimum value $[-A_{\max}, A_{\max}]$. The last action component sets the transmit power, adjusted to any value satisfying $0 \leq p_m^{(n)} \leq p_{\max}$.

- The reward function evaluates the quality of the actions and motivates the agent to take those leading to desirable outcomes. Not only does it capture the cost function, but it satisfies the constraints. As per the reward shaping technique, the reward observed by the mth UAV is [135]

$$r_m\left(s_m^{(n)}, a_m^{(n)}\right) = \sum_{w=1}^{5} r_{m,w}\left(s_m^{(n)}, a_m^{(n)}\right), \tag{4.66}$$

whose terms are detailed next. In particular, $r_{m,1}\left(s_m^{(n)}, a_m^{(n)}\right)$ contains the variation in the cost function. After performing action $a_m^{(n)}$, the state transitions to $s_m^{(n+1)}$, whose

cost is defined as

$$c(s_m^{(n+1)}) = \int_{\mathcal{F}} \tilde{f}\Big(\boldsymbol{q}_m^{(n+1)} : m \in \mathcal{S}_1^{(n+1)}, \boldsymbol{v}\Big) \cdot \psi(\boldsymbol{v}, n+1) d\boldsymbol{v}. \tag{4.67}$$

Thus, to motivate the UAVs to reduce the cost function, $r_{m,1}\Big(s_m^{(n)}, a_m^{(n)}\Big)$ is defined as

$$r_{m,1}\big(s_m^{(n)}, a_m^{(n)}\big) = K_{\mathrm{c}} \left(1 - \frac{c(s_m^{(n+1)})}{I_{\mathrm{r}}^{(n)}}\right) \tag{4.68}$$

where $K_{\mathrm{c}} > 0$ and $I_{\mathrm{r}}^{(n)} = \frac{1}{\Delta} \int_{\mathcal{F}} \psi(\boldsymbol{v}, n+1) d\boldsymbol{v}$. Moreover, to avoid collisions among UAVs, we define

$$r_{m,2}\left(s_m^{(n)}, a_m^{(n)}\right) = \begin{cases} -K_{\mathrm{coll}} & \text{if } \exists\ \mathrm{d}_{m,j}^{(n)} \leq D_{\mathrm{safe}} \\ 0 & \text{otherwise} \end{cases}, \tag{4.69}$$

where $K_{\mathrm{coll}} > 0$. Similarly, we define the reward associated with (4.36) as

$$r_{m,3}\big(s_m^{(n)}, a_m^{(n)}\big) = \begin{cases} -K_{\mathrm{f}} & \text{if } \frac{B}{N} \geq \delta W \overline{\mathrm{SE}}_m^{(n)} \\ 0 & \text{otherwise} \end{cases}, \tag{4.70}$$

where $K_{\mathrm{f}} > 0$. A similar expression is used to define $r_{m,4}(\cdot)$ and $r_{m,5}(\cdot)$, which penalizes actions taking the UAV out of limits and exceeding a maximum velocity, respectively, with respective penalties of $-K_{\mathrm{h}}$ and $-K_{\mathrm{v}}$.

The multi-agent TD3 algorithm used to train a single UAV during the tracking stage is summarized in Alg. 3 whereas Fig. 4.3 presents its block diagram. Note that the factorization of the larger problem into M single-agent problems allows for parallelization, leading to faster convergence and more efficient learning. Given that the UAVs in tracking mode have the same objective, their training will converge to the same models. Therefore, training

can be conducted on one UAV, while the others can either follow the learned policy up to that point in time, perform random movements, or remain static. Besides the initial network parameters, Alg. 3 accepts the number of episodes E, which refers to the number of wildfire realizations the system will observe, as input. Additionally, a replay buffer of size $|\mathcal{M}|$ stores transitions of the type $\left\{s_m^{(n)}, a_m^{(n)}, r_m\left(s_m^{(n)}, a_m^{(n)}\right), s_m^{(n+1)}\right\}$. Finally, F refers to the update frequency over the policy and target networks.

4.5.3 TD3: Charging Stage

Since the tracking and charging stages are optimized separately, two independent models can be trained. The same ideas and algorithms described for tracking can be applied for charging, with only some modifications needed to address (4.41). Concretely, during the charging stage, only one UAV needs to be coordinated. Hence, the states consist of

$$s_m^{(n)} = \left\{ \frac{\mathrm{x}_{u_m} - x_m^{(n)}}{\mathrm{S}}, \frac{\mathrm{y}_{u_m} - y_m^{(n)}}{\mathrm{S}}, \frac{\mathrm{h}_{u_m} - h_m^{(n)}}{h_{\max}}, \mathrm{d}_{m,j}^{(n)}, \frac{E_m^{(n)}}{E_0} \right\}, \tag{4.71}$$

where, recall, the distance between the mth and jth UAVs is $\mathrm{d}_{m,j}^{(n)}$ while E_0 is a positive constant normalizing the remaining energy. While $r_{m,2}(\cdot)$, $r_{m,4}(\cdot)$ and $r_{m,5}(\cdot)$ remain the same, $r_{m,3}\left(s_m^{(n)}, a_m^{(n)}\right) = 0$ as the UAVs have no data to forward. The main difference, though, is in the definition of $r_{m,1}\left(s^{(n)}, a^{(n)}\right)$, in (4.72) atop the next page, which includes the variation in the cost function:

- When the charging point is reached, a positive reward K_{fin} is given.
- The term $-K_{\text{e}} P_m^{(n)} \delta$ aims at minimizing the energy whereas the second component motivates the UAV to minimize the distance with the charging point. If it is approached, the UAV receives a positive reward; otherwise, the reward is negative. Finally, a negative reward of K_{en} is incurred if the remaining energy is exhausted.

$$r_{m,1}(s_m^{(n)}, a_m^{(n)}) = \begin{cases} K_{\text{fin}} & \text{if } \boldsymbol{q}_m^{(n)} = \boldsymbol{c}_{c_m} \\ -K_{\text{e}} P_m^{(n)} \delta + K_{\text{d}} \left(\|\boldsymbol{c}_{c_m} - \boldsymbol{q}_m^{(n)}\| - \|\boldsymbol{c}_{c_m} - \boldsymbol{q}_m^{(n+1)}\| \right) & \text{otherwise} \end{cases}. \tag{4.72}$$

Alg. 3 applies to the charging stage with the above modifications in the rewards. In addition, whereas the tracking stage does not have a final state, the training of the charging model does reach a final state whenever $\boldsymbol{q}_m^{(n)} = \boldsymbol{c}_{c_m}$. Thus, the *while* condition in the loop can be changed from $n < N_e$ to $\boldsymbol{q}_m^{(n)} \neq \boldsymbol{c}_{c_m}$.

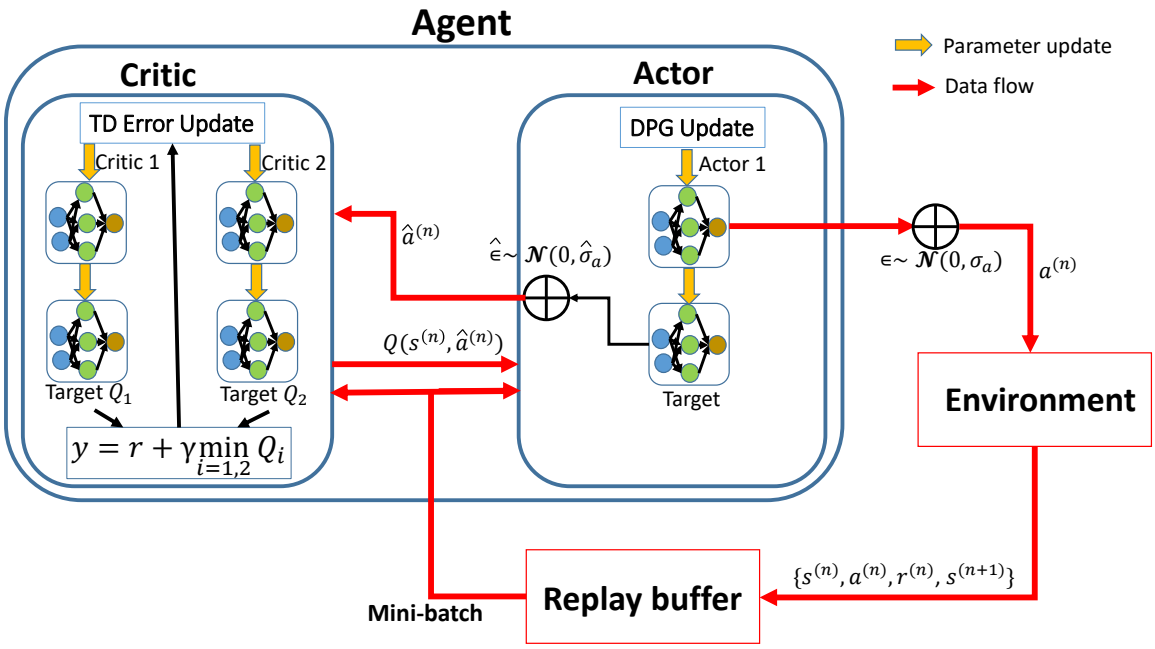


Figure 4.3: TD3 block diagram.

4.6 Numerical Results

For the purpose of performance evaluation, a 300-m^2 environment is considered with the simulation parameters in Table 4.1. The camera parameters are set based on [30, 131], emulating real devices, while the UAV and cell-free parameters are borrowed from [13,33,92].

Algorithm 3 TD3 - Tracking stage

Require: No. of episodes E, initial policy parameters $\boldsymbol{\phi}$, critic parameters $\boldsymbol{\theta}_1$, $\boldsymbol{\theta}_2$, empty memory replay buffer $\mathcal{M}$ and parameter update frequency F.
Initialize $\boldsymbol{\theta}_1^{'} \leftarrow \boldsymbol{\theta}_1$, $\boldsymbol{\theta}_2^{'} \leftarrow \boldsymbol{\theta}_2$, and $\boldsymbol{\phi}^{'} \leftarrow \boldsymbol{\phi}$.
for $e = 1, \ldots, \mathrm{E}$ **do**
Set $n = 0$ and initialize UAV, AP and ignition point locations.
while $n < N_e$ **do**
Obtain the current state $s^{(n)}$.
Perform action $a_m^{(n)} = \text{clip}\left(\pi_{\boldsymbol{\phi}}\left(s_m^{(n)}\right) + \epsilon\right)$, with $\epsilon \sim \mathcal{N}(0, \sigma_a)$, and observe $s_m^{(n+1)}$.
Compute the reward $r\left(s_m^{(n)}, a_m^{(n)}\right)$ as in (4.66).
Store $\left\{s_m^{(n)}, a_m^{(n)}, r_m\left(s_m^{(n)}, a_m^{(n)}\right), s_m^{(n+1)}\right\}$ in $\mathcal{M}$.
Select a minibatch of N_{mem} experiences $\left\{s_m^{(n)}, a_m^{(n)}, r_m\left(s_m^{(n)}, a_m^{(n)}\right), s_m^{(n+1)}\right\}$ from $\mathcal{M}$.
Compute target actions $\hat{a}_m^{(n)}$ as in (4.62).
Compute targets $y(s, a)$ as in (4.56).
Update the critics by minimizing $\mathbb{E}\{|y(s, a) - Q(s, a; \boldsymbol{\theta}_i)|^2\}$ for $i = 1, 2$.
if n mod F **then**
Update $\boldsymbol{\phi}$ via (4.57).
Update target networks $\boldsymbol{\theta}_1^{'}$, $\boldsymbol{\theta}_2^{'}$, and $\boldsymbol{\phi}^{'}$ as in (4.59) and (4.60).
end if
$n = n + 1$.
Optional: move the rest of UAVs.
end while
end for

Aiming at simulating real wildfires, the fire propagation components are chosen from [110–112] whereas the TD3 parameters are configured as suggested in [57,59,129,130]. Particularly, the variance of the noise added to the actions is $\sigma_a = \hat{\sigma}_a = 0.1$, $\hat{\epsilon}_{\max} = 0.5$, and the parameter soft updates are handled with $\tau_{\mathrm{T}} = 0.01$. Also, the learning rate for actor and critics is set to $5 \cdot 10^{-4}$ and $5 \cdot 10^{-3}$, respectively. The normalization constants in the states and rewards are obtained via cross-validation and summarized in Table 4.2; other values could work as well. The initial UAV and AP locations, charging point coordinates, and fire ignition point are randomly generated. At each time slot, the fire perimeter is updated according to Sec. 4.4.3 whereas the UAV locations are updated following the output of the actor block in the trained TD3 algorithm. Finally, given that we solve two separate problems, we first provide training and evaluation results assuming unlimited energy at UAVs. Then, after discussing the charging performance, the combined results of tracking and charging are presented.

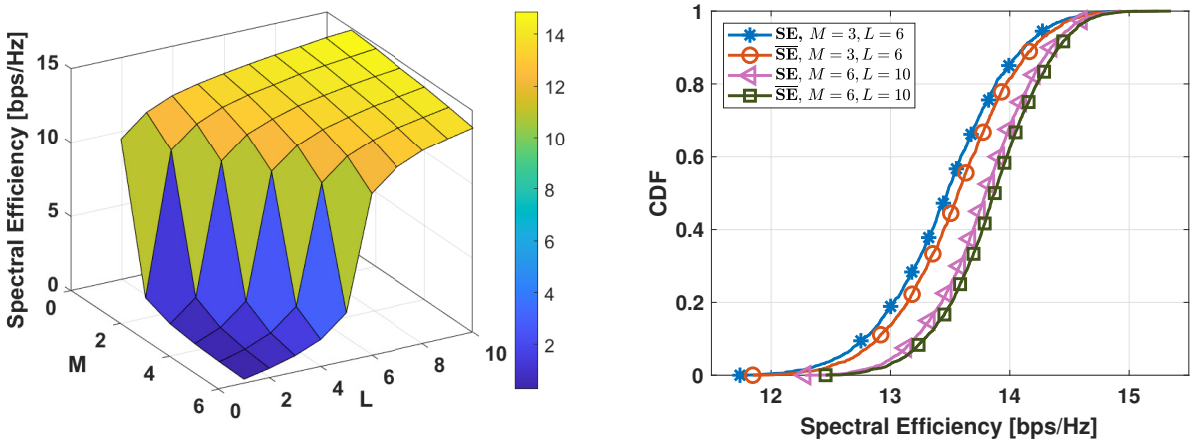
Table 4.1: Simulation parameters

Parameter	Value	Parameter	Value
α_1, α_2	17.5°, 13.125°	$p_{\max}$	100 mW
a, b	10^{-6}, 10	β_0	-30 dB
ρ	0.4	σ^2	-96 dBm
L	10	A_1, A_2	0, 6.4 dB
C	4	Δ	10^{-5}
f_{c}	2.4 GHz	δ	0.5 s
$A_{\max}$	1	B	10 MHz
$V_{\max}$	20	γ	0.85
$H_{\min}$, $H_{\max}$	100 m, 150 m	N	2
τ	200	τ_{c}	6250
U_0	5 m/s	D_{safe}	4 m
σ_U	1 m/s	$\hat{\theta}_{\mathrm{wind}}$	$\mathcal{U}[0, 2\pi]$
R	35 m/min	σ_{wind}	0.1

For starters, let us validate the cell-free asymptotic derivations, as they are leveraged in (4.38). To that end, Fig. 4.4a plots the average user spectral efficiency for different M and L; as one would expect, a smaller M/L, i.e., more APs per UAV, yields a better spectral efficiency. In turn, Fig. 4.4b indicates that Thm. 5 provides an excellent approximation even in low-dimensional systems, say $M = 3$ and $L = 6$.

Table 4.2: State and reward parameters

Description	**Parameter**	**Value**
Normalization in (4.65), (4.71)	S	300
Constant in (4.68)	K_{c}	50
Penalty in (4.69)	K_{coll}	100
Penalty in (4.70)	K_{f}	15
Flying-out-of-limits penalty	K_{h}	60
Normalization in (4.71)	E_0	12000
Rewards and constants in (4.72)	K_{fin},K_{e}, K_{d}	200, 1, 0.1
Out-of-energy penalty	K_{en}	100

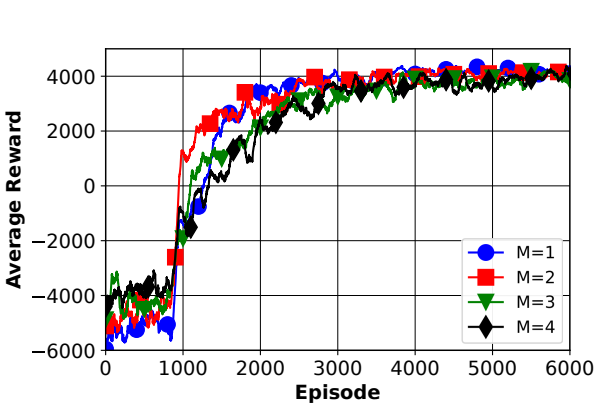


(a) (b)

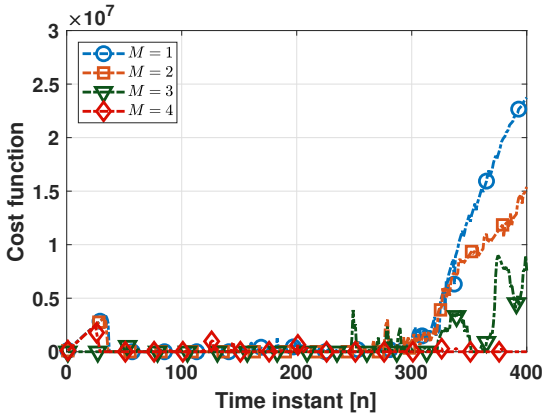
Figure 4.4: (a) Spectral efficiency for different M, L; (b) validation of Theorem 5.

Next, the attention shifts to the tracking stage. Fig. 4.5 assesses the TD3 algorithm's training performance for $M = 1, \ldots, 4$, where a 3-layer feed-forward neural network is used, each layer containing 256 neurons. In the initial 750 episodes, the replay buffer accumulates transitions to ensure that, upon training commencement, every minibatch comprises samples with minimal correlation, enhancing the learning process. Hence, learning truly begins after episode 750, leading to an overall increase in the average reward for all models. Once 3000 episodes are reached, rewards stabilize. For subsequent evaluations, the models saved at the 6000-episode mark are utilized. Fig. 4.5b evaluates the trained models over a single realization of a wildfire with the cost function in (4.38). Despite the random initialization of the UAV locations, the swarm efficiently repositions itself, ensuring full perimeter coverage by $n = 100$. However, as n reaches 300, the perimeter dimensions become too extensive to be managed by $M = 1$ UAV, resulting in a significant cost increase. A similar trend is observed for $M = 2$ UAVs, but $M = 3$ and $M = 4$ UAVs manage to maintain a satisfactory performance. Note how, due to the regularization parameter Δ, the cost function exhibits a high variance. When all points are covered, the function yields a small value. However, if even a single perimeter point remains uncovered, the function experiences sharp spikes. The coverage metric defined in (4.39) then comes handy, as a smoother complement to the cost function, and hence both values are reported in the sequel.

Next, we measure the cumulative distribution function (CDF) of the cost function and fire coverage for $H_{\min} = 125$ and $H_{\max} = 150$ at $n = 350$ over 1,000 independent wildfire events. As one would expect, increasing the number of UAVs results in an increased coverage and a smaller cost function. Particularly, the case with $M = 4$ UAVs achieves a 90% coverage with very high probability. Also, note that both figures consider $n = 350$, i.e., fires that have already expanded and are large. Although not shown for the sake of conciseness, for $n = 100$, meaning for fires that are still small, 90% coverage is achieved regardless of the number of UAVs.

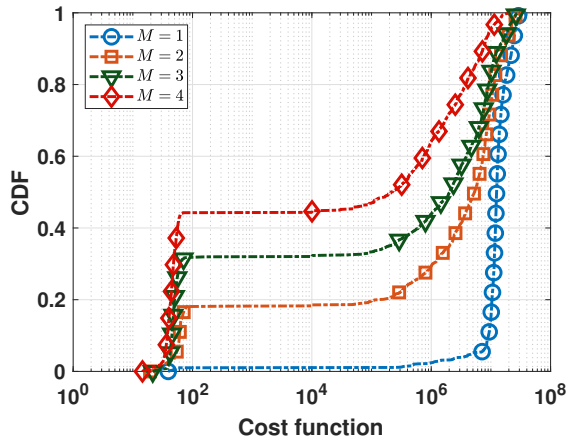


(a)

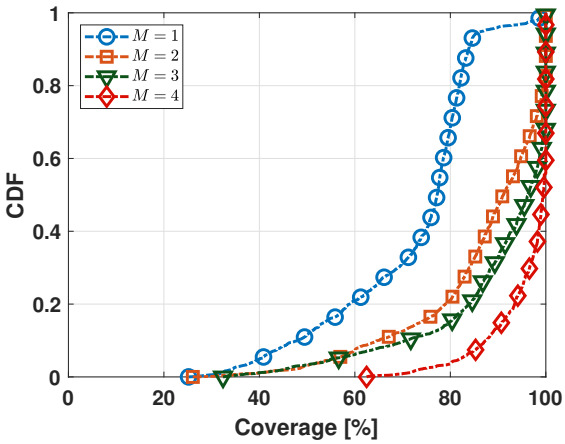


(b)

Figure 4.5: For the tracking problem (a) average training reward; (b) cost function value as a function of n.



(a)



(b)

Figure 4.6: At $n = 350$ for $H_{\min} = 125$ and $H_{\max} = 150$ (a) cost function; (b) coverage.

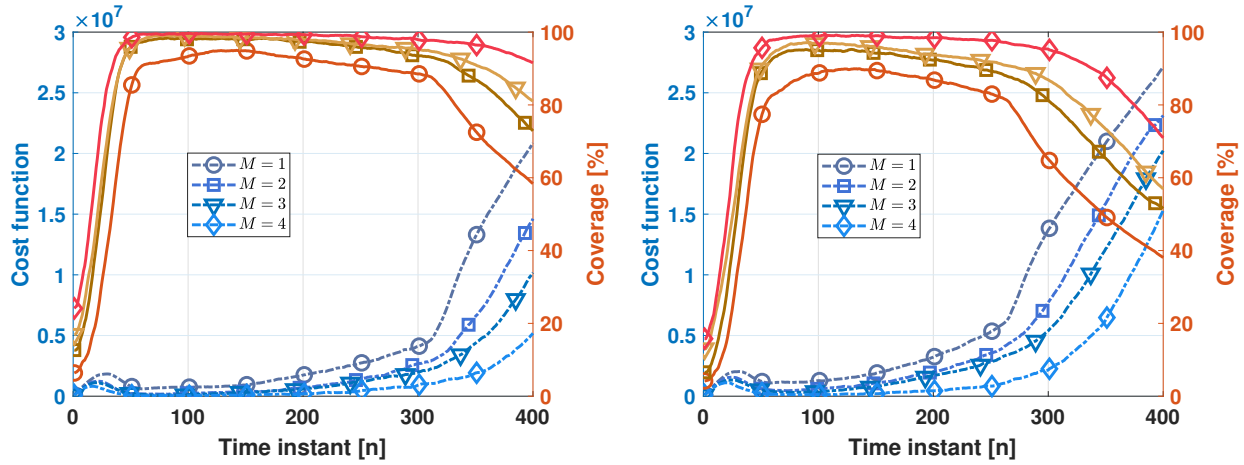
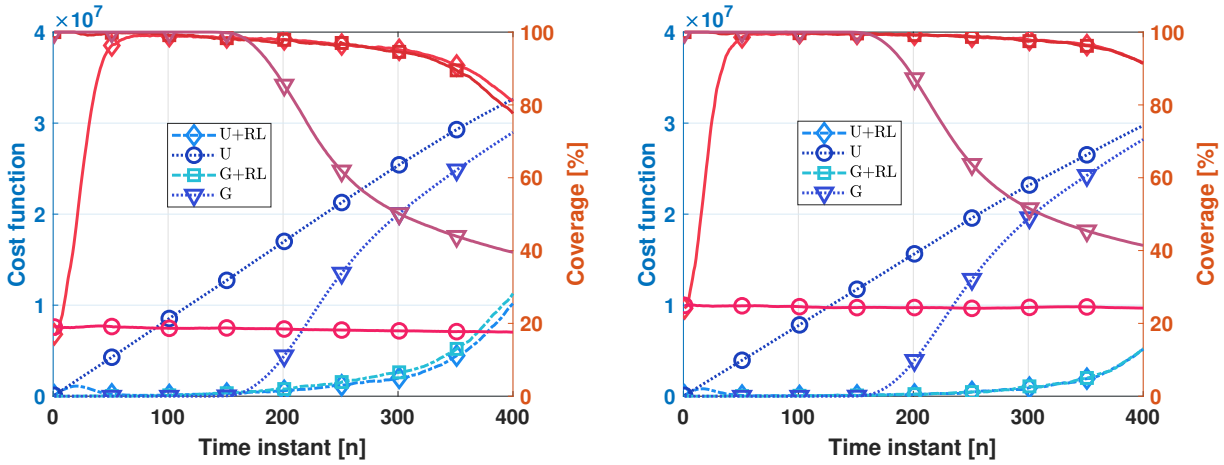


Figure 4.7: Cost and coverage for tracking and $(H_{\min}, H_{\max})$: (a) (125,150) (b) (100,125).

To quantify the impact of the flying altitude on the problem, we include Fig. 4.7. These figures measure the swarm's area per pixel (blue) and coverage (red) over 400 time steps and average the results across 1,000 different fire realizations, considering only the tracking stage where UAVs have unlimited energy. Fig. 4.7a corresponds to $H_{\min} = 125$ and $H_{\max} = 150$, while Fig. 4.7b is for $H_{\min} = 100$ and $H_{\max} = 125$. As expected, reducing the allowed flying altitudes decreases the FoV, leading to a degradation in both the cost function and coverage. This degradation is especially notable when the fire is large, say $n > 300$, whereas, at the onset of the event, lower flying altitudes suffice.

Next, our RL solution is tested against two benchmarks. The first one distributes the UAVs uniformly at random throughout the region and is denoted by U. The second one, denoted by G, places the UAVs around the ignition point following a 2D Gaussian random distribution with covariance $10\boldsymbol{I}$. Against these benchmarks, two initializations are considered for the RL approach, again uniformly at random or Gaussian, denoted respectively by U+RL and G+RL. Fig. 4.8 presents results for $M = 3$ and $M = 4$ with $H_{\min} = 125$ and $H_{\max} = 150$. The following is observed:

- The U method performs poorly. An approximate value for the coverage achieved by



(a) (b)

Figure 4.8: Cost and coverage for tracking under different solutions: (a) $M = 3$ (b) $M = 4$.

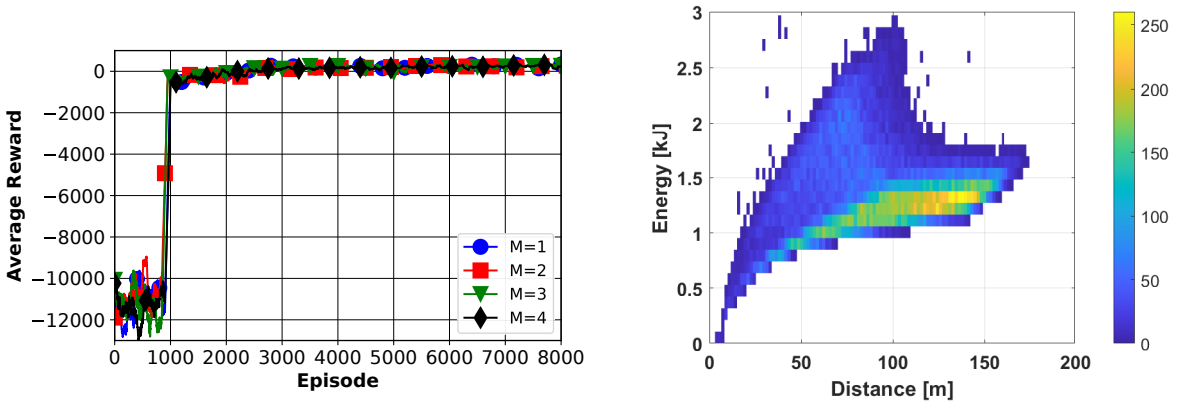
this method can be gauged from the expected coverage with M UAVs, computed as the ratio between the area covered by the UAV's FoV and the total area of the region, namely

$$M \frac{2^2 \, \mathbb{E}\{h_m^{(n)}\}^2 \tan\alpha_1 \tan\alpha_2}{S^2}. \tag{4.73}$$

For $\mathbb{E}\{h_m^{(n)}\} = (H_{\max} + H_{\min})/2$, the values obtained for $M = 3$ and $M = 4$ are, respectively, 0.199 and 0.265, consistent with the coverage curves for the U plots.

- The G benchmark performs well during the initial stages. However, as the fire grows large, say $n > 150$, both the area per pixel and coverage degrade significantly. In contrast, our method manages to provide solid coverage under both initializations. Starting with the UAVs around the ignition point, corresponding to G+RL, ensures better coverage at first, but the U+RL method rapidly relocates the UAVs to cover the fire effectively too.

Before merging the tracking and charging stages, we validate the TD3 solution applied to the charging stage via (4.41), where the same three-layer feedforward neural network is used. For training, episodes are limited to a maximum length of 200. The maximum number of



(a) (b)

Figure 4.9: For the charging problem (a) average training reward; (b) 2D histogram for the distance vs energy in a trained agent.

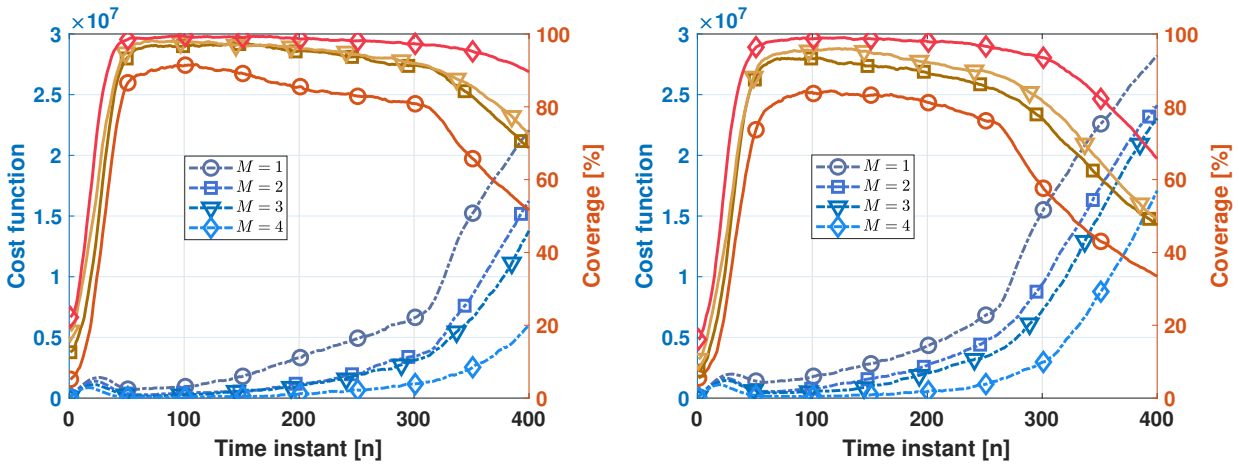
UAVs coincides with that of the tracking stage, i.e., $M = 4$. UAV locations are randomly initialized. During each episode, one UAV intends to reach a charging point, while the remaining UAVs perform random movements. Fig. 4.9a plots the average training reward versus the episode number for various values of M with $\gamma = 0.85$. After 3,000 episodes, the agent is fully trained, achieving an average reward of around 200—the value of r_{fin}—thus resulting in $\boldsymbol{q}_m^{(n)} = \boldsymbol{c}_{c_m}$. Once the agent is trained, the energy threshold at which the UAV switches from tracking to charging needs to be set. For that purpose, a 2D histogram is produced relating the distance between the UAV and the charging point when switching to the charging stage (x-axis), and the amount of energy that the UAV needs to reach the charging point (y-axis). Fig. 4.9b shows such a histogram after evaluating the trained agent over 50,000 episodes. Since the UAV should reach the charging point with very high probability, a conservative threshold should be chosen. For example, the UAV switches to charging when, at a certain distance, its level of energy falls below 1.2 times the maximum energy at that distance in Fig. 4.9b.

Finally, for the combined tracking and charging results, the UAV energies are initialized at random from $\mathcal{U}[3.5, E_{\max}]$ kJ for different $E_{\max}$. This is justified by the UAVs' previous surveillance resulting in different energy values. The minimum energy level for UAVs is set

to 3.5 kJ, ensuring that UAVs can reach a charging point in case their initial energy levels are low. We measure the swarm's coverage over 400 time steps and average the results across 1,000 different fire realizations. Fig. 4.10 plots the cost function (blue) and coverage (red) given $E_{\max} = 125$ kJ for various M and $(H_{\min}, H_{\max})$. While the initial coverage is small, UAVs rapidly adjust their locations to track the varying perimeter. Although, at each time instant, we aim at minimizing the cost function, the corresponding curves are increasing. This is due to the fire perimeter expanding with time, with a higher cost even if full coverage is achieved. Also, note that the values attained in Fig. 4.10 are similar to those in Fig. 4.7, where infinite energy is assumed, only with a 3-5% degradation.

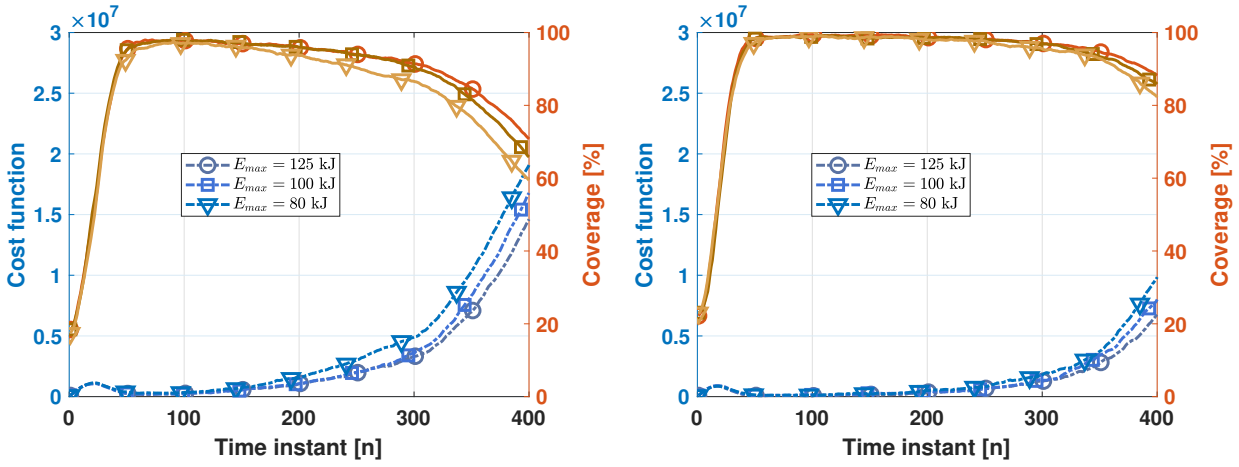
To further assess the effects of finite energy batteries, Fig. 4.11 presents results for lower values of $E_{\max}$, precisely for $E_{\max} = 125$ kJ, $E_{\max} = 100$ kJ and $E_{\max} = 80$ kJ, for a variety of M with $H_{\min} = 125$ and $H_{\max} = 150$. Interestingly, since the UAVs switch to charging more often when their energy levels are lower, the performance worsens. There is a 10% degradation in coverage for $M = 3$ while, for $M = 4$, that degradation is smaller, especially when the fire is extensive. Conversely, when a fire is in its initial stage, coverages above 90% can be achieved at all energy levels.

Fig. 4.12 illustrates the system's operation for $M = 3$ UAVs and a given wildfire realization. Precisely, Figs. 4.12a and 4.12b depict the UAV trajectories until $n = 150$ and $n = 400$, respectively, alongside the wildfire perimeter represented by means of a 2D histogram, where the value of each point represents its importance. For that same realization, the cost function and coverage are depicted in Fig. 4.12c whereas Fig. 4.12d shows the excess bit rate percentage in the cell-free connectivity constraint given by (4.36). At the beginning of the mission, UAV-1 (red) needs to recharge its battery and therefore heads to the closest charging point with coordinates $(225, 80, 10)$. In parallel, UAV-3 (blue) approaches the wildfire and, by $n = 36$, full coverage is achieved; see Fig. 4.12c. However, the energy levels of UAV-3 are low, and at $n = 74$ it switches to charging mode and heads to the charging point located at



(a) (b)

Figure 4.10: Cost and coverage for $E_{\max} = 125$ kJ and $(H_{\min}, H_{\max})$: (a) (125,150) (b) (100,125).



(a) (b)

Figure 4.11: Cost and coverage for different $E_{\max}$ (a) $M = 3$, (b) $M = 4$.

$(230, 225, 10)$, causing a reduction in coverage. At that time, UAV-1 (red) and UAV-2 (green) have enough energy and are in tracking mode, repositioning themselves closer to the wildfire. At $n = 84$, coverage begins to increase, reaching full coverage by $n = 140$. Afterwards, the three UAVs continue adjusting their locations to cover the fire perimeter, maintaining the coverage well above 80% for the remainder of the mission. Finally, Fig. 4.12d presents the excess bit rate in (4.36) as a percentage of $\frac{B}{N}$. (A value of 0 is assigned during the charging periods as UAVs do not transmit.). The combination of trajectory and power optimization yields a positive gap, indicating that the images are correctly relayed to the network.

4.7 Conclusions

This chapter has considered a cell-free UAV network whose aim is to track a wildfire while satisfying a set of mechanical, energetic, and communication constraints. Two complex nonconvex optimization problems have been formulated, for tracking and charging, and a reinforcement learning framework has been applied to tackle them. Particularly, the TD3 algorithm has been used. Extensive results have shown that a small swarm of UAVs can reliably provide coverage. Concretely, if the energy levels and the flying altitudes are moderately high, an average coverage of more than 90% can be achieved with only a few UAVs, with that coverage shrinking with the charging level and the altitude. Altogether, the tradeoff among the number of UAVs, energy, and flying altitude, has been established.

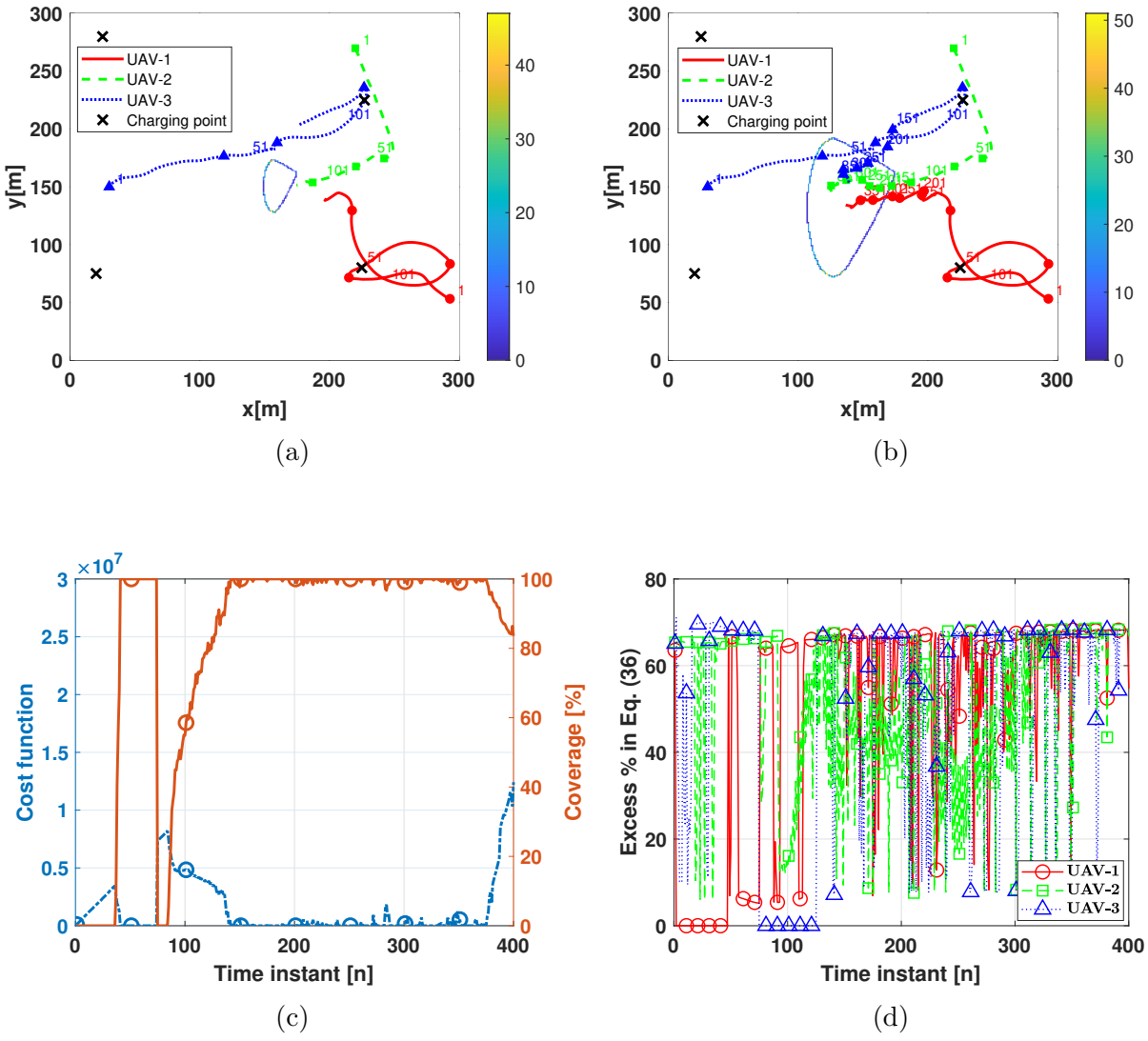


Figure 4.12: For a given wildfire realization: (a) wildfire and UAV trajectories until $n = 150$; (b) wildfire and UAV trajectories until $n = 400$; (c) cost function and coverage over the entire mission; (d) excess bit rate in (4.36).

Chapter 5

Sensing and Communication in UAV Cellular Networks: Design and Optimization

5.1 Introduction

The transition towards 6G networks has started with a potential use case of combining sensing and communications under the same umbrella [1–3]. While static sensing and/or communication networks have been greatly studied, efficient sensing and reliable communication in dynamic networks result in new challenges not perceived by the static counterparts. To circumvent these challenges, the UAV technology presents an appealing framework. Indeed, because of their advantages, e.g. low production cost or easy deployment and control, the use of UAVs in wireless communication systems has attracted significant attention during the last few years. Specifically, deploying UAVs and other nodes under certain optimality criteria is an active field of research [9, 10, 14–19, 21–25, 27, 69–71, 136–140]. For example, authors

in [10, 16, 19] regard UAVs as flying base stations with the goal of maximizing the minimum rate whereas [70, 71] exploit the relaying capabilities of the UAVs. Additionally, [138–140] present a variety of UAV-aided data-collection networks. However, jointly tackling sensing and communications problems is novel with few works dealing with it [49–52].

Compared to earlier results, this manuscript covers unsolved fundamental challenges that naturally arise in joint sensing and communication networks such as the 3D UAV trajectory optimization, interference management, and sensing scheduling. One important defining aspect of these networks is data collection versus sensing. In data collection UAV networks, sensing is done on the ground and a communication link between the ground device and the UAV is needed. On the other hand, in sensing UAV networks, UAVs sense the events cooperatively whilst guaranteeing a successful sensing probability. For example, UAVs can be equipped with a variety of cameras and sensors which, after sensing an event, generate a certain amount of data that is transmitted to the base station (BS) [53–55].

In fact, the vast majority of the literature separates sensing from communications, especially within UAV networks given the complexity of achieving optimal deployments. Therefore, this work aims at filling this gap by focusing on the UAV-energy minimization problem with respect to (*i*) 3D UAV trajectory, (*ii*) power allocation, and (*iii*) sensing in multi-cell UAV-aided networks constrained to a set of sensing, communications, and mechanical technical specifications. Concretely, each cell provides service to a set of GUs. In addition, the central cell also contains a set of events that must be sensed by the UAV, that can be regarded as an aerial user which always reports to the central BS[1]. After sensing an event, usually represented through a probabilistic sensing model [141–143], the UAV generates the sensory data which is transmitted to the central BS over a certain resource block.

Unlike the existing works in the literature, we envision a time-dependent probability of sensing, i.e., longer periods flying close to an event result in higher sensing probabilities. In

[1]Minor modifications would apply if handovers between the UAV and the BSs were considered.

addition, the sensory data transmission is mainly constrained by the information causality, UAV mobility, and interference. In fact, compared to [137, 139], and similar works, the constraints that preserve the causality of the information need to be redefined. Two major differences arise: (i) the UAV uses the same resource block at all times without any time or frequency scheduling, and (ii) there is no link between the events and UAV as previously mentioned. As a consequence, the spectral efficiency of this system is higher although a reformulation of the causality constraint is needed. Moreover, a significant issue that is often ignored when designing UAV networks, as in [21, 23, 25, 27, 70, 71, 137] among others, relates to interference. Particularly, it is assumed that the devices within each cell are assigned an orthogonal resource block; therefore, intra-cell interference is negligible, which can be easily achieved for example by orthogonal frequency division multiple access (OFDMA). However, frequency reuse among cells gives rise to inter-cell interference. Specifically, interference in aerial radio links is predominant compared to the ground counterparts given the Rician nature of air-to-ground channels [72–74]. Consequently, interference arising from the sensory data transmission from the UAV to the BS must be accounted for. Otherwise, correct decoding at certain GUs or BSs may fail. There are two main solutions to mitigate interference: (a) using coding or beamforming at the transmitter [63,65,67,144] and/or (b) perform power control at the UAV [145, 146]. In this work, to handle inter-cell interference, power control is performed at the UAV, to make it possible to carry a single antenna.

Finally, we broaden the model by considering a variable flying time. This is in contrast with most of the UAV literature, in which the flying time is fixed and predetermined. By exploiting the path discretization technique [27], we allow non-uniform time slots, which are shown to boost the UAV's energy efficiency.

The optimization problem we aim to solve is highly nonconvex; hence, it is decomposed into four subproblems and is solved utilizing the block coordinate descend (BCD) approach [147]. The first subproblem deals with the logic-based sensing variables while the second

jointly optimizes the UAV trajectory and time slot length by leveraging the well-known successive convex approximation method (SCA) [105], which is also applied to optimize the UAV altitudes. Finally, it is shown that the analytical expression for the UAV transmit power can be obtained through the Lagrangian method. Therefore, the main contributions of this chapter can be summarized as:

- A novel scenario that integrates sensing and communications is presented in UAV-aided multi-cell cellular networks based on realistic channels, sensing models, and subject to sensing, communications, and mechanical constraints.
- The UAV energy minimization problem is studied as a function of the 3D UAV path, sensing, and transmit power which results in a mixed-integer nonlinear programming problem.
- Capitalizing on four low-complexity subproblems, suboptimal solutions minimizing the on-board UAV energy are obtained. The sensing is handled through a low-complexity binary optimization algorithm while the SCA technique is used to optimize the UAV's path. Finally, the optimal UAV transmit power is analytically obtained.
- The impact and tradeoffs between a variety of parameters is well established. Our results show that the proposed scheme outperforms other benchmark methods and reduces the energy consumption between 33%-200%.

The remainder of the chapter is organized as follows. Section 5.2 presents the complete system model for the UAV-enabled sensing and communications network. The problem formulation and solution are studied in Sections 5.3 and 5.4, respectively. Numerical results are presented and discussed in Section 5.5 while concluding remarks are set forth in Section 5.6.

5.2 System Model

Consider a cellular network composed of multiple cells as shown in Fig. 5.1. Each cell contains one BS while the central cell also features one UAV and M events that need to be sensed with $\mathcal{M} = \{1, \ldots, M\}$ denoting the set of events of interest. Particularly, the mth event of interest is located inside the central cell[2] at $\boldsymbol{\ell}_m = (\boldsymbol{l}_m, 0)$ with $\boldsymbol{l}_m \in \mathbb{R}^{2\times 1}$ whose coordinates are known a priori via different techniques such as Global Positioning System (GPS)[3]. The flying/mission time is represented by T and the corresponding UAV coordinates at time t are given by $\boldsymbol{\ell}(t) = (\boldsymbol{q}(t), H(t))$ where $\boldsymbol{q}(t) \in \mathbb{R}^{2\times 1}$ represents the ground coordinates and $H(t)$ denotes the flying altitude. The BSs are located at the center of each cell. However, provided that the UAV reports to the BS in the central cell, only the location of such BS is relevant, whose coordinates are $\boldsymbol{\ell}_B = (\boldsymbol{q}_B,\, H_B)$ [4]. For the kth GU outside the central cell, its coordinates are $\boldsymbol{\ell}_k = (\boldsymbol{w}_k, 0)$. Particularly, $\{\boldsymbol{q}_B, \boldsymbol{w}_k\} \in \mathbb{R}^{2\times 1}$ and H_B is the BS altitude, common for all of them. The optimization with respect to continuous time variables would yield to an intractable problem since the number of optimization variables is infinite. Therefore, the mission time T is discretized into N non-uniform time slots, denoted by $\delta(n)$, whose lengths are included in the optimization problem. Consequently, $T = \sum_{n=1}^{N} \delta(n)$ and thus the UAV's path is discretized, where the 3D coordinates at slot n are represented by $\boldsymbol{\ell}(n) = (\boldsymbol{q}(n), H(n))$. Usually, discretizing the trajectory only specifies the locations whereas discretizing the path involves both the locations and the time dimension [27]. To maintain the sampling accuracy, an appropriate value for N must be chosen by imposing the following constraint

$$||\boldsymbol{\ell}(n+1) - \boldsymbol{\ell}(n)|| \leq d_{\max}, \tag{5.1}$$

[2]Our formulation and solutions apply to the general case in which the events are located at different cells.

[3]The impact of inaccurate locations is left as future work.

[4]Minor modifications will be needed if the central cell has to handover the UAV to another cell. If such a handover happens, the other BS location will be relevant as well.

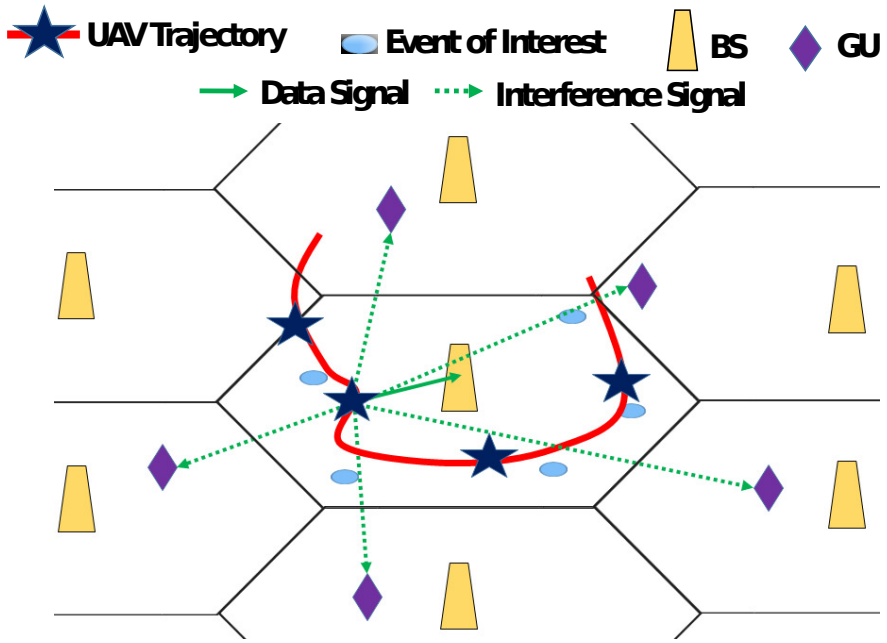


Figure 5.1: A multi-cell network with UAV-aided communications and sensing.

where $d_{\max}$ is a proper value such that the UAV is assumed to fly at a constant velocity within each segment and the parameter variation between two successive time slots is small. In fact, when $d_{\max} \to 0$, then $N \to \infty$ and a nearly continuous trajectory is obtained. To obtain a scalable N that satisfies (5.1), one can first obtain an upper bound on the flying distance, given by $D_{\max}$, and require $(N+1)d_{\max} >> D_{\max}$. Consequently, under the worst case scenario of the UAV flying $D_{\max}$, the sampling accuracy is still preserved. For ease of exposition, we define $d_B(n)$ and $d_k(n)$ as the Euclidean distance from the UAV to the central BS and User k, respectively.

5.2.1 Channel Model

The channel coefficient between the UAV and any of the network elements, denoted by $g_i(n)$ for $i \in \{B, k\}$, follows a Rician distribution, which encompasses two elements: (a) the

LoS component and (b) a Rayleigh-distributed small-scale fading component [86, Sec. 3.4.1]

$$g_i(n) = \sqrt{\frac{\beta_0}{d_i^{\kappa}(n)\,(K_i(n)+1)}}\left[\sqrt{K_i(n)}e^{j\psi_i(n)} + a_i(n)\right] \text{ for } i \in \{B, k\}, \tag{5.2}$$

where β_0 and κ are the path loss at a reference distance of 1m and the path loss exponent, respectively. The Rician factor is $K_i(n)$, which depends on the geometry between transmitter and receiver and environmental parameters [73]. Finally, $\psi_i(n) \sim \mathcal{U}[0, 2\pi]$ and $a_i(n) \sim \mathcal{N}_{\mathbb{C}}(0, 1)$ account for the phase rotation and the small-scale fading, respectively.

5.2.2 Rate Calculation

The UAV can be regarded as an aerial user within the central cell. Therefore, it is assigned an orthogonal resource block, avoiding interference to the GUs within the central cell. Upon sensing an event, the UAV generates and transmits the sensory data to the BS with an instantaneous rate of

$$R(n) = \log_2\left(1 + \frac{|g_B(n)|^2 p(n)}{\sigma^2}\right), \tag{5.3}$$

where σ^2 denotes the noise power and $p(n)$ is the UAV's transmit power, which must not exceed a certain value:

$$p(n) \leq p_{\max}, \tag{5.4}$$

where $p_{\max}$ is the maximum transmit power. A common approach to manage the randomness of the rate is to consider the ergodic capacity, i.e., $\mathbb{E}\{R(n)\}$. Furthermore, application of Jensen's inequality to $\mathbb{E}\{R(n)\}$ removes the effect of the small-scale fading given that $\mathbb{E}\{|g_B(n)|^2\} = \frac{\beta_0}{d_B^{\kappa}(n)}$. As shown in [21], to keep the contribution of the LoS and NLoS channel components, the channel coefficient can be well approximated by a logistic regression.

Therefore, as suggested in [21], the rate is approximated as

$$R(n) = \log_2 \left(1 + \left(C_1 + \frac{C_2}{1 + e^{-(B_1 + B_2 u_B(n))}}\right) \frac{\beta_0 p(n)}{d_B^{\kappa}(n)\sigma^2}\right), \tag{5.5}$$

where C_1, C_2, B_1, and B_2 are obtained from the logistic model and $u_B(n) = \frac{(H(n) - H_B)}{d_B(n)}$.

5.2.3 Interference Management

A key enabler in wireless cellular networks is interference management. While intra-cell interference can be avoided through well-known techniques such as OFDMA, neighboring cells will reuse some of the resources[5]. Consequently, given the dominance of the LoS component in aerial channels, inter-cell interference arises at both the BSs and the GUs originated by the UAV transmissions[6]. Therefore, to keep interference at a bay, a joint optimization of the UAV trajectory and power is considered, in recognition that an isolated study of one aspect may be misleading because of potential bottlenecks in the other. By using the same logistic regression approximation for the channels between the UAV and GUs (or neighboring BSs), the amount of interference that GU k receives from the UAV is:

$$I_k(n) = \left(C_1 + \frac{C_2}{1 + e^{-(B_1 + B_2 u_k(n))}}\right) \frac{\beta_0 p(n)}{d_k^{\kappa}(n)}. \tag{5.6}$$

Therefore, the following interference-related constraint must be met for each of the GUs:

$$I_k(n) \leq I_{\text{th}}, \tag{5.7}$$

[5]A reuse factor of one is assumed while the same framework and solutions apply if other sectorization and reuse factors are assumed. The sectorization or reuse factor will only affect the number of interference-related constraints not their nature.

[6]In practice, neighboring BSs can fully cancel the UAV's interference through beamforming techniques. However, if there is any remaining interference at neighboring BSs, an interference constraint, similar to those of GUs, can be added

where I_{th} is the maximum level of interference each GU can tolerate without compromising the decoding.

5.2.4 Energy Consumption Model

We consider a realistic energy consumption model for the UAV comprised of (i) flying power $p_f\big(n, \boldsymbol{\ell}(n), \delta(n)\big)$, (ii) sensing power p_s, and (iii) communication power $p(n)$ [148–150]. Following [27], for rotatory-wing UAVs, the total UAV energy, spent over the N time slots, is

$$E = \sum_{n=1}^{N} \delta(n) \Bigg(p_f\big(n, \boldsymbol{\ell}(n), \delta(n)\big) \; + \; \sum_{m=1}^{M} \alpha_m(n) p_s \; + \; p(n) \Bigg), \tag{5.8}$$

where $\alpha_m(n)$ is a logic binary variable defined as

$$\alpha_m(n) = \begin{cases} 1 & \text{if event } m \text{ is sensed at time } n \\ 0 & \text{otherwise} \end{cases}. \tag{5.9}$$

Based on [27,126], $p_f\big(n, \boldsymbol{\ell}(n), \delta(n)\big)$ is a function of the UAV locations and the time slots as follows

$$p_f\big(n, \boldsymbol{\ell}(n), \delta(n)\big) = P_0 \left(1 + \frac{3v(n)^2}{U_{tip}^2} \right) + P_i \left(\sqrt{1 + \frac{v(n)^4}{4v_0^4}} - \frac{v(n)^2}{2v_0^2} \right)^{\frac{1}{2}} + \frac{1}{2} d_0 \rho s A v(n)^3 + mgv(n) \sin \tau_c(n), \tag{5.10}$$

where $v(n)$ is the 3D velocity at time slot n defined as:

$$v(n) = \frac{||\boldsymbol{\ell}(n+1) - \boldsymbol{\ell}(n)||}{\delta(n)}, \tag{5.11}$$

$\tau_c(n)$ denotes the climb angle, which is a function of $\boldsymbol{\ell}(n)$, and the rest of parameters are defined in the aforementioned references. Finally, other constraints related to the UAV locations include a minimum and maximum flying altitude, the initial and final positions and a maximum velocity constraint:

$$H_{\min} \leq H(n) \leq H_{\max} \tag{5.12}$$

$$\boldsymbol{\ell}(1) = \boldsymbol{\ell}_i, \qquad \boldsymbol{\ell}(N) = \boldsymbol{\ell}_f, \tag{5.13}$$

$$||\boldsymbol{\ell}(n+1) - \boldsymbol{\ell}(n)|| \leq V_{\max}\delta(n). \tag{5.14}$$

5.2.5 Sensing

The goal of sensing is to collect data from different events. Indeed, optimizing the UAV's trajectory will improve the efficiency and accuracy of the sensing. Particularly, we consider M events and utilize the probabilistic sensing model [141–143] where the dependency between sensing probability and UAV trajectory is through a distance-based exponential function. In other words, the probability of sensing event m at time n is

$$P_m(n) = e^{-\mu d_m(n)}, \tag{5.15}$$

where μ determines the sensing capability of the UAV. Other relevant sensing models can be found in [136]. However, note that (5.15) avoids the dependency with respect to the

time-slot length, i.e., longer $\delta(n)$ might provide better sensing accuracy. In this work, we also study the impact of a time-dependent sensing probability. Inspired by [52], assume that the nth time slot is divided into X sub-slots of equal length t_n, i.e., $\delta(n) = Xt_n$, and that the UAV carries out X trials to sense an event. A failure in the sensing occurs when all X trials fail to correctly sense the event. Hereby, the probability of successful sensing can be obtained through its complementary:

$$P_m(n) = 1 - \left(1 - e^{-\mu d_m(n)}\right)^{\frac{\delta(n)}{t_n}}, \tag{5.16}$$

where (i) a longer $\delta(n)$ results in a higher probability and (ii) $\delta(n) \geq t_n$ to ensure at least one trial. In addition, only one event can be sensed at a time, modeled by

$$0 \leq \sum_{m=1}^{M} \alpha_m(n) \leq 1. \tag{5.17}$$

Moreover, the sensing of the M events follows a certain order, discussed in Sec. 5.4.6, determined by $\mathcal{S} = \{s_1, \ldots, s_M\}$ with $s_m \in \mathcal{M}$. Therefore, the following constraints must be met

$$\sum_{i=1}^{n} \left(\alpha_{s_m}(i) - \alpha_{s_j}(i)\right) \geq 0 \quad \text{for } m < j. \tag{5.18}$$

Furthermore, it is required to sense each event once, which can be mathematically modeled by

$$\sum_{n=1}^{N} \alpha_m(n) = 1, \tag{5.19}$$

where a certain amount of data, C_m, is assumed to be generated by the UAV if $\alpha_m(n) = 1$. Altogether, based on the logic variables $\alpha_m(n)$, the constraint that must be met in terms of

sensing probability is

$$P_{\text{s,th}} \leq \alpha_m(n) P_m(n) + A\big(1 - \alpha_m(n)\big), \tag{5.20}$$

where $P_{\text{s,th}}$ is the minimum sensing probability the UAV requires to correctly sense an event and A is chosen to be a constant larger than $P_{\text{s,th}}$. Note that a sensing variable can be active, i.e., $\alpha_m(n) = 1$, if $P_m(n) \geq P_{\text{s,th}}$. To the contrary, if $\alpha_m(n) = 0$, (5.20) is satisfied since $A \geq P_{s,th}$.

5.2.6 Causality of the information

As stated in the introduction, compared to the existing works in the literature, the constraint that relates to the causality of the information needs to be redefined given that (i) the UAV uses the same resource block at all times without any time or frequency scheduling, and (ii) there is no link between the events and UAV. In fact, upon sensing an event, the UAV generates the sensory data, e.g. a picture or a measurement among others. Let us assume the UAV senses the events in a certain order, as defined by $\mathcal{S}$ in Section 5.2.5, and the amount of data generated by the s_mth event is C_{s_m}. Successful reception of the sensory data at the BS is ensured by the following sensing constraints:

$$\begin{aligned} &\alpha_{s_M}(n) \textstyle\sum_{i=n+D}^{N} \delta(i) R(i) + B\big(1 - \alpha_{s_M}(n)\big) \geq C_{s_M} \\ &\qquad\qquad \vdots \\ &\alpha_{s_1}(n) \textstyle\sum_{i=n+D}^{N} \delta(i) R(i) + B\big(1 - \alpha_{s_1}(n)\big) \geq \sum_{m=1}^{M} C_{s_m} \end{aligned}, \tag{5.21}$$

where B is chosen to be a constant larger than $\sum_{m=1}^{M} C_{s_m}$. For example, before sensing any event or between two events, the binary variables are zero but the constraints are met due to $B \geq \sum_{m=1}^{M} C_{s_m}$. When the last event is sensed, $\alpha_{s_M}(n) = 1$ and therefore after D samples, the

amount of data that the BS receives has to be larger than C_{s_M}. Whenever the penultimate event is sensed, $\alpha_{s_{M-1}}(n) = 1$ for $n = n_{M-1}$. Hence, the BS must be able to receive the data from the last two events in the last $N - n_{M-1} - D$ samples. A similar procedure is followed to generate the causality constraints for the remaining events.

5.3 Problem Formulation

With the aim of producing energy-efficient trajectories for the sensing and communication problem, the objective function in our work is presented in Eq. (5.8). Constraints (5.4) and (5.7) determine the maximum transmit power and maximum GU interference, respectively. The set of constraints presented in (5.11)-(5.14) relate to the UAV mechanical capabilities and starting/ending point of the mission. In addition, introducing the set of logic variables $\alpha_m(n)$ allows us to combine the communication and sensing constraints under the same umbrella. More specifically, constraints (5.17)-(5.21) ensure all M events are sensed and the corresponding sensory data is successfully received at the central BS. To this end, the optimization variables are: (i) sensing variables $\alpha_m(n)$, (ii) UAV trajectory $\big(\boldsymbol{q}(n), H(n)\big)$, (iii) length of the time slots $\delta(n)$, and (iv) transmit UAV power $p(n)$. Therefore, the problem presented in (5.22) can be formulated,

$$\begin{aligned} \min_{\alpha_m(n),\boldsymbol{q}(n),H(n),\delta(n),p(n)} \quad & \sum_{n=1}^{N} \delta(n) \Bigg(p_f\big(n, \boldsymbol{\ell}(n), \delta(n)\big) \; + \; \sum_{m=1}^{M} \alpha_m(n) p_s \; + \; p(n) \Bigg) \\ \text{s.t.} \quad & \alpha_m(n) \in \{0,1\}, \\ & (5.1), (5.4), (5.7), (5.11) - (5.14), (5.17) - (5.21), \end{aligned} \tag{5.22}$$

which falls within the class of nonconvex mixed-integer non-linear programming problems whose solution requires prohibitive time and computational complexity. Accordingly, we split the problem into four subproblems: (i) optimizing sensing with path, length of the

time slots, and power fixed; (*ii*) optimizing 2D trajectory and time slot length with fixed sensing, altitude and power; (*iii*) optimizing altitude with sensing, 2D trajectory, length of the time slots, and power fixed; and (*iii*) optimizing power with sensing, path, and time slots fixed. Once the solution to each of the four subproblems is obtained, a BCD procedure [147] is followed until convergence is achieved.

5.4 Joint Optimization

In this section, we conduct a thorough analysis of the subproblems that arise from (5.22). More specifically, the optimization of the binary sensing variables is covered first. Second, an SCA-based approach for the joint trajectory and time slot optimization is presented. Finally, the optimal UAV transmit power can be derived through the Lagrangian method.

5.4.1 Sensing Optimization

For fixed $\boldsymbol{q}(n)$, $\delta(n)$, $H(n)$, and $p(n)$, we first aim at solving the sensing optimization subproblem. Provided that the only contribution of the sensing variables, $\alpha_m(n)$, in the cost function is through the term that depends on the sensing power, p_s, the first optimization problem reduces to:

$$\begin{aligned} \min_{\alpha_m(n)} \quad & \sum_{n=1}^{N}\sum_{m=1}^{M} \delta(n)\alpha_m(n)p_s \\ \text{s.t.} \quad & \alpha_m(n) \in \{0,1\}, \\ & (5.17) - (5.21). \end{aligned} \tag{5.23}$$

The binary nature of $\alpha_m(n)$ results in an NP hard problem. Relaxing the binary assumption would result in constraint violations, and therefore the problem should be kept in the binary

domain. To cope with the increased complexity, note that $\alpha_m(n) = 0$ when $P_m(n) \leq P_{\text{s,th}}$. Hence, there is no need to search over the entire solution space, but only in the points satisfying $P_m(n) \geq P_{s,th}$. Consequently, the branch and cut is an appealing technique to solve this problem [151]. The complexity of this approach is discussed in Sec. 5.4.5.

5.4.2 Trajectory and Slot Length Optimization

Given any feasible $\alpha_m(n)$, $H(n)$ and $p(n)$, optimizing the 2D trajectory and the length of the time slots reduces to the following problem:

$$\begin{aligned} \min_{\boldsymbol{q}(n),\delta(n)} \quad & \sum_{n=1}^{N} \delta(n) \Bigg(p_f\big(n, \boldsymbol{\ell}(n), \delta(n)\big) \; + \sum_{m=1}^{M} \alpha_m(n) p_s \; + \; p(n) \Bigg) \\ \text{s.t.} \quad & (5.1), (5.7), (5.11) - (5.14), (5.20), (5.21), \end{aligned} \tag{5.24}$$

which is nonconvex. Therefore, we leverage the SCA technique to create an approximated version of (5.24). To full-fill this goal, we first obtain an equivalent problem by adding different slack variables such as $\Delta(n) = ||\boldsymbol{\ell}(n+1) - \boldsymbol{\ell}(n)||$. Given that $v(n) = \frac{\Delta(n)}{\delta(n)}$, the first term in the cost function can be re-written as:

$$\begin{aligned} \sum_{n=1}^{N} \delta(n) p_f\big(n, \boldsymbol{\ell}(n), \delta(n)\big) = \sum_{n=1}^{N} P_0 \left(\delta(n) + \frac{3\Delta(n)^2}{U_{tip}^2 \delta(n)} \right) + P_i y(n) + \frac{1}{2} d_0 \rho s A \frac{\Delta(n)^3}{\delta(n)^2} + \\ mg\Delta(n) \sin \tau_c(n), \end{aligned} \tag{5.25}$$

where $y(n)$, being a second slack variable, is defined below:

$$y(n)^2 = \sqrt{\delta(n)^4 + \frac{\Delta(n)^4}{4v_0^4}} - \frac{\Delta(n)^2}{2v_0^2}. \tag{5.26}$$

$$
\begin{aligned}
\min_{\boldsymbol{q}(n),\delta(n),\Delta(n),y(n),\beta(n)} \quad & \sum_{n=1}^{N} P_0\left(\delta(n)+\frac{3\Delta(m)^2}{U_{tip}^2\delta(n)}\right)+P_i y(n)+\frac{1}{2}d_0\rho s A\frac{\Delta(n)^3}{\delta(n)^2} \\
& +\ mg\Delta(n)\sin\tau_c(n)+\sum_{n=1}^{N}\delta(n)\Big(\sum_{m=1}^{M}\alpha_m(n)p_s\ +\ p(n)\Big) \\
\text{s.t.} \quad & \frac{\beta(n)^2}{\delta(n)}\leq R(n), \\
& \frac{\delta(n)^4}{y(n)^2}\leq y(n)^2+\frac{\Delta(n)^2}{v_0^2}, \\
& ||\boldsymbol{\ell}(n+1)-\boldsymbol{\ell}(n)||\leq\Delta(n), \\
& (5.1),(5.7),(5.11),(5.13),(5.14),(5.20),(5.29).
\end{aligned} \tag{5.30}
$$

After some algebraic manipulations over the previous equation, we have

$$
\frac{\Delta(n)^4}{y(n)^2}=y(n)^2+\frac{\Delta(n)^2}{v_0^2}. \tag{5.27}
$$

Finally, a third slack variable is needed to deal with the rate-related constraints:

$$
\beta(n)^2=\delta(n)R(n). \tag{5.28}
$$

Consequently, the set of constraints in (5.21) can be expressed as a function of $\beta(n)$ in a compact manner for a given event sensing order $\mathcal{S}=\{s_1,\ldots,s_M\}$:

$$
\alpha_{s_m}(n)\sum_{i=n+D}^{N}\beta(i)^2+B\big(1-\alpha_{s_m}(n)\big)\geq\sum_{l=m}^{M}C_{s_l}. \tag{5.29}
$$

As a result, the problem in (5.24) can be re-written as presented in (5.30).

Proposition 6. *The optimization problems presented in* (5.24) *and* (5.30) *are equivalent.*

Proof. The proof can be found in App. A.9. □

Still, some of the constraints in (5.30) are nonconvex which makes SCA relevant in this

work. More precisely, SCA (a) locally convexifies the initial problem and (b) solves a convex approximated version by alternating between two steps: (i) upper (lower) bound a concave (convex) function by its first-order Taylor expansion and (ii) find the optimal solution of the approximated convex problem. In the subsequent, we derive the necessary approximations for the nonconvex constraints in (5.30). For ease of exposition, we denote variables by x and the value of the variable in the approximation point by $\overline{x}$ and first cope with the expressions that relate to $R(n)$.

Proposition 7. *$R(n)$ is jointly convex with respect to $e^{-(B_1+B_2u_B(n))}$ and $d_B^2(n)$.*

Proof. The proof can be found in App. A.10. □

Using Prop. 7, the following lower bound for $R(n)$ can be derived as a function of $\lambda(n) = B_1 + B_2u_B(n)$ and $\boldsymbol{q}(n)$.

Lemma 1. *At any local point for the UAV trajectory $\overline{\boldsymbol{q}}(n)$ and $\overline{\lambda}(n)$, $R(n)$ accepts the following lower bound:*

$$R(n) \geq R^{lb}(n) \tag{5.31}$$

$$= \overline{R}(n) - \phi(n)\big(e^{-\lambda(n)} - e^{-\overline{\lambda}(n)}\big) - \zeta(n)\big(||\boldsymbol{q}(n) - \boldsymbol{q}_B||^2 - ||\overline{\boldsymbol{q}}(n) - \boldsymbol{q}_B||^2\big). \tag{5.32}$$

Proof. The proof and the values of $\phi(n)$ and $\zeta(n)$ can be found in App. A.11. □

Hence, the constraint related to $R(n)$ in (5.30) is locally convex given $R^{lb}(n)$. However, provided that $\lambda(n)$ is nonconvex with respect to $u_B(n)$, since $u_B(n) = \frac{(H(n)-H_B)}{d_B(n)}$, a lower-bound for $u_B(n)$ is required.

Lemma 2. *At any local point for the UAV trajectory $\overline{\boldsymbol{q}}(n)$, $u_B(n)$ accepts the following lower*

bound:

$$u_B(n) \geq u_B^{lb}(n) \tag{5.33}$$

$$= \overline{u}_B(n) - \psi(n)\big(||\boldsymbol{q}(n) - \boldsymbol{q}_B||^2 - ||\overline{\boldsymbol{q}}(n) - \boldsymbol{q}_B||^2\big). \tag{5.34}$$

Proof. The proof and the value of $\psi(n)$ can be found in App. A.12. □

Therefore, the constraint that $\lambda(n)$ must satisfy is

$$\lambda(n) \leq B_1 + B_2 u_B^{lb}(n). \tag{5.35}$$

The set of constraints that takes into account the maximum interference tolerated by out-of-central cell GUs is also nonconvex with respect to $\boldsymbol{q}(n)$. However, an upper bound on the interference can be derived by considering the worst-case scenario of the UAV flying on top of User k:

$$\Big(C_1 + \frac{C_2}{1 + e^{-(B_1 + B_2 u_k(n))}}\Big)\frac{\beta_0 p(n)}{d_k^{\kappa}(n)} \leq I_k^{ub}, \tag{5.36}$$

where

$$I_k^{ub} = \left(C_1 + \frac{C_2}{1 + e^{-B_1 - B_2}}\right)\frac{\beta_0 p(n)}{d_k^{\kappa}}. \tag{5.37}$$

Following the previous equation and rearranging terms, we obtain

$$||\boldsymbol{q}(n) - \boldsymbol{w}_k||^2 \geq \left(\frac{(C_1 + \frac{C_2}{1+e^{-B_1-B_2}})\beta_0 p(n)}{I_{\text{th}}}\right)^{\frac{2}{\kappa}} - H^2, \tag{5.38}$$

$$
\begin{aligned}
\min_{\substack{\boldsymbol{q}(n),\delta(n),\Delta(n),\\ y(n),\beta(n),\tau(n)}} \quad & \sum_{n=1}^{N} P_0\left(\delta(n)+\frac{3\Delta(m)^2}{U_{tip}^2\delta(n)}\right)+P_i y(n)+\frac{1}{2}d_0\rho s A\frac{\Delta(n)^3}{\delta(n)^2}+\ mg\Delta(n)c_z \\
& +\sum_{n=1}^{N}\delta(n)\left(\sum_{m=1}^{M}\alpha_m(n)p_s\ +\ \sum_{m=1}^{M}p(n)\right) \\
\text{s.t.} \quad & \frac{\beta(n)^2}{\delta(n)}\leq R^{lb}(n), \qquad (5.41)\\
& \frac{\delta(n)^4}{y(n)^2}\leq \overline{y}(n)^2+2\overline{y}(n)\big(y(n)-\overline{y}(n)\big)+ \\
& \frac{1}{v_0^2}\left(\overline{\Delta}(n)^2+2\overline{\Delta}(n)\big(\Delta(n)-\overline{\Delta}(n)\big)\right), \\
& (5.1),(5.11),(5.13),(5.14),(5.20),(5.35),(5.39),(5.40).
\end{aligned}
$$

which is still non-convex. Further application of the SCA results in:

$$
||\overline{\boldsymbol{q}}(n)-\boldsymbol{w}_k||^2+2(\overline{\boldsymbol{q}}(n)-\boldsymbol{w}_k)^T(\boldsymbol{q}(n)-\overline{\boldsymbol{q}}(n))\geq\left(\frac{(C_1+\frac{C_2}{1+e^{-B_1-B_2}})\beta_0 p(n)}{I_{\text{th}}}\right)^{\frac{2}{\kappa}}-H^2. \qquad (5.39)
$$

A similar procedure applies to (5.29), which results in the following convex set of constraints:

$$
\alpha_{s_m}(n)\sum_{i=n+D}^{N}\left(\overline{\beta}(i)^2+2\overline{\beta}(i)\big(\beta(i)-\overline{\beta}(i)\big)\right)+B\big(1-\alpha_{s_M}(n)\big)\geq\sum_{l=m}^{M}C_{s_l}. \qquad (5.40)
$$

Armed with the respective upper or lower bounds and after applying the same technique to the expressions that involve $y(n)^2$, $\Delta(n)^2$, and, if needed (5.16), the local approximation of the problem in (5.30) is introduced in (5.41), where, compared to (5.30), we add $c_z \leq 1$ given that $\sin\tau_c(n)\leq 1$ and the sin-function is neither convex nor concave. It can be verified that both the cost function and the constraints in (5.41) are convex. Therefore, (5.41) can be solved using standard optimization solvers such as CVX [106].

$$
\begin{aligned}
\min_{\substack{H(n),\Delta(n),\\ y(n),\beta(n),\tau(n)}} \quad & \sum_{n=1}^{N} P_0\left(\delta(n)+\frac{3\Delta(m)^2}{U_{tip}^2\delta(n)}\right)+P_i y(n)+\frac{1}{2}d_0\rho s A\frac{\Delta(n)^3}{\delta(n)^2}+\ mg\Delta(n)c_z \\
\text{s.t.} \quad & \frac{\beta(n)^2}{\delta(n)} \leq R^{lb}(n), \\
& \frac{\delta(n)^4}{y(n)^2} \leq \overline{y}(n)^2+2\overline{y}(n)\big(y(n)-\overline{y}(n)\big)+ \\
& \quad \frac{1}{v_0^2}\left(\overline{\Delta}(n)^2+2\overline{\Delta}(n)\big(\Delta(n)-\overline{\Delta}(n)\big)\right), \\
& (5.1),(5.12)-(5.14),(5.20),(5.35),(5.39),(5.40).
\end{aligned}
\tag{5.42}
$$

5.4.3 Altitude Optimization

A similar methodology is followed to optimize the altitudes, $H(n)$. Given the space limitations, we skip some derivations since the process is very similar to the 2D optimization. Consequently, the approximated optimization problem for the altitude is presented in (5.42), where $R^{lb}(n)$, (5.35), and (5.39) are slightly modified to account for the gradients w.r.t. to the altitude. Finally, (5.42) is a convex optimization problem and therefore can be efficiently solved [106].

5.4.4 Power Allocation

The last subproblem aims at finding the optimal power allocation for given feasible $\alpha_m(n)$, $\boldsymbol{q}(n)$, $\delta(n)$, and $H(n)$. The terms that include $p(n)$ in the cost function and constraints result in

$$
\begin{aligned}
\min_{p(n)} \quad & \sum_{n=1}^{N}\delta(n)p(n) \\
\text{s.t.} \quad & (5.4),(5.7),(5.21),
\end{aligned}
\tag{5.43}
$$

which is convex and can be analytically solved applying the Lagrangian method.

Proposition 8. *The optimal power allocation for the UAV is given by*

$$p^*(n) = \begin{cases} 0 & n = 1, \dots, D \\ \left[a(n) - \frac{1}{K_B(n)}\right]^+ & n = D+1, \dots, N, \end{cases} \tag{5.44}$$

where $a(n)$ depends on the Lagrangian multipliers and is defined in Eq. (A.63)*, $K_B(n) = \left(C_1 + \frac{C_2}{1+e^{-(B_1+B_2 u_B(n))}}\right)\frac{\beta_0}{\sigma^2 d_B^\kappa(n)}$, and the operator $[x]^+ = \max(x, 0)$.*

Proof. The proof can be found in App. A.13. □

5.4.5 Algorithm Analysis

Based on the solutions to the previous subproblems, we propose an iterative method for the initial nonconvex problem in which we optimize four sets of variables: sensing variables, 3D UAV trajectories, time slot length, and power allocation. The convergence of the proposed BCD approach in Alg. 4 is guaranteed by the following proposition.

Proposition 9. *The sequence of objective values generated by Alg. 4 is monotonically non-increasing with a lower bound, and therefore converges.*

Proof. The proof can be found in App. A.14. □

The complexity of the previous algorithm is given by the combination of individual complexities for solving each of the subproblems. Particularly, it can be shown via induction that, given a sensing order, the maximum number of combinations for $\alpha_m(n)$ in (5.23) is upper bounded by $\sum_{i=1}^{N+M-1}(N-M+2)i$, corresponding to the case in which the M events can be sensed during the N time slots, i.e., a very conservative bound. Additionally, given that (5.41) involves logarithmic forms, the interior point method presents a complexity of $\mathcal{O}\big((7N)^{3.5}\log(1/\epsilon_1)\big)$ where ϵ_1 relates to the convergence accuracy [152]. Similarly, solving

Algorithm 4 BCD updates for the optimization variables

Require: Initial sensing, trajectory, time slots and power variables at the first iteration, $j = 0$, given by $\{\alpha_m^{(0)}(n), \boldsymbol{q}^{(0)}(n), \delta^{(0)}(n), H^{(0)}(n), p^{(0)}(n)\}$ and define by $\eta^{(0)}$ the respective cost function.

while $\frac{|\eta^{(j+1)}-\eta^{(j)}|}{\eta^{(j)}} > \epsilon$ **do**

Fix $\{\boldsymbol{q}^{(j)}(n), \delta^{(j)}(n), H^{(j)}(n), p^{(j)}(n)\}$ and solve (5.23) to obtain $\{\alpha_m^{(j+1)}(n)\}$.

Fix $\{\alpha_m^{(j+1)}(n), H^{(j)}(n), p^{(j)}(n)\}$ and solve (5.30) to obtain $\{\boldsymbol{q}^{(j+1)}(n), \delta^{(j+1)}(n)\}$.

Fix $\{\alpha_m^{(j+1)}(n), \boldsymbol{q}^{(j+1)}(n), \delta^{(j+1)}(n), p^{(j)}(n)\}$ and solve (5.42) to obtain $\{H^{(j+1)}(n)\}$.

Fix $\{\alpha_m^{(j+1)}(n), \boldsymbol{q}^{(j+1)}(n), \delta^{(j+1)}(n), H^{(j+1)}(n)\}$ and solve (5.43) to obtain $\{p^{(j+1)}(n)\}$.

Compute the cost function $\eta^{(j+1)}$.

end while

(5.42) has a complexity of $\mathcal{O}\big((5N)^{3.5}\log(1/\epsilon_2)\big)$. Finally, solving (5.43) has a complexity of $\mathcal{O}((N+N^2)\log(1/\epsilon_3)\big)$ where the term in N^2 arises after applying the ellipsoid to the dual problem [106]. Consequently, the overall complexity is dominated by the SCA-based subproblems.

5.4.6 Algorithm Initialization

Given initial and final trajectory points, the first step is to determine the sensing order $\mathcal{S}$. Different criteria may apply, e.g. based on priority, distance or random. In our case, we set the ordering based on distance. Consequently, $s_1 = \arg\min_m ||\boldsymbol{\ell}(1) - \boldsymbol{\ell}_m||$ while $s_{i+1} = \arg\min_m ||\boldsymbol{\ell}_{s_i} - \boldsymbol{\ell}_m||$ for $m \neq s_j,\ j \leq i$. Next, the value of $D_{\max}$ is obtained, serving as an upper bound on the maximum distance the UAV has to cover, i.e., $D_{\max} \geq ||\boldsymbol{\ell}(1) - \boldsymbol{\ell}_{s_1}|| + \sum_{i=1}^{M-1} ||\boldsymbol{\ell}_{s_i} - \boldsymbol{\ell}_{s_{i+1}}|| + ||\boldsymbol{\ell}_{s_M} - \boldsymbol{\ell}(N)||$. As a consequence, for a fixed $d_{\max}$, the number of slots can be set as $(N+1)d_{\max} >> D_{\max}$. Next, we turn to generating the UAV trajectory. Initially, the UAV path coordinates will lie on the lines joining the departure point, events locations, and the destination point. Therefore, we ensure the initial trajectory is capable of sensing all events. Particularly, until the first event is sensed, i.e., $P_{s_1}(n) \geq P_{\text{s,th}}$ and therefore $\alpha_{s_1}(n) = 1$, the UAV coordinates lie on the line between the departure point

and the first event as:

$$q_1(n+1) = q_1(n) + \alpha \sin\theta \cos\phi, \tag{5.45}$$

$$q_2(n+1) = q_2(n) + \alpha \sin\theta \sin\phi, \tag{5.46}$$

$$H(n+1) = H(n) + \alpha \cos\theta, \tag{5.47}$$

where θ and ϕ are the inclination and azimuth angles between $\boldsymbol{\ell}_i$ and $\boldsymbol{\ell}_{s_1}$, respectively. The trajectory between two events lies on the line connecting the location where the former is sensed and the subsequent, with the corresponding recalculation of θ and ϕ while satisfying the altitude constraints. Finally, after sensing s_M, the same formulation applies with the angles corresponding to the points associated to $\boldsymbol{\ell}_{s_M}$ and $\boldsymbol{\ell}_f$. To maintain the accuracy in the sampling while satisfying the maximum velocity constraint, we have $\alpha \leq \min\{d_{\max}, V_{\max}\delta(n)\}$. Once the trajectory and sensing variables are initialized, the powers are set to the minimum value between $p_{\max}$ and the value that satisfies the interference constraint (5.7). To verify that the causality constraints are met, a search is performed. Otherwise, either more slots can be added or the problem can be declared infeasible.

5.5 Numerical Results

For the purpose of performance evaluation, we consider a cellular network composed by one central and six neighboring cells, each following a hexagonal shape of radius $R = 200$m, although our work is independent of the cell shape. Unless otherwise specified, Table 5.1 lists the parameters used in the simulations, selected from the UAV and sensing literature [19, 49, 71]. Observe that the parameters related to Eq. (5.8) are not included since they are the same as the ones in [27]. To maintain the accuracy in the path discretization process, we consider $d_{\max} = 2$ and an upper bound on the flying distance $D_{\max} = 400$m. Therefore,

Table 5.1: Simulation Parameters

Description	Parameter	Value
Path loss at 1 m	β_0	-30 dB
Path loss exponent	κ	2
Logistic regression parameters	C_1, C_2	0, 1
Logistic regression parameters	B_1, B_2	-4.3221, 6.0750
BS altitude	H_{B}	25 m
Noise power	σ^2	-96 dBm
Minimum sensing probability	$P_{\mathrm{s,th}}$	0.9
Maximum UAV transmit power	$p_{\max}$	100 mW
Maximum GU interference	I_{th}	-73 dBm
Maximum UAV velocity	$V_{\max}$	30 m/s
Sensing capability at the UAV	μ	10^{-3}
Processing delay	D	1
Sensing power	p_s	0.1 mW

$(N+1)d_{\max} > D_{\max}$ is satisfied with $N > 300$. In this case, we use $N = 400$ where the initial length of the slots is $\delta(n) = 0.25$s with a resulting initial flying time of $T = 100$s. The typical scenario that we use in our simulations and the corresponding events and GU locations are presented in Fig. 5.2a, with $M = 8$ events and $K = 6$ GUs, i.e., one per each neighboring cell. The starting and final points are $\boldsymbol{\ell}_i = [130, -30, 50]$ and $\boldsymbol{\ell}_f = [65, 100, 50]$, respectively. The UAV trajectory for two cases of $\mu = 10^{-3}$ and $\mu = 2 \cdot 10^{-3}$ using the above typical scenario is included in Figs. 5.2a and 5.2b, with the former showing the 2D projection and the latter plotting the 3D trajectories to gain intuition on the altitude variations. Though more insight will be provided about the influence of μ, it is shown that higher values, i.e., worse sensing capability, require the UAV to fly closer to the events. In fact, from Fig. 5.2b it can be shown that the UAV tends to fly at lower altitudes for $\mu = 2 \cdot 10^{-3}$ to satisfy the sensing requirements compared to $\mu = 10^{-3}$.

We first verify Prop. 9 in Fig. 5.3 for different initializations and BCD orderings. Particularly, in O1, the optimization order in Alg. 4 is: $\{\alpha_m(n), \boldsymbol{q}(n), \delta(n), H(n), p(n)\}$ whereas in O2, the optimization order is $\{H(n), \boldsymbol{q}(n), \delta(n), p(n), \alpha_m(n)\}$. In addition, 2D initializes the altitude at a constant value $H(n) = 50$m while 3D uses the method described in Sec. 5.4.6.

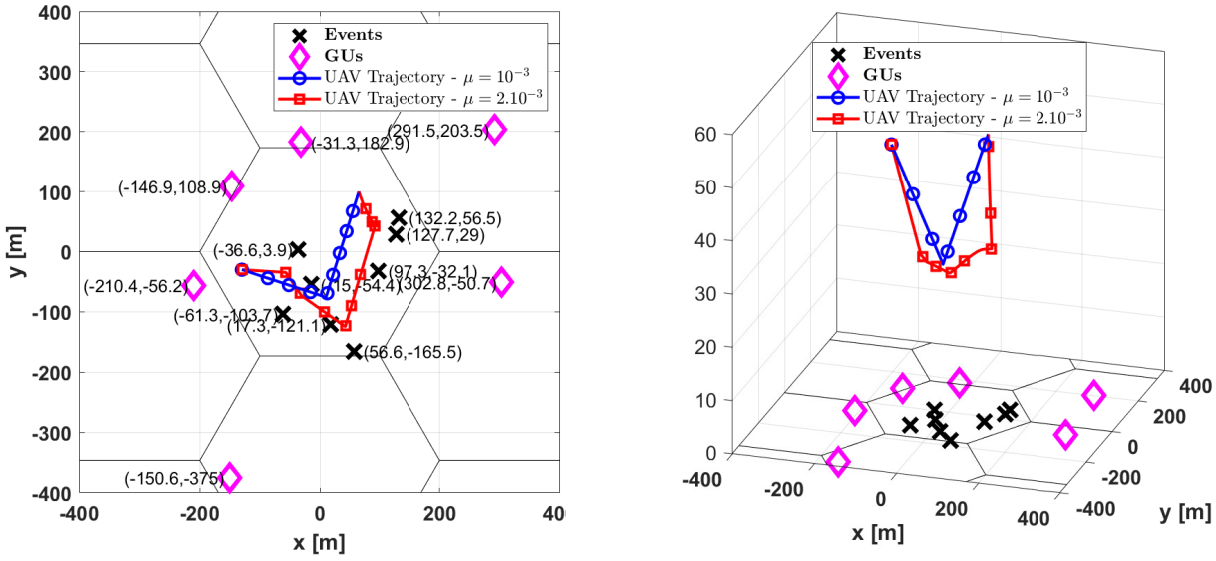


Figure 5.2: Typical scenario with $M = 8$ events, $K = 6$ GUs, and the UAV trajectory for $\mu = 10^{-3}$ and $\mu = 2 \cdot 10^{-3}$; (a) 2D, (b) 3D.

Finally, different values of c_z are tested as well to see its effect in the final solution. First, note that no matter what ordering and initialization setup we utilize, convergence in Alg. 4 is achieved only after a few iterations. Additionally, the difference in terms of cost function is minimal. However, the solutions differ. This can be seen in Fig. 5.4, where for a variety of initializations, the UAV altitudes may be different though requiring a similar energy budget. Such a tendency is mainly because all methods converge to a solution where the UAV flies at the velocity that minimizes the required energy, as described in [27, Sec. II-B]. More details on this phenomena are provided in subsequent paragraphs where the UAV velocity is also a matter of study.

Next, in Figs. 5.5a and 5.5b we compare the energy and flying time, respectively, for different benchmarks parametrized by M. Particularly, "Proposed" stands for the algorithm presented in this work while "Min-Time" solves the flying time minimization problem. Additionally, "Equal-$\delta(n)$" utilizes a similar algorithm as ours, but with fixed $\delta(n) = 0.25$s for all n. Finally, the "Max-Vel" benchmark is an heuristic algorithm based on the one described

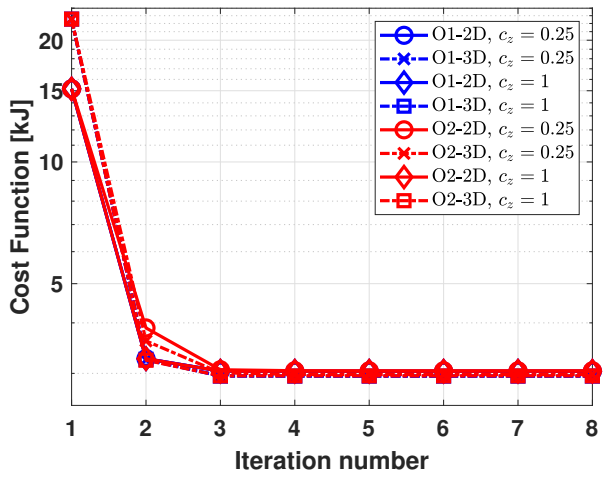


Figure 5.3: Convergence of Prop. 9 for a variety of initializations, BCD orderings and c_z.

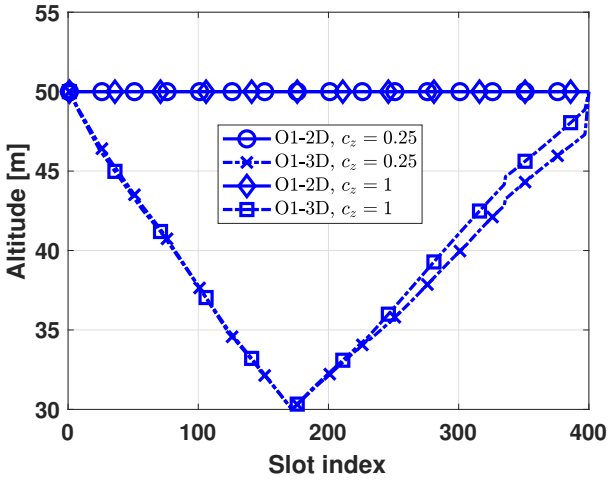
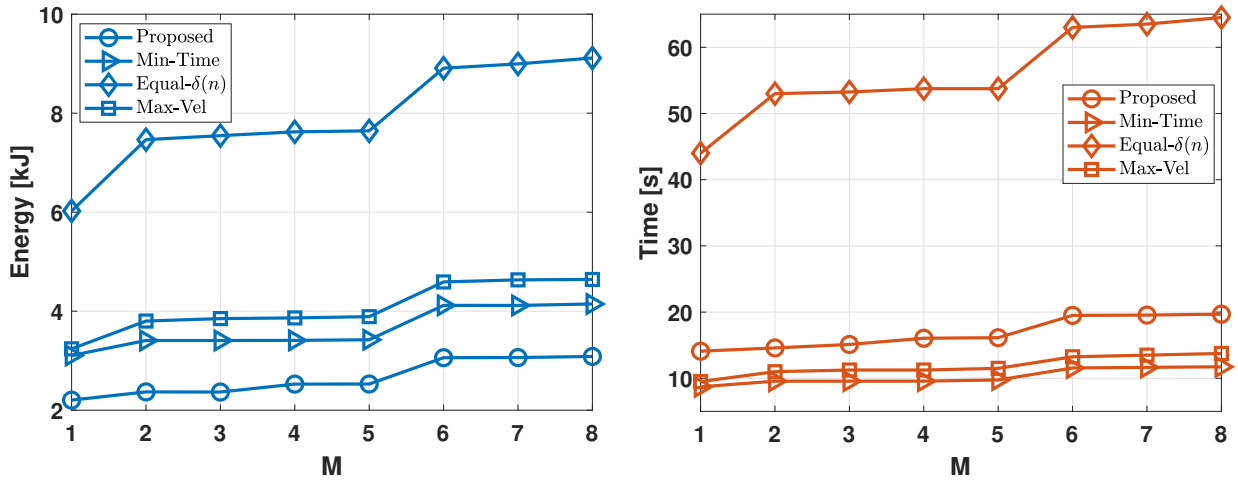


Figure 5.4: Final UAV flying altitude for a variety of initializations and c_z.

in Sec. IV-F, where the UAV flies at maximum velocity between each pair of points ensuring the rest of constraints are met. As expected, the more events, i.e., increasing M, the higher the energy and flying times are. Clearly, in terms of required on-board energy, our method outperforms the rest of benchmarks by saving at least 25% of the energy, whereas the minimum flying time is attained by the "min-Time" benchmark, with our method being close. Note that given the correlation between the "Proposed" and "min-Time" algorithms, their performance is similar, where shorter flying time results in less energy. In addition, the gap between the "Proposed" and "Equal-$\delta(n)$" curves arises by adding $\delta(n)$ into the optimization problem. Therefore, it is clear that non-uniform slots make a big difference in terms of energy-efficiency and flying time given the added degrees of freedom and the enlarged feasibility region compared to a fixed time slot.

Fig. 5.6 studies the impact of N in the simulation environment for the proposed algorithm and a variety of benchmarks. While Figs. 5.6a and 5.6b plot the energy and flying time, respectively, parametrized by N, Fig. 5.6c presents the UAV velocity for $N = 400$ under three algorithms. It can be concluded from Figs. 5.6a and 5.6b that the methods that include $\delta(n)$ in the optimization, i.e., "Proposed" and "Min-Time", converge to the same

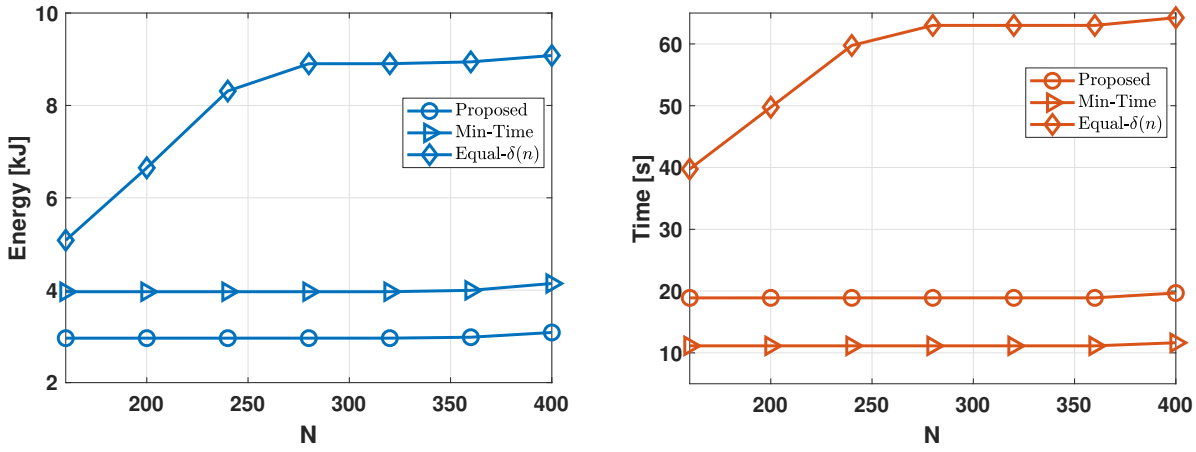


(a) (b)

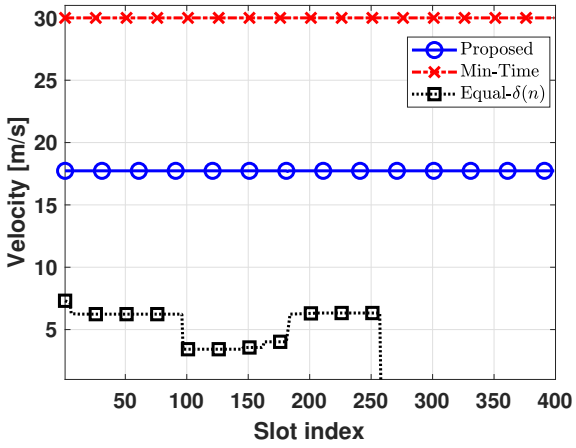
Figure 5.5: (a) Energy and (b) flying time for different M and a variety of algorithms.

cost function independent of the value of N. This is in conjunction with the flying velocities presented in Fig. 5.6c, where the "Proposed" method adjusts the UAV path to fly at the velocity that minimizes the energy, as explained in [27, Sec. II-B], achieved if $v(n) = 17.7$ m/s. In addition, to minimize the flying time, the UAV flies at its maximum velocity, in this case $v(n) = 30$ m/s. To the contrary, if $\delta(n)$ is not included in the optimization: (i) the energy and flying time tend to be higher than the other methods and (ii) after a certain value of N, both energy and time stabilize given that adding more slots is no longer helpful. In this case, it can be concluded from the three pictures that $N > 260$ does not provide any change given that the UAV can finalize its mission in less slots and therefore will hover at the destination point spending unnecessary energy.

Fig. 5.7 examines the dependency of the amount of needed energy and time to complete the sensing mission with respect to the sensing probability, $P_{\text{s,th}}$. The blue curves correspond to the required energy while the red curves relate to the flying time. Figs. 5.7a and 5.7b utilize the model in (5.15), with Fig. 5.7b considering a fixed flying altitude of $H(n) = 50$m. Finally, Fig. 5.7c considers the model in (5.16). In addition, different values for the sensing capability, μ, are also presented. Clearly, a smaller μ allows the UAV to sense the events at



(a) (b)



(c)

Figure 5.6: (a) Energy and (b) flying time for different N and a variety of algorithms. In (c), the UAV velocity for $N = 400$.

larger distances compared to a higher μ, as shown in Figs. 5.2a and 5.2b. Therefore, the use of smaller μ results in trajectories whose energy and time serve as a lower bound for higher values of μ. Additionally, increasing the value of $P_{\mathrm{s,th}}$ requires the UAV to fly closer to the events. As a consequence, both energy and flying time tend to increase with the threshold probability. By comparing Figs. 5.7a and 5.7b, it is shown that for higher values of $P_{\mathrm{s,th}}$, optimizing the altitude, as in Fig. 5.7a, reduces the required energy and time given that the distance between the UAV and the event can be smaller if the UAV has freedom to adapt its altitude. Finally, the difference between the values in Figs. 5.7a and 5.7c is minimal. Although the latter considers a sensing model that depends on the length of the time-slot, i.e., higher $\delta(n)$ results in a higher sensing probability, the energy and flying time mainly depend on the velocity. In fact, as shown in Fig. 5.7d, which plots the velocity for the models in (5.15) (solid-blue) and (5.16) (dashed-red), in both cases the UAV velocity converges to the value that minimizes the required energy, i.e., $v(n) = 17.7$ m/s. As a consequence, the energy and time obtained using (5.15) and (5.16) are similar.

Next, we study the effects of the maximum interference tolerated by the GUs, given by I_{th}. In Figs. 5.8a and 5.8b, we present the variation of energy and time parametrized by I_{th}, respectively. We present a variety of algorithms, as in previous figures. While the "Proposed" and "min-Time" refer to the same systems as in other figures, the "max-Vel" also adjusts the power as in (5.43) provided that the initialization algorithm might use more power than what is needed for the communication part. As in previous figures, the proposed method outperforms the rest in terms of required energy by at least 25% though needs more time to complete the mission. Additionally, given that the cost function is dominated by the flying energy, variations in the transmit power are not distinguishable in Fig. 5.8a. Second, in general, the amount of needed energy and time tend to increase as I_{th} decreases given that the feasible regions shrink for decreasing I_{th}. Note that for $I_{\mathrm{th}} \to 0$, the problem might be infeasible.

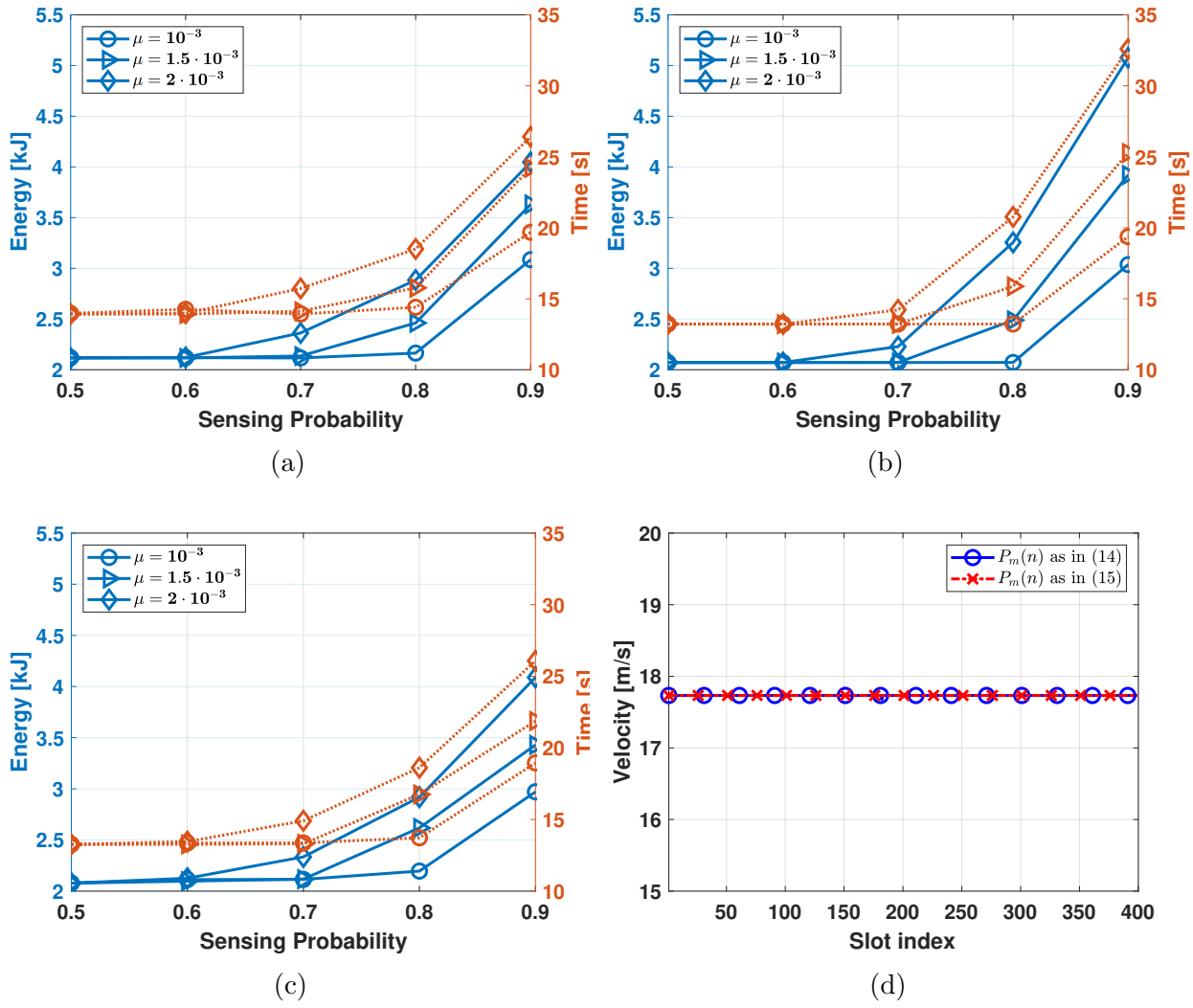
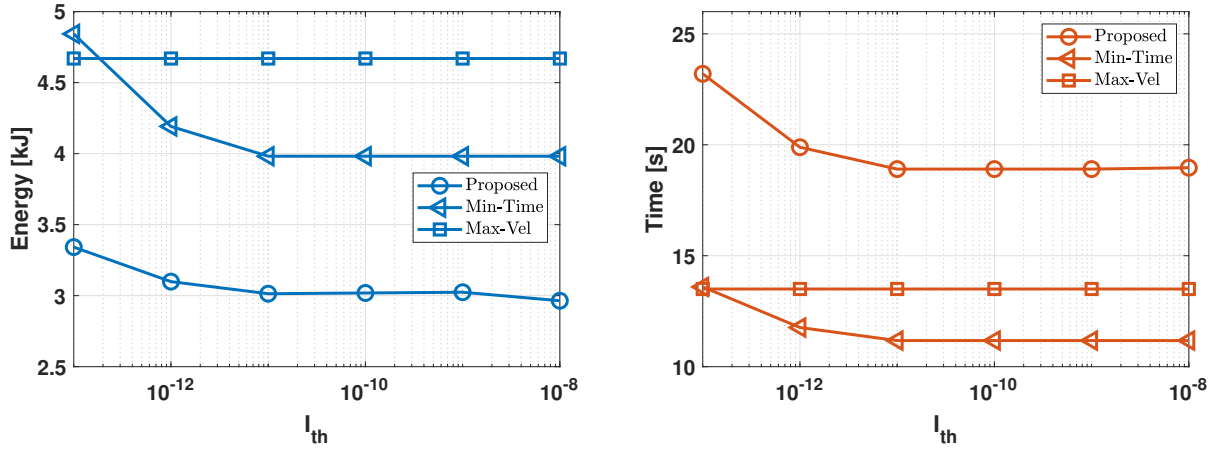


Figure 5.7: In (a)-(c), energy (blue) and flying time (red) for different μ, varying the threshold for the sensing probability for (a) sensing model in (5.15) with 3D optimization; (b) sensing model in (5.15) with 2D optimization; (c) sensing model in (5.16) with 3D optimization. In (d) velocity for $\mu = 10^{-3}$ and $P_{\text{s,th}} = 0.9$.

To gain more insight on the effects of interference, we include Fig. 5.9. Particularly, Fig. 5.9a plots the gap in the interference constraint, defined as:

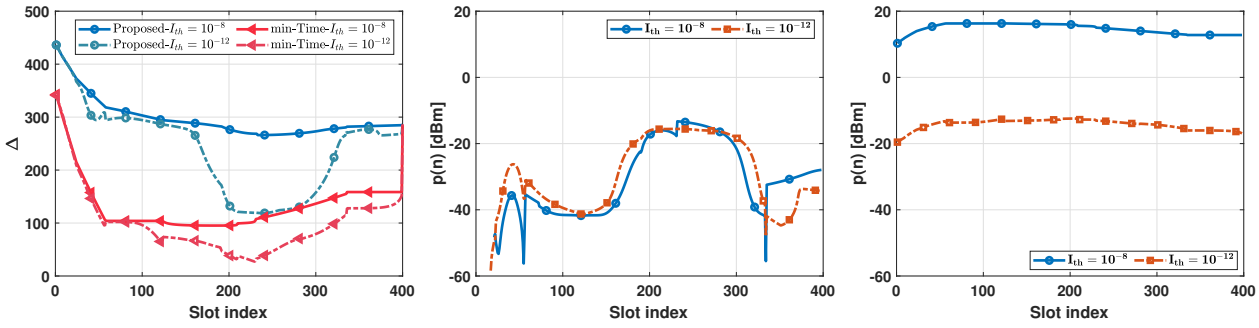
$$\Delta = \sqrt{||\boldsymbol{q}(n) - \boldsymbol{w}_k||^2 + H^2} - \sqrt{\left(\frac{(C_1 + \frac{C_2}{1+e^{-B_1-B_2}})\beta_0 p(n)}{I_{\text{th}}}\right)^{\frac{2}{\kappa}}} \tag{5.48}$$

for the GU at [302.8,−50.7]. The definition of Δ arises from calculating the difference between the square root of the left and right hand sides of the constraint presented in (5.38). Note that in fact, (5.48) provides a notion on how (5.38) is met. Higher values of Δ mean the UAV easily meets the constraint while smaller values of Δ mean the UAV finds it harder to meet the interference constraint. More particularly, by looking at Fig. 5.9a, it can be verified that smaller I_{th} yields a smaller Δ since the regions where the UAV can fly, meeting the interference constraints, become smaller. In addition, note that in fact, the "min-Time" algorithm provides smaller Δ given that higher transmit powers can still meet the constraints since its goal is to minimize the flying time, not the flying energy/power. Additionally, Figs. 5.9b and 5.9c plot the transmit power, $p(n)$, for different values of I_{th} using the "Proposed" and "min-Time" algorithms, respectively. Note that these curves compare an easy-to-meet interference constraint, i.e., $I_{\text{th}} = 10^{-8}$, versus limiting values of I_{th}. More importantly, the orders of magnitude for the transmit power are very different in Figs. 5.9b and 5.9c. While the former adjusts its power to satisfy the communication requirements with minimum power, the latter does not perform power minimization, which yields to a much higher power consumption for communication purposes. Finally, the water-filling nature of the solution obtained in (5.43) can be verified from Fig. 5.9b. Between the slots 200 and 300, the UAV flies close to the BS. Therefore, it experiences favorable channel conditions which result in the increase of the transmit power.



(a) (b)

Figure 5.8: Energy (solid/blue) and flying time (dot/red) for different I_{th} and for (a) $H = 50$m and (b) $H = 30$m.



(a) (b) (c)

Figure 5.9: For different values of I_{th} and algorithms, (a) Δ; (b) $p(n)$ for the "Proposed" algorithm; and (c) $p(n)$ for the "Min-Time" algorithm.

5.6 Conclusions

This chapter has considered energy-efficient communications and sensing where an aerial vehicle senses multiple events of interest. After generating the sensory data, the UAV ensures its reception by the BS while managing the interference effect to GUs located at neighboring cells. We have considered a generic cellular network with Rician channel models and presented mechanical-related, communication-related, and sensing-related constraints that must be satisfied to complete the mission. We presented a novel logic-based approach to formulate (a) the 3D path planning, (b) sensing, and (c) transmit power subproblems. This formulation has allowed us to use classic optimization techniques. Most remarkably, we have studied the dependency of the UAV trajectory with respect to different parameters and benchmarks, namely: (*i*) maximum velocity, (*ii*) minimum flying time, and (*iii*) fixed-slot duration. Comparative studies across various number of events have demonstrated that our proposed approach results in a reduction of energy consumption between 33%-50%. Moreover, the proposed scheme outperformed the minimum flying time and maximum velocity benchmarks by consuming 33%-41% less energy, depending on the maximum level of interference the GUs can tolerate. Finally, the number of time slots and time slot duration have a significant impact on overall performance. Our solution consumes 33% and 200% less energy compared to the minimum flying time and fixed-slot duration benchmarks, respectively.

Chapter 6

Multi-UAV Reinforcement Learning for Data Collection in Cellular MIMO Networks

6.1 Introduction

In the realm of 6G networks, data collection, sensing, and communicating stand as pivotal elements for fostering unprecedented advancements in communication technologies. The complexity and dynamic nature of 6G networks calls for a comprehensive understanding of user behaviors, network performance, and environmental factors. Through data collection, valuable insights into user preferences, network congestion patterns, and emerging usage trends can be estimated [153]. Moreover, as 6G networks aspire to integrate diverse technologies such as artificial intelligence, augmented reality, and the Internet of Things (IoT), data collection becomes instrumental in training and refining intelligent algorithms, enabling networks to adapt in real-time.

UAVs emerge as an ideal solution for data collection challenges in wireless communication networks because of their inherent versatility and mobility [7, 56–59]. UAVs offer a dynamic and adaptable platform that can be efficiently deployed to address specific data collection needs. In the context of wireless communications networks, UAVs can navigate diverse terrains and reach locations that traditional data collection methods cannot reach. In fact, the deployment of sensors and access points and the UAV deployment/trajectory optimization in wireless networks is a problem of growing interest, where different groups have studied its impact under a variety of setups [4, 9, 13–21, 23–25, 27–29, 70, 71, 154–157].

Within the UAV trajectory optimization framework, three main challenges arise: (*i*) Optimizing the UAV trajectory is challenging due to the non-convex nature of the involved functions. To address this challenge, various works adopt simplistic models, such as pure LoS channels [7, 58], single-antenna BSs [29, 70, 71], or approximating the involved functions using convex representations [23, 25, 27, 29, 70, 71, 156]. (*ii*) Once UAVs successfully collect information in a network, ensuring that subsequent transmissions of this data do not interfere with the established links connecting GUs is critical. Hence, proper interference control for the UAV transmissions is needed to avoid degrading the ongoing communications between ground-based devices. (*iii*) Delay-sensitive applications may require a maximum delay for the receipt of the collected data at the BS. Consequently, we employ the age of information (AoI) as a metric to measure the delay, initially introduced in [158]. Essentially, the larger the AoI, the less fresh the information is, whereas a smaller AoI indicates that the data is received by the BS shortly after being collected by the UAV. In fact, AoI has been used as a metric in recent UAV trajectory optimization problems, such as those discussed in [52, 159, 160].

To the best of our knowledge, this is the first work that considers a multi-UAV trajectory optimization problem that deals with the combination of the three aforementioned challenges, namely, non-convexity, interference with GUs and AoI. Particularly, effective interference

control mechanisms must be implemented. Typically, within the UAV literature, orthogonality among the UAV and/or GU resources is assumed for simplicity [7, 57, 70, 71, 156]. The use of single-antenna receivers in such an orthogonal setup makes the channel models and optimization problems very simple. However, such approach requires a large number of orthogonal resources, either in time and/or frequency. To use resources more efficiently, in this work, we allow UAVs to share resources with a set of GUs and take into account multi-antenna BSs. Particularly: (a) GUs continuously transmit to the BS, and (b) UAVs relay the collected data from a number of IoT devices to the same BS. Given that current deployments feature multi-antenna BSs, a variety of MIMO reception techniques can be applied to mitigate interference between links [82, 83]. In this work, we assume MMSE reception which is optimal from the SINR perspective [161]. This will increase the complexity of the optimization problem given the non-convexity of the SINR with respect to the UAV trajectories. Additionally, MMSE reception is combined with power control, which is known to further reduce interference [162, 163]. Consequently, the joint optimization of UAV trajectories and transmit powers serves a dual purpose: efficiently collecting data from diverse devices and concurrently preventing interference to GU-BS links.

Furthermore, this work considers imperfect CSI under Rician fading, being the appropriate channel model given the air-to-ground nature of the channels under consideration [72–74]. Other critical features, such as the UAV's energy consumption model or mechanical-related constraints are accounted for. Consequently, we formulate a multi-objective optimization problem aimed at minimizing the maximum UAV flying time while maximizing the GUs spectral efficiency, the latter contributing to reduced interference.

The emerging non-convex optimization problem poses challenges for conventional optimization methods. Therefore, we employ RL, a well-suited approach for tackling complex problems. In RL, an agent learns optimal policies through interactions with the environment, receiving feedback in the form of rewards for its actions [47]. As our problem involves con-

tinuous states and actions, DQL enhances RL by integrating deep neural networks to handle continuous state and action spaces [118–123]. Notably, various DQL algorithms have been applied to optimize UAV trajectories recently [124–128]. A specific DQL algorithm, TD3, excels in performance due to its robustness against sparse rewards and its ability to enhance stability and exploration [48]. Recent studies in the UAV literature demonstrate the advantages of using TD3 compared to its predecessors [57, 59, 129, 130], and it is therefore applied in this work. In summary, the main contributions of the chapter are:

- The analytical framework needed to model a multi-UAV data collection problem is presented. We consider the uplink of a MIMO cellular network, with aerial and terrestrial links, imperfect channel state information, and MMSE reception.
- Based on a variety of data collection, communication, energy-related, and mechanical constraints, a multi-UAV optimization problem is formulated.
- The UAV trajectory and transmit power optimization are identified using the TD3 method. The objective is to (i) minimize the maximum flying time among UAVs and (ii) keep the interference to GUs at bay.
- The tradeoff in terms of mission completion time, number of IoT devices, UAVs, cellular network size, and UAVs' energy is established.

The rest of the chapter is structured as follows: Section 6.2 introduces the system model for the multi-UAV data collection problem. In Section 6.3, the optimization problem is detailed, and Section 6.4 reveals the proposed solution using RL. Numerical results are presented in Section 6.5 and concluding remarks are discussed in Section 6.6.

6.2 System Model

Consider a cellular network featuring a base station and K GUs, whose locations are given by $\boldsymbol{w}_k = \big(\mathrm{x}_k, \mathrm{y}_k, 0\big)$. Additionally, the network includes M UAVs, whose coordinates at time n are $\boldsymbol{q}_m^{(n)} = \big(x_m^{(n)}, y_m^{(n)}, h_m^{(n)}\big)$. Finally, within the same cell, L IoT devices perform sensing tasks and are located at $\boldsymbol{i}_\ell = (\mathrm{x}_\ell, \mathrm{y}_\ell, 0)$.

Under this setup, UAVs jointly cooperate to collect and forward the sensory data from the IoT devices under a variety of energy, mechanical, and communication constraints. Consequently, UAVs can be seen as aerial users, sharing the same resources with the GUs. As previously discussed, orthogonal multiple access techniques require a large number of time and or frequency resources. Hence, spatial division multiple access (SDMA), where GUs and UAVs share the same time and frequency, is adopted in this work. Therefore, the BS is equipped with N antennas, where $N \geq K + M$ to mitigate interference. Additionally, the L IoT devices are scheduled over a different band, making N independent of L.

Finally, each of the UAVs has a certain flying time T_m, which, for uniform time slots of length δ, is given by $T_m = N_m\delta$, where N_m is the number of time slots to complete the mission of the mth UAV. In this work, we consider variable N_m while keeping δ fixed. As will be described in subsequent sections, the goal is to design the UAV trajectories and transmit powers such that the individual flying time of each UAV is minimum while collecting and forwarding the IoT data to the BS and minimizing interference to the GUs. A toy example for the networks under consideration is depicted in Fig. 6.1, with $M = 2$ UAVs, $K = 3$ GUs and $L = 3$ IoT devices.

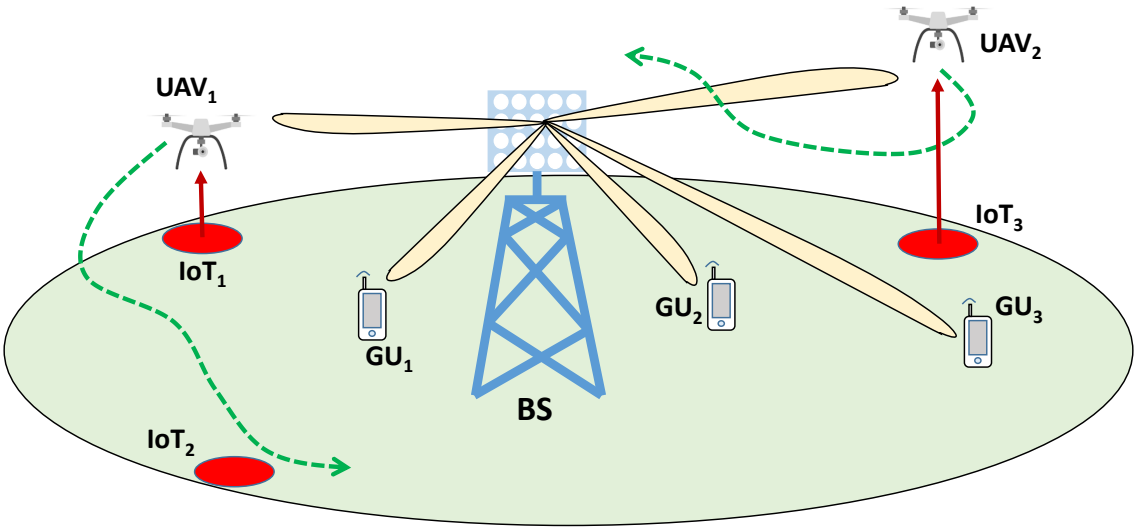


Figure 6.1: Network example with $M = 2$, $L = 3$, and $K = 3$.

6.2.1 UAV's Energy Consumption

We first consider the constraints imposed by limited-energy UAV batteries, as the model would otherwise lack a vital component, rendering it unrealistic. Hence, we first define the velocity of the mth UAV over the nth time slot as:

$$v_m^{(n)} = \frac{||\boldsymbol{q}_m^{(n+1)} - \boldsymbol{q}_m^{(n)}||}{\delta}. \tag{6.1}$$

Then, under a certain climb angle $\tau_m^{(n)}$, the 3D energy consumption model for a quad-rotor UAV is given by [126]

$$P_m^{(n)} = P_0\left(1 + \frac{3v_m^{(n)2}}{U_{\text{tip}}^2}\right) + P_i\left(\sqrt{1 + \frac{v_m^{(n)4}}{4v_0^4}} - \frac{v_m^{(n)2}}{2v_0^2}\right)^{1/2} + \frac{1}{2}d_0\iota sAv_m^{(n)3} + mgv_m^{(n)}\sin\tau_m^{(n)}, \tag{6.2}$$

where the parameters are defined in [126]. Each UAV should design its trajectory to complete the cooperative data collection mission without running out of energy. This is modeled as

$$\sum_{n=1}^{N_m} P_m^{(n)} \delta \leq E_m^{\max}, \tag{6.3}$$

where $E_m^{\max}$ is the mth UAV's energy level when it starts the mission. Finally, other constraints related to the UAV locations include the initial and final positions, limits on the flying altitude, a maximum velocity constraint, and the avoidance of collisions:

$$\boldsymbol{q}_m^{(1)} = \boldsymbol{q}_{m,\mathrm{i}}, \qquad \boldsymbol{q}_m^{(N_m)} = \boldsymbol{q}_{m,\mathrm{f}}, \tag{6.4}$$

$$H_{\min} \leq h_m^{(n)} \leq H_{\max}, \tag{6.5}$$

$$||\boldsymbol{q}_m^{(n+1)} - \boldsymbol{q}_m^{(n)}|| \leq V_{\max}\delta, \tag{6.6}$$

$$d_{m,j}^{(n)} \geq D_{\mathrm{safe}}, \tag{6.7}$$

where $\boldsymbol{q}_{m,\mathrm{i}}$ and $\boldsymbol{q}_{m,\mathrm{f}}$ represent the initial and final UAV locations, respectively. Also, the minimum and maximum allowed flying altitudes are denoted by $H_{\min}$ and $H_{\max}$, respectively. In addition, $V_{\max}$ is the maximum flying speed whereas $d_{m,j}^{(n)}$ and D_{safe} denote the distance between UAVs m and j at time n and the minimum safety distance between UAVs, respectively.

6.2.2 Data Collection

Given L IoT devices performing sensing tasks, M single-antenna UAVs jointly cooperate to collect their sensory data. Particularly, at least $C_{\ell,\min}$ bits of information must be collected from the ℓth IoT device during the mission. The capacity of the channel between the mth UAV and the ℓth IoT device at time n is denoted by $C_{m,\ell}^{(n)}$. In this work, it is assumed that the IoT devices include a single antenna and use a low transmit power, represented by p_s. Consequently, UAVs should fly near the IoT devices to collect the data which results in a LoS propagation for such links. Finally, the IoT devices operate at a different frequency band than the cellular network, avoiding interference between the two. Hence, using Shannon's capacity formula, $C_{m,\ell}^{(n)}$ is given by

$$C_{m,\ell}^{(n)} = B\delta \log_2 \left(1 + \frac{p_s \beta_0}{\left[d_{m,\ell}^{(n)}\right]^{\kappa} \sigma^2}\right), \tag{6.8}$$

where β_0 and κ are the pathloss at a reference distance of 1 m and the pathloss exponent, respectively, and $d_{m,\ell}^{(n)}$ is the distance between the mth UAV and the ℓth IoT device. In addition, σ^2 is the noise variance whereas B is the transmission bandwidth[1]. Consequently, to ensure proper data collection at the IoT devices, the following constraints must be satisfied

$$\left\{\exists\, m, n : C_{m,\ell}^{(n)} \geq C_{\ell,\min}\ ,\ \forall \ell\right\}. \tag{6.9}$$

For ease of notation, we introduce a binary variable $\alpha_{m,\ell}^{(n)}$, which takes a value of one whenever the correct collection of data from an IoT device at a certain time n occurs and zero otherwise. Minor modifications are required in (6.9) in case all UAVs are mandated to collect data from all or a subset of IoT devices.

[1]Note that while the interference between UAVs and GUs can be significant, the interference between IoT devices is neglected due to their small transmit power p_s. Fundamentally, in the presence of such an interference, the problem and proposed solution are the same with only a minor modification in Eq. (6.8)

6.2.3 Communications Process

After collecting data from an IoT device, UAVs communicate with the BS. In parallel, all K GUs actively transmit data to the BS during the mission [2]. Therefore, the communication process over the cellular network for both aerial and terrestrial users needs to be modeled.

Channel Models

Let us denote the channel from UAV m to the BS by $\boldsymbol{g}_m^{(n)}$. The N channel components are modeled by Rician distribution as

$$\boldsymbol{g}_m^{(n)} = \sqrt{\frac{\beta_0}{\left[d_m^{(n)}\right]^{\kappa}(K_m^{(n)}+1)}}\left[\sqrt{K_m^{(n)}}e^{j\psi_m}\boldsymbol{s}_m^{(n)} + \boldsymbol{a}_m^{(n)}\right], \tag{6.10}$$

where $d_m^{(n)}$ is the UAV-to-BS distance. The Rician factor is

$$K_m^{(n)} = A_1 \exp\left(A_2 \arcsin\left(\frac{h_m^{(n)}}{d_m^{(n)}}\right)\right), \tag{6.11}$$

where A_1 and A_2 are parameters that depend on the environment [73]. Moreover, ψ_m reflects the drifting, modelled as uniformly random within $[0, 2\pi]$. Also, the steering vector $\boldsymbol{s}_m^{(n)} \in \mathbb{C}^{N\times 1}$ captures the array response to the mth UAV, whose entries are

$$[\boldsymbol{s}_m^{(n)}]_n = e^{j\frac{2\pi f_c}{c}d(n-1)\sin(\theta_m^{(n)})\cos(\phi_m^{(n)})}, \tag{6.12}$$

for a uniform linear array. In (6.12), f_c represents the operating frequency, $\theta_m^{(n)}$ and $\phi_m^{(n)}$ are the azimut and elevation angles between transmitter and receiver, respectively, the antenna spacing is denoted by d, and c represents the speed of light. The small scale fading $\boldsymbol{a}_m^{(n)}$ is modeled by $\mathcal{N}_{\mathbb{C}}(\boldsymbol{0}, \boldsymbol{R_a})$ where $\boldsymbol{R_a}$ is the spatial correlation matrix at the BS. Therefore, the

[2]The solution proposed in this work is also applicable to networks where the BS schedules K different users at different time slots.

overall aerial channel covariance matrix is

$$\boldsymbol{R}_m^{(n)} = \mathbb{E}\{\boldsymbol{g}_m^{(n)} \boldsymbol{g}_m^{(n)*}\} \tag{6.13}$$

$$= \frac{\beta_0}{\left[d_m^{(n)}\right]^{\kappa}(K_m^{(n)}+1)} \left[K_m^{(n)} \boldsymbol{s}_m^{(n)} \boldsymbol{s}_m^{(n)*} + \boldsymbol{R}_{\boldsymbol{a}} \right]. \tag{6.14}$$

The channel between the kth user and the BS is Rayleigh and denoted by $\mathbf{g}_k^{(n)}$. Consequently, it follows the same expression as (6.10) by setting $K_m^{(n)} = 0$ and covariance

$$\mathbf{R}_k^{(n)} = \frac{\beta_0}{\left[\mathrm{d}_k^{(n)}\right]^{\kappa}} \mathbf{R}_{\boldsymbol{a}}, \tag{6.15}$$

where $\mathbf{R}_{\boldsymbol{a}}$ is the spatial correlation matrix at the BS and $\mathrm{d}_k^{(n)}$ denotes the distance between such BS and the kth GU.

Channel State Information

This work considers imperfect channel estimates at the BS, a novel feature within the UAV trajectory optimization literature given the added complexity. Neglecting pilot contamination, under MMSE estimation, we have $\boldsymbol{g}_m^{(n)} = \hat{\boldsymbol{g}}_m^{(n)} + \tilde{\boldsymbol{g}}_m^{(n)}$ where $\hat{\boldsymbol{g}}_m^{(n)}$ is zero-mean with covariance matrix

$$\boldsymbol{\Phi}_m^{(n)} = \mathbb{E}\left\{\hat{\boldsymbol{g}}_m^{(n)} \hat{\boldsymbol{g}}_m^{(n)*}\right\} \tag{6.16}$$

$$= \boldsymbol{R}_m^{(n)} \boldsymbol{\Psi}_m^{(n)-1} \boldsymbol{R}_m^{(n)}, \tag{6.17}$$

where $\boldsymbol{\Psi}_m^{(n)} = \boldsymbol{R}_m^{(n)} + \frac{\sigma^2}{p^t \tau} \boldsymbol{I}$, p^t is the pilot power, and τ is the pilot length. The error $\tilde{\boldsymbol{h}}_m^{(n)}$ is zero-mean with covariance $\boldsymbol{C}_m^{(n)} = \boldsymbol{R}_m^{(n)} - \boldsymbol{\Phi}_m^{(n)}$. The same derivations and results apply for the channels between GUs and BS, decomposed as $\mathbf{g}_k^{(n)} = \hat{\mathbf{h}}_k^{(n)} + \tilde{\mathbf{h}}_k^{(n)}$ with covariance matrices $\boldsymbol{\Phi}_k^{(n)}$ and $\mathbf{C}_k^{(n)}$ for $\hat{\mathbf{h}}_k^{(n)}$ and $\tilde{\mathbf{h}}_k^{(n)}$, respectively.

Data Transmission

On a given time-frequency resource, dropping the time index for simplicity, the received signal at the N-antenna BS, $\boldsymbol{y} = (y_1, \ldots, y_N)^{\mathrm{T}}$, is composed by GU and UAV data:

$$\boldsymbol{y} = \sum_{m=1}^{M} \boldsymbol{g}_m x_m + \sum_{k=1}^{K} \mathbf{g}_k \mathrm{x_k} + \boldsymbol{n} \tag{6.18}$$

$$= \underbrace{\sum_{m=1}^{M} \hat{\boldsymbol{g}}_m x_m + \sum_{k=1}^{K} \hat{\mathbf{g}}_k \mathrm{x_k}}_{\text{signals}} + \underbrace{\sum_{m=1}^{M} \tilde{\boldsymbol{g}}_m x_m + \sum_{k=1}^{K} \tilde{\mathbf{g}}_k \mathrm{x_k} + \boldsymbol{n}}_{\text{effective noise: } \boldsymbol{v}}. \tag{6.19}$$

Note that the UAVs do not need to be transmitting all the time, but only after collecting data from the IoT sensors. In (6.18), the UAV transmit data is $x_m = \sqrt{p_m} s_m$, where s_m is a complex symbol with unit power and p_m is the transmit power. Similarly, for the ground users, $\mathrm{x}_k = \sqrt{\mathrm{p}_k} \mathrm{s}_k$. In addition, $\boldsymbol{n} \sim \mathcal{N}_{\mathbb{C}}(\mathbf{0}, \sigma^2 \boldsymbol{I})$. A variety of combining techniques are available in the literature. However, MMSE reception is known to maximize the SINR, which for the kth GU attains a value of

$$\mathrm{SINR}_k = \hat{\mathbf{g}}_k^* \left(\sum_{j \neq k} \hat{\mathbf{g}}_j \hat{\mathbf{g}}_j^* \mathrm{p}_j + \sum_{m=1}^{M} \hat{\boldsymbol{g}}_m \hat{\boldsymbol{g}}_m^* p_m + \boldsymbol{\Sigma} \right)^{-1} \hat{\mathbf{g}}_k \mathrm{p}_k, \tag{6.20}$$

where $\boldsymbol{\Sigma}$ is the effective noise covariance $\boldsymbol{\Sigma} = \mathbb{E}\{\boldsymbol{v}\boldsymbol{v}^*\} = \sum_{\forall m} \boldsymbol{C}_m p_m + \sum_{\forall k} \mathbf{C}_k \mathrm{p}_k + \sigma^2 \boldsymbol{I}$. The same expression applies for the UAV's SINR after the respective modifications. Hence, the spectral efficiency is

$$\mathrm{SE}_k^{(n)} = B\delta \left(1 - \frac{\tau}{\tau_c}\right) \mathbb{E}\{\log_2(1 + \mathrm{SINR}_k^{(n)})\}, \tag{6.21}$$

where the time index has been restored, B is the transmission bandwidth and $\frac{\tau}{\tau_c}$ accounts for the pilot overhead. However, note that a closed form expression for (6.21) is not available. As a result, any optimization utilizing (6.21) will encounter stability and convergence issues. By leveraging random matrix theory, closed forms for the spectral efficiency that exclusively

depend on large-scale parameters can be obtained, providing an alternative to numerical evaluations as shown next.

Large-Dimensional Analysis

In the asymptotic regime, i.e., $N, M+K \to \infty$ with finite $\frac{N}{M+K} > 1$, the spectral efficiency value in (6.21) converges to a deterministic quantity if $\mathbf{\Phi}_m^{(n)}$ and $\mathbf{\Phi}_k^{(n)}$ have uniformly bounded spectral norms. Upon dropping the time index, the following theorem can be derived.

Theorem 6. *With $N, M+K \to \infty$ and MMSE reception, $\mathrm{SINR}_k - \overline{\mathrm{SINR}}_k \to 0$ almost surely (a.s.) where*

$$\overline{\mathrm{SINR}}_k = \mathrm{p}_k \operatorname{tr}\left[\mathbf{\Phi}_k \left(\sum_{j \neq k}^{K} \frac{\mathbf{\Phi}_j \mathrm{p}_j}{1+\mathrm{e}_j} + \sum_{m=1}^{M} \frac{\mathbf{\Phi}_m p_m}{1+e_m} + \mathbf{\Sigma} \right)^{-1}\right]. \tag{6.22}$$

Proof. The proof follows a similar procedure as the one described in Appendix A.1. □

Restoring the time index and applying the continuous mapping theorem [89], we have $\mathrm{SE}_k^{(n)} - \overline{\mathrm{SE}}_k^{(n)} \to 0$ a.s. where

$$\overline{\mathrm{SE}}_k^{(n)} = B\delta \left(1 - \frac{\tau}{\tau_\mathrm{c}}\right) \log_2\left(1 + \overline{\mathrm{SINR}}_k^{(n)}\right). \tag{6.23}$$

Note that the previous equation only depends on large-scale parameters, enhancing stability during the optimization process.

AoI of the sensory data

Once a UAV collects $C_{m,\ell}^{(n)}$ bits from an IoT device, it needs to forward them to the BS. To maintain data freshness, we utilize the concept of AoI, which essentially measures the delay between when data is collected by a UAV and when it is received at the BS. In other words, a

larger AoI implies a longer delay in the communication process. Particularly, when $\alpha_{m,\ell}^{(n)} = 1$, the mth UAV receives $C_{m,\ell}^{(n)}$ bits from the ℓth IoT device, which will be transmitted to the BS in future slots. Therefore, we can define the AoI as [158]

$$\eta_m^{(n)} = \begin{cases} \eta_m^{(n-1)} + 1 & \text{if } \Lambda_m^{(n)} = 0 \\ 1 & \text{otherwise} \end{cases}, \tag{6.24}$$

where $\Lambda_m^{(n)}$ is a binary variable defined as

$$\Lambda_m^{(n)} = \begin{cases} 1 & \text{if } \sum_{i=1}^{n} \overline{\text{SE}}_m^{(i)} \geq \sum_{i=1}^{n} \sum_{\ell=1}^{L} \alpha_{m,\ell}^{(i)} C_{m,\ell}^{(i)} \\ 0 & \text{otherwise} \end{cases}. \tag{6.25}$$

In other words, if the UAV fails to send the data after collecting it, the AoI increases until all the data gathered by the mth UAV, given by $\sum_{i=1}^{n} \sum_{\ell=1}^{L} \alpha_{m,\ell}^{(i)} C_{m,\ell}^{(i)}$, is correctly received by the BS. Hence, a constraint on the AoI can be included, enforcing freshness on the collected data:

$$\eta_m^{(n)} \leq \eta_{\max} \; \forall m, n, \tag{6.26}$$

where $\eta_{\max}$ is the maximum AoI allowed by the system.

6.3 Problem Formulation

In this section, we formulate a multi-objective optimization problem, whose goal is to minimize (i) UAVs flying time, and (ii) UAV-to-GU interference. We can combine the two

metrics to define the following cost function:

$$D\left(N_m, \boldsymbol{q}_m^{(n)}, p_m^{(n)}\right) = \max_m N_m \delta + \upsilon \sum_{n=1}^{\max_m N_m} \sum_{k=1}^{K} \lambda^{(n)} \frac{1}{\mathrm{SE}_k^{(n)}}, \tag{6.27}$$

where

$$\lambda^{(n)} = \begin{cases} 1 & \text{if } \exists\, p_m^{(n)} > 0 \\ 0 & \text{otherwise} \end{cases}, \tag{6.28}$$

and $\upsilon \in [0, 1)$ controls the trade-off between the maximum flying time and the interference. A small υ prioritizes finishing the mission as fast as possible whereas a large υ curtails the UAVs' interference to the ground users. Note that the indicator $\lambda^{(n)}$ is needed since only the time slots with an active UAV transmission are relevant in minimizing the interference. The overall minimization problem is formulated as

$$\begin{aligned} \min_{N_m, \boldsymbol{q}_m^{(n)}, p_m^{(n)}} \quad & D\left(N_m, \boldsymbol{q}_m^{(n)}, p_m^{(n)}\right) \\ \text{s.t.} \quad & p_m^{(n)} \leq p_{\max}, \\ & (6.3) - (6.9), (6.26), \end{aligned} \tag{6.29}$$

which falls in the category of non-convex mixed integer programming optimization problems. Consequently, conventional techniques such as the successive convex approximation may fail to provide close-to-optimal solutions. This is because these techniques (*i*) require the derivation of convex surrogate functions, which can be challenging to find, (*ii*) exhibit a dependency on the initialization parameters, and (*iii*) require solving the convex optimization problems for large number of variables, incurring high complexity.

6.4 DRL-Based Solution

RL offers a robust framework suitable for tackling complex problems like (6.29), thanks to its ability to adapt to non-convex and dynamic scenarios. Thus, we initially introduce the concept of Markov decision processes (MDPs), for which RL can achieve optimality. Let us denote the state and action spaces by $\mathcal{S}$ and $\mathcal{A}$, respectively. Considering $\mathcal{P}$ as the state transition probability and $R(\cdot)$ as the reward function, $(\mathcal{S}, \mathcal{A}, \mathcal{P}, R)$ defines the MDP. RL aims at finding a policy $\pi : \mathcal{S} \to \mathcal{A}$ that maximizes the expected discounted reward,

$$\pi^* = \arg\max_{\pi} \mathbb{E}\left\{ \sum_{j=0}^{\infty} \gamma^j r\big(s^{(j)}, a^{(j)}\big) \,\middle|\, \pi \right\}, \tag{6.30}$$

where $r\big(s^{(j)}, a^{(j)}\big)$ is the reward for taking action $a^{(j)}$ at state $s^{(j)}$, γ represents the discount factor, and the expectation is over $a^{(n)} \sim \pi\big(\cdot|s^{(n)}\big)$ and $s^{(n+1)} \sim P\big(s^{(n+1)}|s^{(n)}, a^{(n)}\big)$. Note that solving (6.30) is not trivial and is usually done through a surrogate function called the Q-function. The Q-function is defined as the expected cumulative reward for taking a particular action a in a given state s and subsequently adhering to policy π:

$$Q^{\pi}(s, a) = \mathbb{E}_{\pi}\left\{ \sum_{j=0}^{\infty} \gamma^j r\left(s^{(j')}, a^{(j')}\right) \,\middle|\, s^{(n)} = s, a^{(n)} = a \right\}, \tag{6.31}$$

where $j' = j + n + 1$. In fact, [47] demonstrates the feasibility of obtaining the Q-function for a small number of states and actions. However, because of the continuous domain of the involved functions, the number of states and actions in (6.29) is infinite and more innovative solutions are needed. We use DQL wherein the agent incorporates various neural networks to handle continuous states and actions. The Q-function is approximated through a neural network with parameters $\boldsymbol{\theta}$, i.e., $Q(s, a) \approx Q(s, a; \boldsymbol{\theta})$. Defining the target value

$$y(s, a) = r(s, a) + \gamma \max_{a'} Q(s', a'; \boldsymbol{\theta}'), \tag{6.32}$$

where $\boldsymbol{\theta}^{'}$ denotes the target network parameters, $\boldsymbol{\theta}$ are optimized by minimizing the loss function

$$L(\boldsymbol{\theta}) = \mathbb{E}\big\{|y(s,a) - Q(s,a;\boldsymbol{\theta})|^2\big\}\,. \tag{6.33}$$

The inclusion of $Q(s,a;\boldsymbol{\theta})$ allows handling continuous states; however, the possible actions are still discrete, restricting the UAV movements and therefore the optimality of the solution. To include continuous actions, a policy-based algorithm like TD3 can be used. Before digging into the TD3 details, we present the main ideas behind deep deterministic policy gradient (DDPG) method [48] that is a predecessor to TD3. The DDPG agent is composed of two main blocks: (i) the actor and (ii) the critic. The actor outputs the actions and aims at learning a policy, which is now modelled through a neural network with parameters $\boldsymbol{\phi}$, i.e., $\pi_{\boldsymbol{\phi}}$. The $\boldsymbol{\phi}$ updates are based on the following gradient

$$\nabla J(\boldsymbol{\phi}) = \mathbb{E}\big\{\nabla_{\boldsymbol{\phi}}\pi_{\boldsymbol{\phi}}(s)\ \nabla_a Q(s,a;\boldsymbol{\theta})|_{a=\pi_{\boldsymbol{\phi}}(s)}\big\}\,. \tag{6.34}$$

The critic is in charge of learning the Q-function via (6.33) and provides feedback to the actor regarding how beneficial an action is. To enhance the learning stability of DDPG, TD3 is leveraged in this work, as introduced in Chapter 3.

The optimization problem (6.29) is a single-agent MDP that can be decomposed into an M-agent MDP with the same goal for all agents [133,134]. Consequently, for the same UAV goals, the final models will be the same as well. Additionally, such factorization facilitates parallelization, resulting in expedited convergence and more efficient learning. Moreover, this approach scales effectively to larger environments and agent populations, maintaining its efficacy without incurring substantial increases in computational overhead. Therefore, we focus on the learning for a single UAV assuming that the remaining UAVs remain static, move randomly, or adhere to the learned policy up to that point in time.

6.4.1 TD3 Multi-UAV Data Collection

Solving (6.29) via DRL requires defining the states, actions, and rewards, as presented next.

State space

State space encompasses details concerning (i) inter-UAV distances, (ii) IoT locations, (iii) collected data awaiting transmission, (iv) destination point, and (v) distance to GUs. Particularly, the first $M-1$ state components relate to the inter-UAV distances:

$$v_m^{(n)} = \left\{ \frac{1}{C_{\mathrm{d}}} d_{m,j}^{(n)}, \text{ for } m \neq j \right\}, \tag{6.35}$$

where $d_{m,j}^{(n)}$ is the Euclidean distance between the mth and jth UAVs and $C_{\mathrm{d}} > 0$ is a normalization constant. Next, the following $3L$ states account for the IoT locations:

$$e_m^{(n)} = \left\{ i_\ell^{(n)} \left(\frac{\mathrm{x}_\ell - \mathrm{x}_{\mathrm{m}}^{(\mathrm{n})}}{C_{\mathrm{s}}}, \frac{\mathrm{y}_\ell - \mathrm{y}_{\mathrm{m}}^{(\mathrm{n})}}{C_{\mathrm{s}}}, \frac{-h_m^{(n)}}{h_{\max}} \right), \forall \ell \right\}, \tag{6.36}$$

where $C_{\mathrm{s}} > 0$ is a constant and $i_\ell^{(n)} = 1$ if data from the ℓth IoT device still needs to be collected and $i_\ell^{(n)} = 0$ otherwise. Subsequently, upon collecting data from an IoT device, the UAV forwards it to the BS. The state component that accounts for the AoI metric relates to the amount of data that still needs to be transmitted:

$$i_m^{(n)} = \max \left\{ 0, \frac{\sum_{i=1}^{n} \left(\sum_{\ell=1}^{L} \left(\alpha_{m,\ell}^{(i)} C_{m,\ell}^{(i)} \right) - \overline{\mathrm{SE}}_m^{(i)} \right)}{C_{\mathrm{r}}} \right\}. \tag{6.37}$$

Moreover, three states that include information about the UAV's destination point are added as follows:

$$f_m^{(n)} = \left\{ i^{(n)} \left(\frac{x_{m,f} - x_m^{(n)}}{C_\mathrm{s}}, \frac{y_{m,f} - y_m^{(n)}}{C_\mathrm{s}}, \frac{h_{m,f} - h_m^{(n)}}{h_\mathrm{max}} \right) \right\}, \tag{6.38}$$

where $i^{(n)} = 1$ if the data from all L events have been correctly collected and $i^{(n)} = 0$ otherwise. The following K UAV-to-GU distances, denoted by $d_{m,k}^{(n)}$, are also part of the state space to take into account the GUs' spectral efficiency optimization:

$$u_m^{(n)} = \left\{ \frac{1}{C_\mathrm{d}} d_{m,k}^{(n)},\ \forall k \right\}. \tag{6.39}$$

Finally, we also include the UAV's remaining energy as part of the state space as follows:

$$r_m^{(n)} = \frac{E_m^{\max} - \delta \sum_{i=1}^{n} P_m^{(i)}}{C_\mathrm{e}}, \tag{6.40}$$

where $C_\mathrm{e} > 0$. Altogether, the mth UAV's state is an $M + 3L + K + 4$ dimensional vector composed by

$$s_m^{(n)} = \left\{ v_m^{(n)}, e_m^{(n)}, i_m^{(n)}, f_m^{(n)}, u_m^{(n)}, r_m^{(n)} \right\}. \tag{6.41}$$

Action Space

Action space includes the set of feasible actions available to the UAVs while interacting with the environment. In this case, for a given UAV, $a^{(n)} = \left\{ a_1^{(n)}, \ldots, a_4^{(n)} \right\}$. To be precise, the first three components relate to the 3D location variations and must satisfy $-\Delta d \leq a_i^{(n)} \leq \Delta d$ for $i = 1, 2, 3$ where Δd is the maximum variation along each axis. The last component of

$$r_1(s_m^{(n)}, a^{(n)}) = \upsilon \sum_{k=1}^{K} \lambda^{(n)} \mathrm{SE}_k^{(n)} + \begin{cases} i^{(n)} K_{\mathrm{fin}} & \text{if } \boldsymbol{q}_m^{(n)} = \boldsymbol{q}_{m,f} \\ i^{(n)} K_{\mathrm{d}} \big(d_1^{(n)} - d_2^{(n)}\big) & \text{otherwise} \end{cases} \tag{6.43}$$

.

the action space relates to the UAVs' transmit power, which must satisfy $0 \leq a_4^{(n)} \leq p_{\max}$ [3].

Reward Function

Upon performing an action, the environment feeds back to the agent a scalar value that guides the agent's learning and reflects desired outcomes. Such signal is known as reward function and encodes information about the cost function and the constraints. Following the reward shaping technique [135], we use the following reward function:

$$r\big(s^{(n)}, a^{(n)}\big) = \sum_{j=1}^{6} r_j\big(s^{(n)}, a^{(n)}\big)\,, \tag{6.42}$$

where the individual rewards are defined next. First, $r_1\big(s^{(n)}, a^{(n)}\big)$ is defined in Eq. (6.43), on top of the next page, and relates to the cost function, where $d_1^{(n)} = \|\boldsymbol{q}_{m,f} - \boldsymbol{q}_m^{(n)}\|$ and $d_2^{(n)} = \|\boldsymbol{q}_{m,f} - \boldsymbol{q}_m^{(n+1)}\|$. The initial term encourages the maximization of the GU's spectral efficiency, aiming to reduce interference. The second term motivates the agent to reach the final destination once data from all IoT devices are correctly collected, i.e., if $i^{(n)} = 1$. Furthermore, K_{fin} denotes the reward for reaching the final destination, with $0 < K_{\mathrm{d}} < K_{\mathrm{fin}}$ introducing minor rewards or penalties if the UAV flies closer to or farther away from the destination point, respectively. To avoid collisions among UAVs, a negative reward of K_{coll}

[3]Note that if either all the collected data have been transmitted or the UAV has not gathered any data, the power level that minimizes interference to the GUs is $a_4^{(n)} = p_m^{(n)} = 0$.

$$r_{3,\ell}\left(s^{(n)},a^{(n)}\right)=\begin{cases} K_{\mathrm{c}} & \text{if } \alpha_{m,\ell}^{(n)}=1 \\ -K_{\mathrm{f}} & \text{if } done=True \text{ and } \sum_{m=1}^{M}\sum_{n=1}^{N_m}\alpha_{m,\ell}^{(n)}<L \\ K_{\mathrm{d}}\big(d_{m,\ell}^{(n)}-d_{m,\ell}^{(n+1)}\big) & \text{if } done=False \text{ and } \sum_{i=1}^{n}\alpha_{m,\ell}^{(i)}=0 \\ 0 & \text{otherwise} \end{cases}, \quad (6.45)$$

is added when (6.7) is violated

$$r_2\left(s^{(n)},a^{(n)}\right)=\begin{cases} -K_{\mathrm{coll}} & \text{if } \exists\ d_{m,j}^{(n)}\le D_{\mathrm{safe}} \\ 0 & \text{otherwise} \end{cases}. \quad (6.44)$$

Similarly, $r_3\left(s^{(n)},a^{(n)}\right)=\sum_{\ell} r_{3,\ell}\left(s^{(n)},a^{(n)}\right)$, where the individual rewards $r_{3,\ell}\left(s^{(n)},a^{(n)}\right)$ are presented in (6.45), on top of the next page, assigns rewards when the UAV collects data from the IoT devices. Note that a positive reward is assigned when collecting data from the ℓth device while a negative reward of $-K_{\mathrm{f}}$ is allocated when the episode ends and the UAV failed in collecting data from all IoT devices. The condition for the episode to end, i.e., $done=True$, is based on different circumstances, which can be any of the following: (i) a collision occurs, (ii) the UAV runs out of energy, or (iii) the UAV reaches the final destination. In addition, to incentivize the UAV to get closer to the ℓth IoT device that still requires its data to be collected, i.e., $\sum_{i=1}^{n}\alpha_{m,\ell}^{(i)}=0$, the third row in (6.45) is included, where $d_{m,\ell}^{(n)}=\|\boldsymbol{i}_\ell-\boldsymbol{q}_m^{(n)}\|$. To make sure the agent meets the AoI constraint presented in (6.26), $r_4\left(s^{(n)},a^{(n)}\right)$ is defined using an expression similar to (6.44) with a negative reward of $-K_{\mathrm{d}}$. Finally, $r_5\left(s^{(n)},a^{(n)}\right)$ and $r_6\left(s^{(n)},a^{(n)}\right)$ make sure the UAV stays within certain limits and does not run out of energy, respectively. The corresponding penalties for not meeting such constraints are K_{h} and K_{e}, respectively.

The same algorithm presented in Alg. 3 can be employed for training a single UAV and Fig. 6.2 shows the corresponding block diagram. To initialize such algorithm, we require the number of episodes N_{e}, representing the network realizations observed during training.

The replay buffer, initially empty, has a size of $|\mathcal{M}|$ and stores transitions of the form $\{s^{(n)}, a^{(n)}, r(s^{(n)}, a^{(n)}), s^{(n+1)}\}$. Finally, recall that an episode ends when $done = True$, which can be met by any of the following situations: (i) a collision occurs, (ii) the UAV runs out of energy, or (iii) the UAV reaches the final destination.

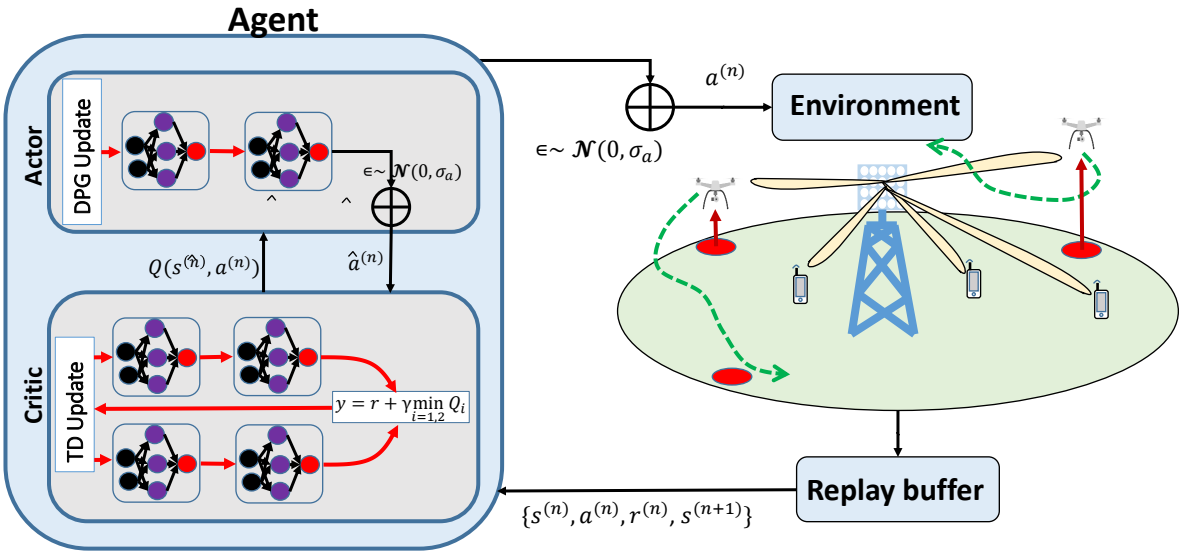


Figure 6.2: Involved blocks within the TD3 algorithm and their interactions.

6.5 Numerical Results

To evaluate the performance of the proposed approach, we consider a region of S-m^2. The BS is located at the center of the cell, whereas the GU locations, departure/destination UAV locations, and IoT coordinates are generated at random following a uniform distribution within the cell and altitude limits. Table 6.1 presents the simulation parameters, based on the UAV and MIMO literature [13, 18, 92]. Finally, the learning rate for the actor and critics on the Adam optimizer [164] is set to 10^{-4} whereas we employ a neural network with two fully connected layers of 400 and 300 neurons, respectively, for both components of the agent.

We first validate the results obtained in Theorem 1 given its inclusion in the optimization problem through (6.26). Fig. 6.3a plots the average spectral efficiency that a generic user can

Table 6.1: Simulation parameters

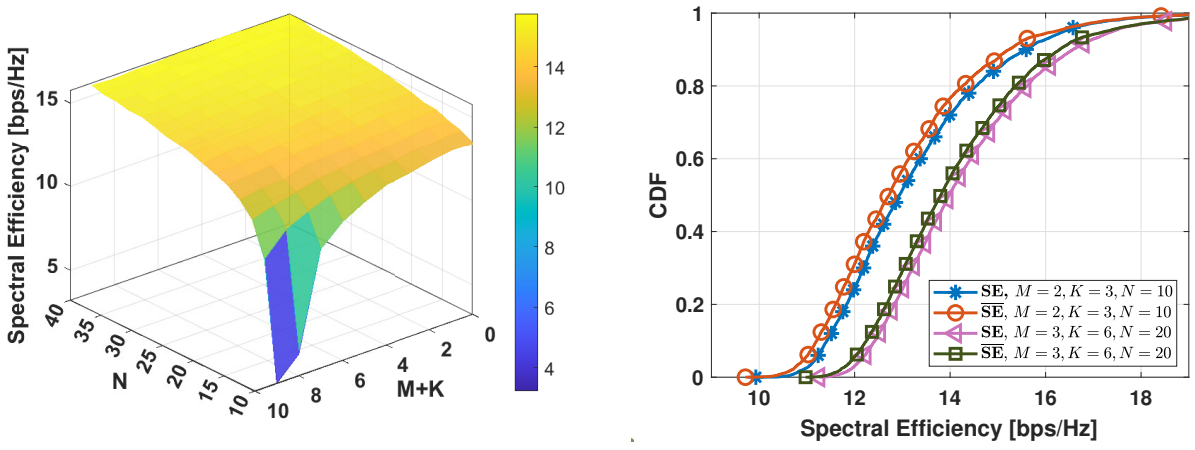
Parameter	Value	Parameter	Value
δ	0.5 s	$E_m^{\max}$	30 kJ
$H_{\min}$, $H_{\max}$	25 m, 50 m	N	20
$p_{\max}$	100 mW	K	5
Δd	8 m	D_{safe}	4 m
B	10 MHz	σ^2	-96 dBm
S	250	$\eta_{\max}$	L+1
β_0	-30 dB	A_1, A_2	0, 6.4 dB
p_s	1 mW	f_{c}	2.4 GHz
$C_{\ell,\min}$	1.5 Mbits	$\boldsymbol{R_a}$	$\boldsymbol{I}_N$
τ	200	τ_{c}	6250
p^t	100 mW	υ	0.1
c	3e8	d	0.0625 m

Table 6.2: State & reward parameters

Description	**Parameter**	**Value**
Normalization in (6.35)	C_{d}	$\sqrt{2S^2 + H_{\max}^2}$
Normalization in (6.36)	C_{s}	S
Normalization in (6.37)	C_{r}	$15LB\delta$
Normalization in (6.40)	C_{e}	30e3
Reward & constant in (6.43)	K_{fin}, K_{d}	100, 0.1, 0.1
Penalty in (6.44)	K_{coll}	10
Reward in (6.45)	K_{c}	20
Reward & penalty forward data	K_{s}, K_{d}	10, 20
Noise variance in (**??**)	σ_a	0.2
Noise variance in Fig. 6.2	$\hat{\sigma}_a$	0.2
Soft-updates memory in Alg. 3	τ_T	0.005

achieve, either GU or UAV, for $M = 3$ while varying $K = 1, \ldots, 10$ and N after normalizing by δB. As expected, increasing the ratio $N/(M + K)$ results in the increase of the spectral efficiency. Next, and maintaining the normalization by δB, Fig. 6.3b validates Theorem 1, demonstrating that even finite-dimensional systems provide tight approximations. For instance, with $N = 20$, $M = 3$, and $K = 6$.

We now discuss the training performance of the TD3 algorithm for $S = 250$-m^2 in Figs. 6.4a and 6.4b. Fig. 6.4a measures the average reward for different values of M for $K = 4$ with an averaging window of 100 samples. In addition, we include the training curves attained by

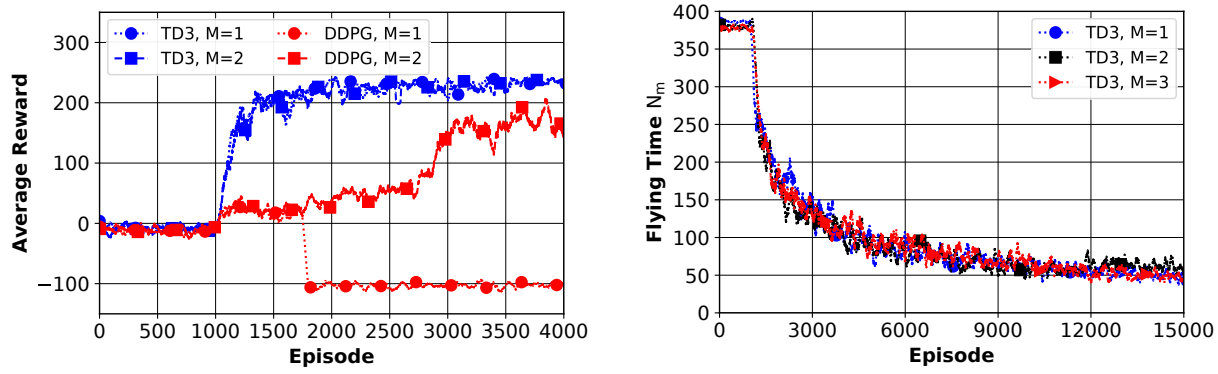


(a) (b)

Figure 6.3: (a) Spectral efficiency for different N and $M + K$; (b) validation of Theorem 1.

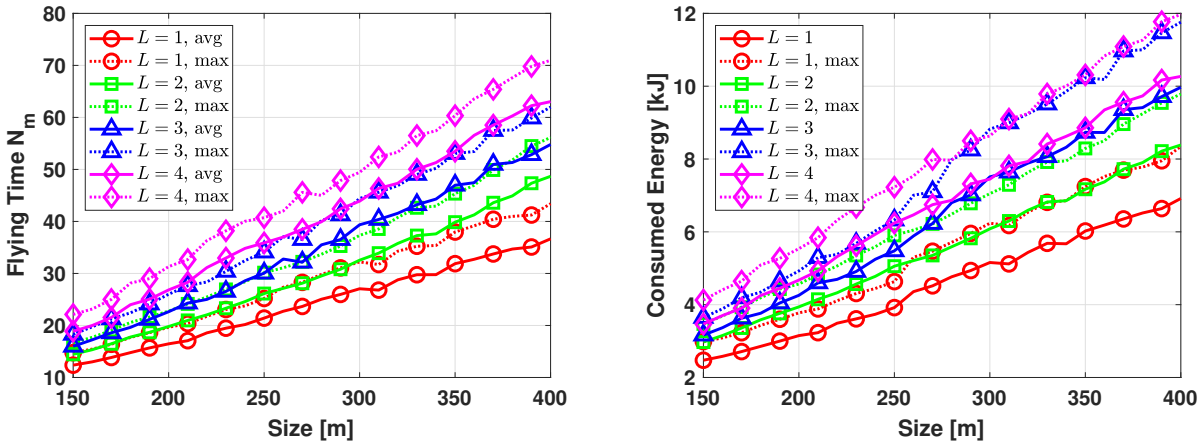
its predecessor, the DDPG algorithm, which serves as benchmark. Once the replay buffer stores enough experiences, $|\mathcal{M}| = 200,000$ in this case, the learning starts, i.e., at the $1,000$ episode mark. Clearly, the blue curves, corresponding to the TD3 method, ensure a fast convergence. To the contrary, the DDPG algorithm fails to learn when $M = 1$ whereas the learning curve for $M = 2$ is much slower compared to the TD3's curve. In addition, Fig. 6.4b presents the average flying time as a function of the number of training episodes under the TD3 approach for varying M and $L = 6$ IoT devices. In line with Fig. 6.4a, after $1,000$ episodes, the learning starts and therefore $\max_m N_m$ decreases, showing the efficacy of our proposed approach when minimizing the UAVs' flying time.

Next, we study the relationship between the flying time, the consumed energy, and the size of the region S. Specifically, for the case of $M = 2$ and varying L and S, we evaluate both the average and maximum flying times among the M UAVs, as well as the energy consumption. These metrics are illustrated in Figs. 6.5a and 6.5b, respectively, encompassing the average value from $1,000$ distinct network realizations. Unsurprisingly, the expansion of the region size correlates with an increase in both flying time and energy consumption, exhibiting a nearly linear relationship. Particularly, the ratio between flying time and network size ranges from 0.096 to 0.18 for $1 \leq L \leq 4$. A similar analysis reveals that the ratio for energy



(a) (b)

Figure 6.4: (a) Average reward during training for TD3 and DDPG with $L = 4$; (b) Average flying time during training for TD3 with $L = 6$.

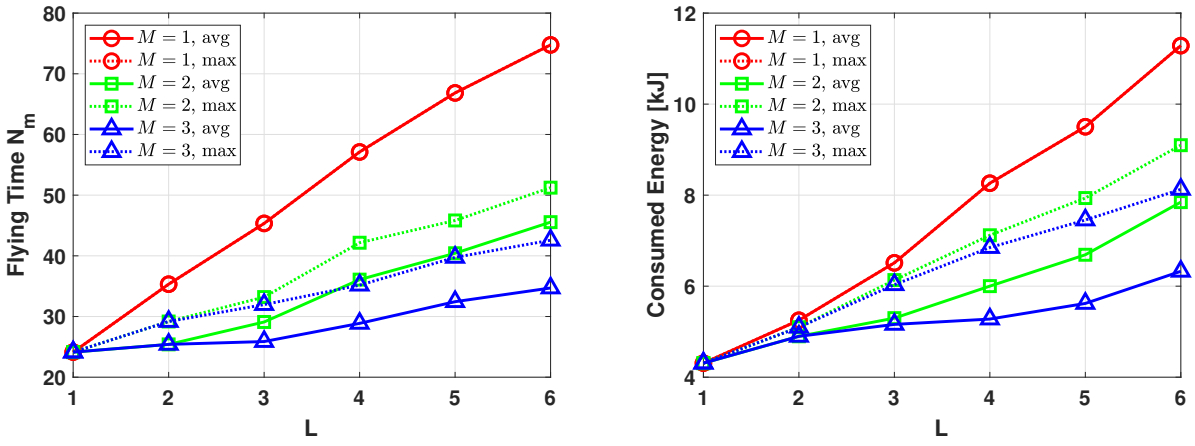


(a) (b)

Figure 6.5: For different network size S and number of IoT devices L (a) flying time; (b) consumed energy.

consumption ranges from 0.023 to 0.036. Hence, in general, for $M = 2$, $L = 4$ requires approximately twice the time and 1.5 times the energy that $L = 1$ requires.

Figs. 6.6a and 6.6b present results for the flying time and energy consumption, respectively, as L and M vary. As expected, increasing L results in an increase in both flying time and energy consumption, with different slopes observed for different values of M; specifically, lower values of M exhibit steeper slopes compared to larger values of M. For example, for $L \geq 3$, $M = 3$ needs approximately 50% of the flying time that $M = 1$ requires, whereas the

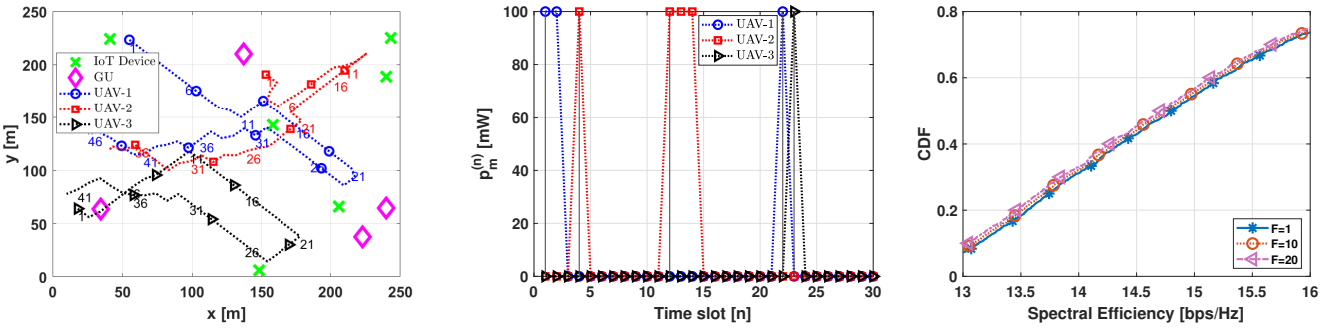


(a) (b)

Figure 6.6: For different number of IoT devices L and UAVs M (a) flying time; (b) consumed energy.

energy savings vary between 18% to 47%. In addition, while the dependency with respect to the size appears to be linear for both the flying time and energy, as seen in Fig. 6.5, when L varies for a fixed size, the flying time maintains a linear dependency, whereas the energy consumption seems to follow a different non-linear trend, with a more pronounced curve observed for smaller values of M, i.e., the lower M the higher the increase in the amount of needed energy.

Additionally, in Figs. 6.7a and 6.7b, we present a network realization along with the corresponding UAV trajectories and transmit powers, respectively. Fig. 6.7a visually demonstrates that despite training being conducted by one UAV, utilizing the same model for all M UAVs efficiently results in cooperation among them. Therefore, all UAVs can effectively collaborate and collect data from all IoT devices while minimizing their flying time, interference to GUs, avoiding collisions, and without running out of energy. In addition, Fig. 6.7b depicts the transmit power of the UAVs and when data from an IoT device has been collected (solid vertical black line). For example, UAV-1 collects data from an IoT device at $n = 1$ and given that it is far from the BS, it needs two time slots to finish the transmission, which yields to an AoI of 2. Interestingly, at $n = 11$, UAV-2 collects data from two IoT

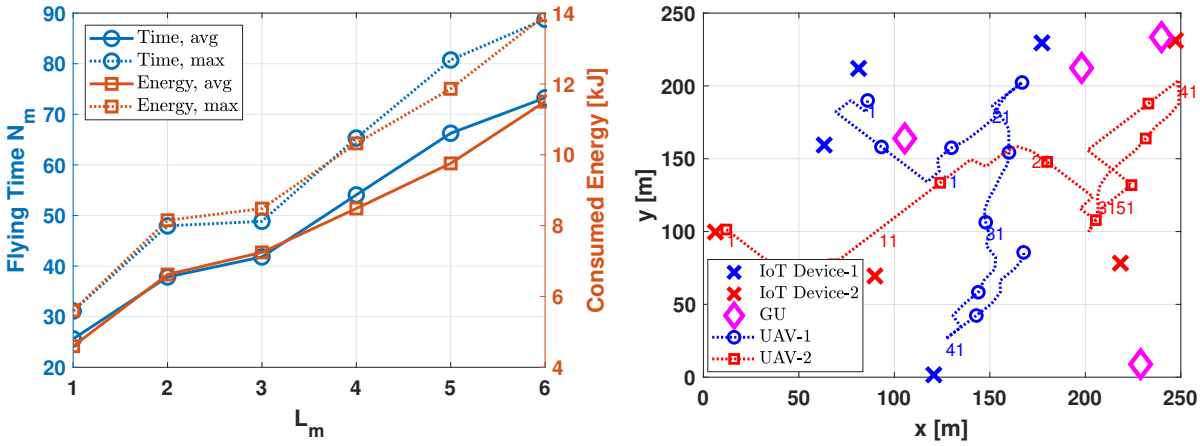


(a) (b) (c)

Figure 6.7: (a) Network realization with $M = 3$, $L = 6$, $K = 4$ and the UAV trajectories; (b) UAV transmit power for the scenario in (a); (c) GU spectral efficiency for different power allocation strategies controlled by F.

devices and needs three time slots to forward such data, making the AoI to increase up to 3. In addition, from Fig. 6.7b, it can be seen that the algorithm learns to transmit the collected data at a larger power during a few slots. This strategy is, in fact, superior to reducing the power and transmitting the collected data over more time slots when trying to avoid interference to GUs. To support this claim, we include Fig. 6.7c, which measures the spectral efficiency of GUs only, averaged over twenty time slots. The value of F determines the number of time slots a UAV is active, and to make a fair comparison, the power is also scaled by a factor of F. In other words, for large values of F, a UAV transmits data over more time slots at a lower power. It can be seen that $F = 1$ provides higher GUs spectral efficiencies, being equivalent to less interference caused by the UAVs. However, the gain is marginal compared to the values achieved by $F = 10$ or $F = 20$.

Finally, we demonstrate that the proposed approach remains valid even if each UAV collects data from a subset of IoT devices. Specifically, the learned models can be extended to accommodate this scenario by making slight modifications to Eq. (6.36) and considering only the IoT devices whose data needs to be collected by the mth UAV. Consequently, the total number of IoT devices this approach can handle becomes $L_{\text{comb}} = \sum_m L_m$, where L_m represents the number of IoT devices associated with UAV m. The primary advantage of this model is the reduction in shared information among UAVs. In other words, if UAVs



(a) (b)

Figure 6.8: (a) Flying time and consumed energy for $M = 2$ and different L; (b) network realization with $M = 2$, $L = 4$ and $K = 4$ and the UAV trajectories.

collaborate to collect data from L devices, they need to share which sensors still need to be visited. However, in a distributed approach, each UAV requires information only about the IoT devices from which it needs to collect data. To show the effectiveness of this method, we include Figs. 6.8a and 6.8b. Fig. 6.8a illustrates the flying time and consumed energy, with the respective maximum values, averaged over $1,000$ realizations for different values of L_m and M. We simplify the analysis by assuming $L_m = L_j$ for all m, j. While the trend resembles that of Fig. 6.6, the flying times in Fig. 6.8a increase due to the larger number of IoT devices per UAV. Particularly, the average flying time increases up to 65% whereas the consumed energy does so up to 51%. Nevertheless, the linear relationship between flying time and the number of IoT devices remains consistent. Additionally, Fig. 6.8b depicts a specific realization for $M = 2$ and $L_m = 4$, i.e., $L_{\text{comb}} = 8$, illustrating that each UAV only approaches IoT devices from which it must collect data. Particularly, the IoT devices associated to each UAV are depicted using the same color.

6.6 Conclusions

This chapter explores multi-UAV data collection over cellular networks. UAVs, acting as aerial users, gather data from IoT sensors and transmit them to the BS while keeping interference to GUs at bay. Key aspects such as multi-antenna BSs, imperfect CSI, MIMO reception techniques, and AoI are accounted for. Additionally, UAV-related factors including finite battery capacity, collision avoidance, and maximum speed constraints are considered. Together, these elements form a comprehensive framework for optimizing data collection, with the goal of reducing the UAVs' flying time and interference to GUs through multi-objective optimization.

Given the complexity of the problem with respect to the UAV trajectories and transmit powers, we employ a RL framework, specifically utilizing the TD3 algorithm. Extensive results are presented, elucidating the relationship between the flying time and consumed energy with respect to the network size, the number of IoT devices, and the number of UAVs. Particularly, for fixed M, increasing the network size results in a linear increase of the flying time and consumed energy with the ratio between slopes varying between $[1, 2]$ for the flying time and $[1, 1.5]$ for the consumed energy. Also, the number of UAVs plays a key role, where a reduction of up to 50% of the flying time and up to 47% of the consumed energy can be achieved. Finally, it is demonstrated that the same model applies to a distributed problem in which each UAV collects data from a set of IoT devices. However, given the non-cooperative nature of such scenario, the flying time and consumed energy can increase up to 65% and 51%, respectively, whereas the number of IoT devices that can be accounted for is larger.

Chapter 7

Conclusions

This thesis provides various analytical frameworks for cell-free and cellular UAV networks, where UAVs are utilized for communications, sensing, and/or data collection purposes. Precisely, each chapter studies either the UAV deployment or the UAV trajectory optimization combined with other resource allocation subproblems that arise in each of the scenarios under consideration.

In Chapter 2, cell-free UAV networks are introduced and investigated, where UAVs take on the role of aerial base stations. This chapter explores different channel models and reception techniques, for which deterministic equivalents for the SINR and their corresponding spectral efficiency are derived. Capitalizing on these deterministic equivalents, a deployment optimization problem is proposed for each of the architectures, where the resulting improvements in minimum SINR and in sum spectral efficiency have been thoroughly quantified. Notably, MMSE and MRC reception techniques yield significant improvements, achieving minimum SINR increases of 15-25 dB and 15-30 dB, respectively, across fully and partially centralized networks.

Chapter 3 expands upon the framework introduced in Chapter 2 by integrating wireless fron-

thaul links, transforming UAVs into flying relays. The investigation explores three multiple access schemes for the fronthaul, namely FDMA, SDMA, and FDMA-SDMA. Deterministic equivalents for the SINR with MMSE reception are provided for the three fronthaul schemes. Utilizing these equivalents, the optimization approach targets maximizing the minimum SINR with respect to the 3D UAV positions as well as user and UAV transmit powers. Results show significant improvements, ranging from 5 to 27 dB, mainly attributed to the deployment optimization.

Next, in Chapter 4, the use of UAVs for wildfire tracking is investigated. Equipped with cameras, the goal is to track a wildfire while satisfying a set of mechanical, energetic, and cell-free communication constraints. With limited battery energy, UAVs alternate between tracking and charging modes, each posing a unique nonconvex optimization problem addressed via reinforcement learning. Extensive findings indicate that a small UAV swarm can consistently ensure coverage. Specifically, under conditions of moderately high energy levels and standard flying altitudes, an average coverage exceeding 90% is achievable. However, this coverage shrinks with decreasing charging levels and flying altitudes. Altogether, the tradeoff among the number of UAVs, energy, and flying altitude, is established.

Chapter 5 explores energy-efficient communication and sensing scenarios featuring a UAV capable of sensing multiple events. After gathering sensory data, the UAV ensures its reception by the BS while managing interference to GUs in neighboring cells. This chapter considers cellular networks, along with various mechanical, communication, and sensing constraints essential for mission completion. A novel logic-based approach is introduced for formulating the 3D path planning, sensing, and transmit power subproblems, enabling the use of classic optimization techniques. Comparative studies across different benchmarks demonstrate energy consumption reductions ranging from 33% to 50% with the proposed approach depending on the maximum interference level tolerated by GUs. The number of time slots and slot duration significantly impact performance, with the proposed solution consuming 33%

and 200% less energy compared to other benchmarks.

Finally, Chapter 6 explores multi-UAV data collection in cellular networks. Serving as aerial users, UAVs collect data from ground IoT sensors and transmit it to the BS while mitigating interference to GUs under AoI, mechanical and energy constraints. Using reinforcement learning, significant reductions in flying time and energy consumption are achieved. Precisely, extensive findings reveal the correlation between flying time, energy use, network size and number of UAVs. In fact, increasing the network size results in proportional increases in the flying time and energy use, albeit with varying ratios. Furthermore, varying the number of UAVs can yield reductions of up to 50% in flying time and 47% in energy consumption. Additionally, the model extends to distributed setups, though potential increases in flying time and energy use may occur, allowing for greater IoT device coverage.

Similar methodologies and approaches could be extrapolated to diverse scenarios in both communication and sensing domains. Future work might entail exploring UAV deployment strategies or trajectory optimization for various events, extending to scenarios beyond wildfire tracking. Additionally, integrating real data into our algorithms and conducting empirical tests with physical drones in controlled environments could offer invaluable insights into their real-world efficacy and performance.

Bibliography

[1] F. Liu *et al.*, "Integrated Sensing and Communications: Toward Dual-Functional Wireless Networks for 6G and Beyond," *IEEE Journal on Sel. Areas in Commun*, vol. 40, pp. 1728–1767, Mar. 2022.

[2] T. Wild, V. Braun, and H. Viswanathan, "Joint Design of Communication and Sensing for Beyond 5G and 6G Systems," *IEEE Access*, vol. 9, pp. 30845–30857, Feb. 2021.

[3] C. De Lima *et al.*, "Convergent Communication, Sensing and Localization in 6G Systems: An Overview of Technologies, Opportunities and Challenges," *IEEE Access*, vol. 9, pp. 26902–26925, Jan. 2021.

[4] H. Jafarkhani, "Taking to the air to help on the ground: How UAVs can help fight wildfires," in *IEEE ComSoc Technology News*, Oct. 2022.

[5] M. C. P. Santos *et al.*, "A novel null-space-based UAV trajectory tracking controller with collision avoidance," *IEEE/ASME Trans. Mechatronics*, vol. 22, pp. 2543–2553, Sept. 2017.

[6] K. Meng *et al.*, "UAV trajectory and beamforming optimization for integrated periodic sensing and communication," *IEEE Wireless Commun. Lett.*, vol. 11, pp. 1211–1215, Mar. 2022.

[7] C. Zhan *et al.*, "Energy-efficient data collection in UAV enabled wireless sensor network," vol. 7, pp. 328–331, Nov. 2018.

[8] Y.-J. Chen *et al.*, "Autonomous tracking using a swarm of UAVs: A constrained multi-agent reinforcement learning approach," *IEEE Trans. Vehicular Technology*, vol. 69, pp. 13702–13717, Sept. 2020.

[9] C. Diaz-Vilor, A. Lozano, and H. Jafarkhani, "Cell-free UAV networks: Asymptotic analysis and deployment optimization," *IEEE Trans. Wireless Commun.*, pp. 1–1, Oct. 2022.

[10] C. Diaz-Vilor, A. Lozano, and H. Jafarkhani, "On the deployment problem in Cell-Free UAV networks," in *2021 IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2021.

[11] J. Zheng, J. Zhang, and B. Ai, "UAV communications with WPT-aided cell-free massive MIMO Systems," *IEEE J. Sel. Areas in Commun.*, pp. 1–1, Jan. 2021.

[12] J. An and F. Zhao, "Trajectory optimization and power allocation algorithm in MBS-assisted cell-free massive MIMO systems," *IEEE Access*, vol. 9, pp. 30417–30425, 2021.

[13] C. Diaz-Vilor, A. Lozano, and H. Jafarkhani, "Cell-free UAV networks with wireless fronthaul: Analysis and optimization," *IEEE Trans. Wireless Commun.*, pp. 1–1, 2023.

[14] J. Guo, P. Walk, and H. Jafarkhani, "Optimal deployments of UAVs with directional antennas for a power-efficient coverage," *IEEE Trans. Commun.*, vol. 68, pp. 5159–5174, Aug. 2020.

[15] E. Koyuncu, M. Shabanighazikelayeh, and H. Seferoglu, "Deployment and trajectory optimization of UAVs: A quantization theory approach," *IEEE Trans. Wireless Commun.*, vol. 17, pp. 8531–8546, Dec. 2018.

[16] C. Diaz-Vilor and H. Jafarkhani, "Optimal 3D-UAV trajectory and resource allocation of DL UAV-GE links with directional antennas," *2020 IEEE Global Commun. Conf.*, pp. 1–6, Dec. 2020.

[17] J. Guo and H. Jafarkhani, "Movement-efficient sensor deployment in wireless sensor networks with limited communication range," *IEEE Trans. Wireless Commun.*, vol. 18, pp. 3469–3484, July 2019.

[18] Y. Zeng, X. Xu, and R. Zhang, "Trajectory design for completion time minimization in UAV-enabled multicasting," *IEEE Trans. Wireless Commun.*, vol. 17, pp. 2233–2246, Apr. 2018.

[19] Q. Wu, Y. Zeng, and R. Zhang, "Joint trajectory and communication design for multi-UAV enabled wireless networks," *IEEE Trans. Wireless Commun.*, vol. 17, pp. 2109–2121, Mar. 2018.

[20] J. Guo and H. Jafarkhani, "Sensor deployment with limited communication range in homogeneous and heterogeneous wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 15, pp. 6771–6784, Oct. 2016.

[21] C. You and R. Zhang, "3D trajectory optimization in rician fading for UAV-enabled data harvesting," *IEEE Trans. Wireless Commun.*, vol. 18, pp. 3192–3207, Jun. 2019.

[22] J. Guo, E. Koyuncu, and H. Jafarkhani, "A source coding perspective on node deployment in two-tier networks," *IEEE Trans. Commun.*, vol. 66, pp. 3035–3049, Jul. 2018.

[23] F. Cheng, S. Zhang, Z. Li, Y. Chen, N. Zhao, F. R. Yu, and V. C. Leung, "UAV trajectory optimization for data offloading at the edge of multiple cells," *IEEE Trans. Vehicular Technology*, vol. 67, pp. 6732–6736, Jul. 2018.

[24] S. Karimi-Bidhendi, J. Guo, and H. Jafarkhani, "Energy-efficient node deployment in heterogeneous two-tier wireless sensor networks with limited communication range," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 40–55, 2021.

[25] Y. Zeng and R. Zhang, "Energy-efficient UAV communication with trajectory optimization," *IEEE Trans. Wireless Commun.*, vol. 16, pp. 3747–3760, Jun. 2017.

[26] C. Sun, W. Ni, and X. Wang, "Joint computation offloading and trajectory planning for uav-assisted edge computing," *IEEE Trans. Wireless Commun.*, vol. 20, pp. 5343–5358, Mar. 2021.

[27] Y. Zeng, J. Xu, and R. Zhang, "Energy minimization for wireless communication with rotary-wing UAV," *IEEE Trans. Wireless Commun.*, vol. 18, pp. 2329–2345, Apr. 2019.

[28] S. Karimi-Bidhendi *et al.*, "Energy-efficient deployment in static and mobile heterogeneous multi-hop wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 21, no. 7, pp. 4973–4988, 2022.

[29] C. Diaz-Vilor *et al.*, "Sensing and communication in UAV cellular networks: Design and optimization," *IEEE Trans. Wireless Commun.*, pp. 1–1, 2023.

[30] N. V. Cuong, Y.-W. P. Hong, and J.-P. Sheu, "UAV trajectory optimization for joint relay communication and image surveillance," *IEEE Trans. Wireless Commun.*, vol. 21, pp. 10177–10192, Jun. 2022.

[31] S. Venkatesan, A. Lozano, and R. Valenzuela, "Network MIMO: Overcoming intercell interference in indoor wireless systems," in *Asilomar Conf. on Signals, Systems and Computers*, pp. 83–87, 2007.

[32] H. Q. Ngo, A. Ashikhmin, H. Yang, E. G. Larsson, and T. L. Marzetta, "Cell-free massive MIMO versus small cells," *IEEE Trans. Wireless Commun.*, vol. 16, pp. 1834–1850, Mar. 2017.

[33] H. Q. Ngo, L. Tran, T. Q. Duong, M. Matthaiou, and E. G. Larsson, "On the total energy efficiency of cell-free massive MIMO," *IEEE Trans. Green Commun. and Net.*, vol. 2, pp. 25–39, Nov. 2018.

[34] M. Bashar, K. Cumanan, A. G. Burr, M. Debbah, and H. Q. Ngo, "On the uplink max–min SINR of cell-free massive MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 18, pp. 2021–2036, Jan. 2019.

[35] E. Björnson and L. Sanguinetti, "Making cell-free massive MIMO competitive with MMSE processing and centralized implementation," *IEEE Trans. Wireless Commun.*, vol. 19, pp. 77–90, Jan. 2020.

[36] M. Attarifar, A. Abbasfar, and A. Lozano, "Subset MMSE receivers for cell-free networks," *IEEE Trans. Wireless Commun.*, vol. 19, pp. 4183–4194, Jun. 2020.

[37] R. Mosayebi, M. M. Mojahedian, and A. Lozano, "Linear interference cancellation for the cell-free C-RAN uplink," *IEEE Trans. Wireless Commun.*, vol. 20, pp. 1544–1556, Nov. 2021.

[38] G. Interdonato, P. Frenger, and E. G. Larsson, "Scalability aspects of cell-free massive MIMO," in *2019 IEEE Int'l Conf. on Commun. (ICC)*, pp. 1–6, May 2019.

[39] E. Björnson and L. Sanguinetti, "Scalable cell-free MIMO massive systems," *IEEE Trans. Commun.*, vol. 68, pp. 4247–4261, Jul. 2020.

[40] M. Attarifar, A. Abbasfar, and A. Lozano, "Random vs structured pilot assignment in cell-free massive MIMO wireless networks," in *Int'l Conf. Commun. Workshops (ICC Workshops)*, 2018.

[41] C. D'Andrea *et al.*, "Analysis of UAV communications in cell-free massive mimo systems," *IEEE Open Journal of the Commun. Soc.*, vol. 1, pp. 133–147, Jan. 2020.

[42] Ö. T. Demir and E. Björnson, "Joint power control and lsfd for wireless-powered cell-free massive mimo," *IEEE Trans. Wireless Commun.*, vol. 20, pp. 1756–1769, Mar. 2020.

[43] O. Özdogan, E. Björnson, and J. Zhang, "Performance of cell-free massive MIMO with rician fading and phase shifts," *IEEE Trans. Wireless Commun.*, vol. 18, pp. 5299–5315, Nov. 2019.

[44] J. T. Abatzoglou and A. P. Williams, "Impact of anthropogenic climate change on wildfire across western US forests," *Proc. of the National Academy of Sciences*, vol. 113, no. 42, pp. 11770–11775, 2016.

[45] E. Chuvieco *et al.*, "Development of a framework for fire risk assessment using remote sensing and geographic information system technologies," *Ecological modelling*, vol. 221, pp. 46–58, Jan. 2010.

[46] M. P. Thompson and D. E. Calkin, "Uncertainty and risk in wildland fire management: A review," *Journal of Environmental Management*, vol. 92, pp. 1895–1909, Aug. 2011.

[47] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*, vol. 2. MIT press, Nov. 2018.

[48] S. Fujimoto, H. Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *Int'l Conf. on Machine Learning*, pp. 1587–1596, PMLR, 2018.

[49] J. Hu, H. Zhang, and L. Song, "Reinforcement Learning for Decentralized Trajectory Design in Cellular UAV Networks With Sense-and-Send Protocol," *IEEE IoT Journal*, vol. 6, pp. 6177–6189, Aug. 2019.

[50] S. Zhang, H. Zhang, B. Di, and L. Song, "Cellular Cooperative Unmanned Aerial Vehicle Networks with Sense-and-Send Protocol," *IEEE Internet of Things Journal*, vol. 6, pp. 1754–1767, Oct. 2018.

[51] J. Hu, H. Zhang, L. Song, Z. Han, and H. V. Poor, "Reinforcement Learning for a Cellular Internet of UAVs: Protocol Design, Trajectory Control, and Resource Management," *IEEE Wireless Commun.*, vol. 27, pp. 116–123, Feb. 2020.

[52] S. Zhang, H. Zhang, Z. Han, H. V. Poor, and L. Song, "Age of Information in a Cellular Internet of UAVs: Sensing and Communication Trade-Off Design," *IEEE Trans. Wireless Commun.*, vol. 19, pp. 6578–6592, Jun. 2020.

[53] Y. Yang, Z. Zheng, K. Bian, L. Song, and Z. Han, "Real-Time Profiling of Fine-Grained Air Quality Index Distribution Using UAV Sensing," *IEEE Internet of Things Journal*, vol. 5, pp. 186–198, Nov. 2018.

[54] T. Kersnovski, F. Gonzalez, and K. Morton, "A UAV system for autonomous target detection and gas sensing," in *2017 IEEE Aerospace Conf.*, pp. 1–12, Jun. 2017.

[55] L. Yu, N. Wang, and X. Meng, "Real-time forest fire detection with wireless sensor networks," in *Proc. 2005 Intl Conf. on Wireless Commun., Networking and Mobile Computing, 2005.*, vol. 2, pp. 1214–1217, Dec. 2005.

[56] S. Gao, H. Zhang, and S. K. Das, "Efficient data collection in wireless sensor networks with path-constrained mobile sinks," *IEEE Trans. Mobile Computing*, vol. 10, pp. 592–608, Apr. 2011.

[57] M. Sun, X. Xu, X. Qin, and P. Zhang, "AoI-energy-aware UAV-assisted data collection for IoT networks: A deep reinforcement learning method," *IEEE IoT Journal*, vol. 8, pp. 17275–17289, May 2021.

[58] J. Gong *et al.*, "Flight time minimization of UAV for data collection over wireless sensor networks," *IEEE Journal on Selected Areas in Commun.*, vol. 36, pp. 1942–1954, Aug. 2018.

[59] Y. Wang *et al.*, "Trajectory design for UAV-based internet of things data collection: A deep reinforcement learning approach," *IEEE IoT Journal*, vol. 9, pp. 3899–3912, Aug. 2022.

[60] T. L. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE Trans. Wireless Commun.*, vol. 9, pp. 3590–3600, Nov. 2010.

[61] Y. Jing and H. Jafarkhani, "Network beamforming using relays with perfect channel information," *IEEE Trans. Inf. Theory*, vol. 55, pp. 2499–2517, Jun. 2009.

[62] D. Gesbert, S. Hanly, H. Huang, S. Shamai Shitz, O. Simeone, and W. Yu, "Multi-cell MIMO cooperative networks: A new look at interference," *IEEE J. Sel. Areas in Commun.*, vol. 28, pp. 1380–1408, Dec. 2010.

[63] E. Koyuncu and H. Jafarkhani, "Distributed beamforming in wireless multiuser relay-interference networks with quantized feedback," *IEEE Trans. Inf. Theory*, vol. 58, pp. 4538–4576, Jul. 2012.

[64] Y.-W. Hong, W.-J. Huang, F.-H. Chiu, and C.-C. J. Kuo, "Cooperative communications in resource-constrained wireless networks," *IEEE Sig. Process. Mag.*, vol. 24, pp. 47–57, May 2007.

[65] J. Kazemitabar and H. Jafarkhani, "Performance analysis of multiple antenna multi-user detection," in *Inf. Theory and Applications Workshop*, pp. 150–159, Jan. 2009.

[66] G. Nigam, P. Minero, and M. Haenggi, "Coordinated multipoint joint transmission in heterogeneous networks," *IEEE Trans. Commun.*, vol. 62, pp. 4134–4146, Oct. 2014.

[67] A. Adhikary, A. Ashikhmin, and T. L. Marzetta, "Uplink interference reduction in large-scale antenna systems," *IEEE Trans. Commun.*, vol. 65, pp. 2194–2206, Jan. 2017.

[68] M. Jiang and L. Hanzo, "Multiuser MIMO-ofdm for next-generation wireless systems," *Proced. of the IEEE*, vol. 95, pp. 1430–1469, Aug. 2007.

[69] M. Alzenad, A. El-Keyi, F. Lagum, and H. Yanikomeroglu, "3-d placement of an unmanned aerial vehicle base station (UAV-BS) for energy-efficient maximal coverage," *IEEE Wireless Commun. Lett.*, vol. 6, pp. 434–437, Aug. 2017.

[70] G. Zhang, H. Yan, Y. Zeng, M. Cui, and Y. Liu, "Trajectory optimization and power allocation for multi-hop UAV relaying communications," *IEEE Access*, vol. 6, pp. 48566–48576, Aug. 2018.

[71] S. Zhang, H. Zhang, Q. He, K. Bian, and L. Song, "Joint trajectory and power optimization for UAV relay networks," *IEEE Commun. Lett.*, vol. 22, pp. 161–164, Jan. 2018.

[72] A. A. Khuwaja, Y. Chen, N. Zhao, M. S. Alouini, and P. Dobbins, "A survey of channel modeling for UAV communications," *IEEE Commun. Surveys and Tutorials*, vol. 20, pp. 2804–2821, Oct. 2018.

[73] S. Shimamoto and Iskandar, "Channel characterization and performance evaluation of mobile communication employing stratospheric platforms," *IEICE Trans. Commun.*, vol. E89-B, pp. 937–944, Mar. 2006.

[74] A. A. Khuwaja, Y. Chen, N. Zhao, M. Alouini, and P. Dobbins, "A survey of channel modeling for UAV communications," *IEEE Commun. Surveys Tutorials*, vol. 20, pp. 2804–2821, Apr. 2018.

[75] D. W. Matolak and R. Sun, "Air–ground channel characterization for unmanned aircraft systems—part i: Methods, measurements, and models for over-water settings," *IEEE Trans. Vehicular Technology*, vol. 66, pp. 26–44, Jan. 2017.

[76] R. Sun and D. W. Matolak, "Air–ground channel characterization for unmanned aircraft systems part ii: Hilly and mountainous settings," *IEEE Trans. Vehicular Technology*, vol. 66, pp. 1913–1925, Mar. 2017.

[77] D. W. Matolak and R. Sun, "Air–ground channel characterization for unmanned aircraft systems—part iii: The suburban and near-urban environments," *IEEE Trans. Vehicular Technology*, vol. 66, pp. 6607–6618, Jan. 2017.

[78] C. A. Balanis, *Antenna Theory : Analysis and Design*. Wiley, 4th ed., 2016.

[79] W. Hachem, M. Kharouf, J. Najim, and J. W. Silverstein, "A CLT for information-theoretic statistics of non-centered gram random matrices," *Random Matrices: Theory and Applications*, vol. 1, no. 02, pp. 1–50, 2012.

[80] W. Hachem, P. Loubaton, J. Najim, and P. Vallet, "On bilinear forms based on the resolvent of large random matrices," in *Annales de l'IHP Probabilités et statistiques*, vol. 49, pp. 36–63, 2013.

[81] A. Kammoun, L. Sanguinetti, M. Debbah, and M.-S. Alouini, "Asymptotic analysis of RZF in large-scale MU-MIMO systems over rician channels," *IEEE Trans. Inf. Th.*, vol. 65, no. 11, pp. 7268–7286, 2019.

[82] S. Wagner, R. Couillet, M. Debbah, and D. T. M. Slock, "Large system analysis of linear precoding in correlated MISO broadcast channels under limited feedback," *IEEE Trans. Inf. Th.*, vol. 58, pp. 4509–4537, Mar. 2012.

[83] J. Hoydis, S. ten Brink, and M. Debbah, "Massive MIMO in the UL/DL of cellular networks: How many antennas do we need?," *IEEE J. on Sel. Areas in Commun.*, vol. 31, pp. 160–171, Feb. 2013.

[84] P. Bremaud, *Markov Chains: Gibbs Fields, Monte Carlo Simulation, and Queues*, vol. 31. Springer, 1999.

[85] L. P. Qian, Y. J. A. Zhang, and M. Chiang, "Distributed nonconvex power control using gibbs sampling," *IEEE Trans. Commun.*, vol. 60, pp. 3886–3898, Dec. 2012.

[86] R. W. Heath Jr. and A. Lozano, *Foundations of MIMO communication*. Cambridge University Press, 2019.

[87] Q. Sun, D. C. Cox, A. Lozano, and H. C. Huang, "Training-based channel estimation for continuous flat fading BLAST," in *IEEE Int'l Conf. Commun. (ICC)*, vol. 1, pp. 325–329, 2002.

[88] A. Lozano, "Interplay of spectral efficiency, power and Doppler spectrum for reference-signal-assisted wireless communication," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 5020–5029, 2008.

[89] H. B. Mann and A. Wald, "On stochastic limit and order relationships," *Annals of Mathematical Statistics*, vol. 14, pp. 217–226, 1943.

[90] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Unmanned aerial vehicle with underlaid device-to-device communications: Performance and tradeoffs," *IEEE Trans. Wireless Commun.*, vol. 15, pp. 3949–3963, Jun. 2016.

[91] P. Chandhar, D. Danev, and E. G. Larsson, "Massive MIMO for communications with drone swarms," *IEEE Trans. Wireless Commun.*, vol. 17, pp. 1604–1629, March 2018.

[92] "Enhanced LTE support for aerial vehicles," Tech. Rep. 36.777, 3GPP, Dec. 2017.

[93] A. Al-Hourani and K. Gomez, "Modeling cellular-to-UAV path-loss for suburban environments," *IEEE Wireless Commun. Lett.*, vol. 7, no. Jan., pp. 82–85, 2018.

[94] R. Nikbakht, R. Mosayebi, and A. Lozano, "Uplink fractional power control and downlink power allocation for cell-free networks," *IEEE Wireless Commun. Letters*, vol. 9, no. 6, pp. 774–777, 2020.

[95] W. Jiang, B. Han, M. A. Habibi, and H. D. Schotten, "The road towards 6G: A comprehensive survey," *IEEE Open Journal of the Commun. Soc.*, vol. 2, pp. 334–366, Feb. 2021.

[96] Y. Jing and H. Jafarkhani, "Network beamforming with channel means and covariances at relays," in *2008 IEEE Int'l Conf. on Commun.*, pp. 3743–3747, 2008.

[97] A. Checko, H. L. Christiansen, Y. Yan, L. Scolari, G. Kardaras, M. S. Berger, and L. Dittmann, "Cloud RAN for mobile networks—a technology overview," *IEEE Commun. Surveys Tutorials*, vol. 17, pp. 405–426, Sept. 2015.

[98] J. Wu *et al.*, "Cloud radio access network (C-RAN): a primer," *IEEE Network*, vol. 29, pp. 35–41, Jan. 2015.

[99] P. Rost, C. J. Bernardos, A. D. Domenico, M. D. Girolamo, M. Lalam, A. Maeder, D. Sabella, and D. Wübben, "Cloud technologies for flexible 5g radio access networks," *IEEE Commun. Magazine*, vol. 52, pp. 68–76, Sept. 2014.

[100] V. Tentu, E. Sharma, D. N. Amudala, and R. Budhiraja, "UAV-enabled hardware-impaired spatially correlated cell-free massive MIMO systems: Analysis and energy efficiency optimization," *IEEE Trans. Commun*, vol. 70, no. 4, pp. 2722–2741, 2022.

[101] B. Wang, J. Zhang, and A. Host-Madsen, "On the capacity of MIMO relay channels," *IEEE Trans. Inform. theory*, vol. 51, no. 1, pp. 29–43, 2005.

[102] X. Wang, J. Cheng, C. Zhai, and A. Ashikhmin, "Partial cooperative zero-forcing decoding for uplink cell-free massive MIMO," *IEEE Internet of Things Journal*, vol. 9, pp. 10327–10339, Oct. 2022.

[103] X. Wang, A. Ashikhmin, Z. Dong, and C. Zhai, "Two-stage channel estimation approach for cell-free iot with massive random access," *IEEE J. on Sel. Areas in Commun.*, vol. 40, pp. 1428–1440, Jan. 2022.

[104] L. Zheng *et al.*, "Wireless max–min utility fairness with general monotonic constraints by perron–frobenius theory," *IEEE Trans. Information Th.*, vol. 62, pp. 7283–7298, Dec. 2016.

[105] M. Diehl, F. Glineur, E. Jarlebring, and W. Michiels, *Recent Advances in optimization and its applications in engineering*. Springer Berlin Heidelberg, 2010.

[106] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004.

[107] J. Cohen, "The wildland-urban interface fire problem," *Fremontia*, vol. 38, no. 2, p. 3, 2010.

[108] M. A. Finney *et al.*, "A simulation of probabilistic wildfire risk components for the continental united states," *Stochastic Environmental Research and Risk Assessment*, vol. 25, pp. 973–1000, Mar. 2011.

[109] G. Perry, "Current approaches to modelling the spread of wildland fire: a review," *Progress in Physical Geography: Earth and Environment*, vol. 22, no. 2, p. 222–245, 1998.

[110] M. A. Finney, "FARSITE: Fire area simulator-model development and evaluation," tech. rep., 1998.

[111] E. Seraj and M. Gombolay, "Coordinated control of UAVs for human-centered active sensing of wildfires," in *2020 American Control Conf. (ACC)*, pp. 1845–1852, Jul. 2020.

[112] K. A. Ghamry *et al.*, "Cooperative control of multiple UAVs for forest fire monitoring and detection," in *IEEE Int'l Conf. on Mechatronic and Embedded Systems and Applications*, Oct. 2016.

[113] K. A. Ghamry and Y. Zhang, "Fault-tolerant cooperative control of multiple UAVs for forest fire detection and tracking mission," in *Conf. on Control and Fault-Tolerant Systems (SysTol)*, pp. 133–138, 2016.

[114] E. Seraj, A. Silva, and M. Gombolay, "Multi-UAV planning for cooperative wildfire coverage and tracking with quality-of-service guarantees," *Autonomous Agents and Multi-Agent Systems*, vol. 36, no. 2, p. 39, 2022.

[115] C. Yuan, Z. Liu, and Y. Zhang, "Fire detection using infrared images for UAV-based forest fire surveillance," in *2017 Int'l Conf. on Unmanned Aircraft Systems (ICUAS)*, pp. 567–572, IEEE, 2017.

[116] C. Yuan *et al.*, "UAV-based forest fire detection and tracking using image processing techniques," in *2015 Int'l Conf. on Unmanned Aircraft Systems (ICUAS)*, pp. 639–643, IEEE, 2015.

[117] H. Ucgun *et al.*, "A review on applications of rotary-wing unmanned aerial vehicle charging stations," *Int'l Journal of Advanced Robotic Systems*, vol. 18, p. 17298814211015863, May 2021.

[118] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Proc. Magazine*, vol. 34, pp. 26–38, Nov. 2017.

[119] V. Mnih *et al.*, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[120] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," *Advances in Neural Information Proc. Systems*, vol. 12, 1999.

[121] V. Konda and J. Tsitsiklis, "Actor-critic algorithms," *Advances in Neural Information Proc. Systems*, vol. 12, 1999.

[122] V. Mnih *et al.*, "Asynchronous Methods for Deep Reinforcement Learning," in *Int'l Conf. on Machine Learning*, pp. 1928–1937, PMLR, 2016.

[123] T. P. Lillicrap *et al.*, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.

[124] H. Yang *et al.*, "Privacy-preserving federated learning for UAV-enabled networks: Learning-based joint scheduling and resource management," *IEEE J. on Sel. Areas in Commun.*, vol. 39, pp. 3144–3159, Jun. 2021.

[125] Y. Yu *et al.*, "Multi-objective optimization for UAV-assisted wireless powered IoT networks based on extended DDPG algorithm," *IEEE Trans. Commun.*, vol. 69, pp. 6361–6374, Jun. 2021.

[126] R. Ding, F. Gao, and X. S. Shen, "3D UAV Trajectory Design and Frequency Band Allocation for Energy-Efficient and Fair Communication: A Deep Reinforcement Learning Approach," *IEEE Trans. Wireless Commun.*, vol. 19, pp. 7796–7809, Aug. 2020.

[127] Z. Xia *et al.*, "Multi-agent reinforcement learning aided intelligent UAV swarm for target tracking," *IEEE Trans. Vehicular Tech.*, vol. 71, pp. 931–945, Nov. 2022.

[128] R. Ding *et al.*, "Trajectory design and access control for air–ground coordinated communications system with multiagent deep reinforcement learning," *IEEE IoT Journal*, vol. 9, pp. 5785–5798, Feb. 2022.

[129] D. Hong *et al.*, "Energy-efficient online path planning of multiple drones using reinforcement learning," *IEEE Trans. Vehicular Tech.*, vol. 70, pp. 9725–9740, Aug. 2021.

[130] Y. Li and A. H. Aghvami, "Radio resource management for cellular-connected UAV: A learning approach," *IEEE Trans. Commun.*, vol. 71, pp. 2784–2800, Mar. 2023.

[131] M. Schwager, B. J. Julian, M. Angermann, and D. Rus, "Eyes in the sky: Decentralized control for the deployment of robotic camera networks," *Proceedings of the IEEE*, vol. 99, pp. 1541–1561, July 2011.

[132] Q. Sun, D. Cox, A. Lozano, and H. Huang, "Training-based channel estimation for continuous flat fading BLAST," in *IEEE Int'l Conf. Commun. (ICC)*, vol. 1, pp. 325–329, 2002.

[133] C. Guestrin, D. Koller, and R. Parr, "Multiagent planning with factored mdps," in *Advances in Neural Information Processing Systems* (T. Dietterich, S. Becker, and Z. Ghahramani, eds.), vol. 14, MIT Press, 2001.

[134] T. Degris, O. Sigaud, and P.-H. Wuillemin, "Learning the structure of factored markov decision processes in reinforcement learning problems," in *Proc. of the 23rd Int'l Conf. on Machine Learning*, (New York, NY, USA), p. 257–264, Association for Computing Machinery, 2006.

[135] A. Y. Ng *et al.*, "Policy invariance under reward transformations: Theory and application to reward shaping," in *Proc. of the Sixteenth Int'l Conf. on Machine Learning*, ICML '99, (San Francisco, CA, USA), p. 278–287, Morgan Kaufmann Publishers Inc., 1999.

[136] J. Guo and H. Jafarkhani, "Sensor Deployment with Limited Communication Range in Homogeneous and Heterogeneous Wireless Sensor Networks," *IEEE Trans. Wireless Commun.*, vol. 15, pp. 6771–6784, Jul. 2016.

[137] Z. Na, C. Ji, B. Lin, and N. Zhang, "Joint Optimization of Trajectory and Resource Allocation in Secure UAV Relaying Communications for Internet of Things," *IEEE IoT Journal*, vol. 9, pp. 16284–16296, Feb. 2022.

[138] Z. Wang, R. Liu, Q. Liu, J. S. Thompson, and M. Kadoch, "Energy-Efficient Data Collection and Device Positioning in UAV-Assisted IoT," *IEEE IoT Journal*, vol. 7, pp. 1122–1139, Nov. 2020.

[139] T. Liu, G. Zhang, M. Cui, C. You, Q. Wu, S. Ma, and W. Chen, "Task Completion Time Minimization for UAV-Enabled Data Collection in Rician Fading Channels," *IEEE IoT Journal*, vol. 10, no. 2, pp. 1134–1148, 2023.

[140] C. Zhan and Y. Zeng, "Completion Time Minimization for Multi-UAV-Enabled Data Collection," *IEEE Trans. Wireless Commun.*, vol. 18, pp. 4859–4872, Jul. 2019.

[141] A. Hossain, P. K. Biswas, and S. Chakrabarti, "Sensing Models and Its Impact on Network Coverage in Wireless Sensor Network," in *2008 IEEE Region 10 and the 3rd. Int. Conf. on Industrial and Information Systems*, pp. 1–5, 2008.

[142] Y. Zou and K. Chakrabarty, "A distributed coverage- and connectivity-centric technique for selecting active nodes in wireless sensor networks," *IEEE Trans. Computers*, vol. 54, pp. 978–991, Jun. 2005.

[143] A. Chakraborty, R. R. Rout, A. Chakrabarti, and S. K. Ghosh, "On Network Lifetime Expectancy With Realistic Sensing and Traffic Generation Model in Wireless Sensor Networks," *IEEE Sensors Journal*, vol. 13, pp. 2771–2779, Jul. 2013.

[144] J. Kazemitabar and H. Jafarkhani, "Multiuser interference cancellation and detection for users with more than two transmit antennas," *IEEE Trans. Commun.*, vol. 56, pp. 574–583, Apr. 2008.

[145] W. Mei, Q. Wu, and R. Zhang, "Cellular-connected UAV: Uplink association, power control and interference coordination," *IEEE Trans. Wireless Commun.*, vol. 18, pp. 5380–5393, Nov. 2019.

[146] W. Mei and R. Zhang, "Uplink cooperative NOMA for cellular-connected UAV," *IEEE Journal of Sel. Topics in Sig. Proc.*, vol. 13, pp. 644–656, Feb. 2019.

[147] Z. Luo and P. Tseng, "On the convergence of the coordinate descent method for convex differentiable minimization," *J. of Optimization Theory and Applications*, vol. 72, pp. 7–35, Jan. 1992.

[148] H. Yousefi'zadeh, H. Jafarkhani, and M. Moshfeghi, "Power optimization of wireless media systems with space-time block codes," *IEEE Trans. Image Proc.*, vol. 13, pp. 873–884, Jun 2004.

[149] W. Liu *et al.*, "Energy harvesting wireless sensor networks: Delay analysis considering energy costs of sensing and transmission," *IEEE Trans. Wireless Commun*, vol. 15, pp. 4635–4650, Mar. 2016.

[150] Z. Sheng, C. Mahapatra, V. C. Leung, M. Chen, and P. K. Sahu, "Energy efficient cooperative computing in mobile wireless sensor networks," *IEEE Trans. Cloud Computing*, vol. 6, pp. 114–126, Jul. 2015.

[151] J. E. Mitchell, "Branch-and-cut algorithms for combinatorial optimization problems," *Handbook of applied optimization*, vol. 1, no. 1, pp. 65–77, 2002.

[152] M. Hua, L. Yang, Q. Wu, and A. L. Swindlehurst, "3D UAV Trajectory and Communication Design for Simultaneous Uplink and Downlink Transmission," *IEEE Trans. Commun.*, vol. 68, pp. 5908–5923, Jun. 2020.

[153] Y. Xiao, G. Shi, Y. Li, W. Saad, and H. V. Poor, "Toward self-learning edge intelligence in 6G," *IEEE Commun. Magazine*, vol. 58, pp. 34–40, Dec. 2020.

[154] S. Karimi-Bidhendi and H. Jafarkhani, "Outage-aware deployment in heterogeneous Rayleigh fading wireless sensor networks," *IEEE Trans. Commun.*, vol. 72, no. 2, pp. 1146–1161, 2024.

[155] A. M. Abdelhady, A. Celik, C. Diaz-Vilor, H. Jafarkhani, and A. M. Eltawil, "Laser-empowered UAVs for aerial data aggregation in passive iot networks," *IEEE Open Journal of the Commun. Society*, pp. 1–1, 2024.

[156] Q. Wu and R. Zhang, "Common throughput maximization in UAV-enabled OFDMA systems with delay consideration," *IEEE Trans. Commun.*, vol. 66, pp. 6614–6627, Aug. 2018.

[157] S. Arzykulov, A. Celik, G. Nauryzbayev, and A. M. Eltawil, "UAV-assisted cooperative cognitive NOMA: Deployment, clustering, and resource allocation," *IEEE Trans. Cognitive Commun. and Networking*, vol. 8, no. 1, pp. 263–281, 2022.

[158] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?," in *2012 Proceedings IEEE INFOCOM*, pp. 2731–2735, 2012.

[159] M. A. Abd-Elmagid and H. S. Dhillon, "Average peak age-of-information minimization in UAV-assisted iot networks," *IEEE Trans. Vehicular Technology*, vol. 68, pp. 2003–2008, Feb. 2019.

[160] J. Liu, X. Wang, B. Bai, and H. Dai, "Age-optimal trajectory planning for UAV-assisted data collection," in *IEEE INFOCOM 2018*, pp. 553–558, 2018.

[161] H. Jafarkhani, *Space-Time Coding: Theory and Practice*. Cambridge University Press, 2005.

[162] K. Guo *et al.*, "Uplink power control with MMSE receiver in multi-cell MU-massive-MIMO systems," in *2014 IEEE Int'l Conf. on Commun. (ICC)*, pp. 5184–5190, 2014.

[163] R. Chen, J. G. Andrews, R. W. Heath, and A. Ghosh, "Uplink power control in multi-cell spatial multiplexing wireless systems," *IEEE Trans. Wireless Commun.*, vol. 6, pp. 2700–2711, Jul. 2007.

[164] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

Appendix A

Supplementari Proofs

Theorem 7. *([82, Theorem 1]) Let $\boldsymbol{D} \in \mathbb{C}^{M\times M}$ be known and deterministic. In addition, $\boldsymbol{S} \in \mathbb{C}^{M\times M}$ is Hermitian nonnegative-definite while $\boldsymbol{H} \in \mathbb{C}^{M\times M}$ is a random matrix with zero-mean independent column vectors, $\boldsymbol{h}_k$, each with covariance matrix $\frac{1}{M}\mathbf{R}_k$. Finally, $\boldsymbol{D}$ and $\boldsymbol{R}_k$ have uniformly bounded spectral norm w.r.t. M. For $z > 0$ and $M, K \to \infty$,*

$$\frac{1}{M}\operatorname{tr}\big[\boldsymbol{D}\big(\boldsymbol{H}\boldsymbol{H}^* + \boldsymbol{S} + z\boldsymbol{I}_M\big)^{-1}\big] - \frac{1}{M}\operatorname{tr}[\boldsymbol{D}\boldsymbol{T}] \overset{a.s.}{\to} 0,$$

where

$$\boldsymbol{T} = \left(\frac{1}{M}\sum_{j=1}^{K}\frac{\boldsymbol{R}_j}{1+e_j} + \boldsymbol{S} + z\boldsymbol{I}_M\right)^{-1} \tag{A.1}$$

with coefficients $e_k = lim_{n\to\infty} e_k^{(n)}$ for

$$e_k^{(n)} = \frac{1}{M}\operatorname{tr}\left[\boldsymbol{R}_k\left(\frac{1}{M}\sum_{j=1}^{K}\frac{\boldsymbol{R}_j}{1+e_j^{(n-1)}} + \boldsymbol{S} + z\boldsymbol{I}_M\right)^{-1}\right] \tag{A.2}$$

with initial values $e_k^{(0)} = M$.

Theorem 8. *([82, Theorem 2]) Let $\boldsymbol{\Phi} \in \mathbb{C}^{M\times M}$ be Hermitian nonnegative-definite. Under the same conditions as Thm. 7, for $M, K \to \infty$,*

$$\frac{1}{M}\operatorname{tr}\big[\boldsymbol{D}\big(\boldsymbol{H}\boldsymbol{H}^* + \boldsymbol{S} + z\boldsymbol{I}_M\big)^{-1}\boldsymbol{\Phi}\big(\boldsymbol{H}\boldsymbol{H}^* + \boldsymbol{S} + z\boldsymbol{I}_M\big)^{-1}\big] - \frac{1}{M}\operatorname{tr}[\boldsymbol{D}\boldsymbol{T}'(z,\boldsymbol{\Phi})] \overset{a.s.}{\to} 0,$$

where $\boldsymbol{T}'(z,\boldsymbol{\Phi})$ is defined as

$$\boldsymbol{T}'(z,\boldsymbol{\Phi}) = \boldsymbol{T}\boldsymbol{\Phi}\boldsymbol{T} + \boldsymbol{T}\frac{1}{M}\sum_{k=1}^{K}\frac{\boldsymbol{R}_k e_k'(z,\boldsymbol{\Phi})}{(1+e_k)^2}\boldsymbol{T}, \tag{A.3}$$

with $\boldsymbol{T}$ and e_k given in Thm. 7 for given z and $\boldsymbol{e}'(z,\boldsymbol{\Phi}) = \big(e_1'(z),\ldots,e_K'(z)\big)$ computed as

$$\boldsymbol{e}'(z,\boldsymbol{\Phi}) = \big(\boldsymbol{I} - \boldsymbol{J}(z)\big)^{-1}\boldsymbol{v}(z,\boldsymbol{\Phi}), \tag{A.4}$$

with $\boldsymbol{J}(z) \in \mathbb{C}^{K\times K}$ and $\boldsymbol{v}(z) \in \mathbb{C}^{K\times 1}$ defined as

$$\big(\boldsymbol{J}(z)\big)_{k,l} = \frac{\frac{1}{M}\operatorname{tr}\big[\boldsymbol{R}_k\boldsymbol{T}\boldsymbol{R}_l\boldsymbol{T}\big]}{M(1+e_l)^2}, \tag{A.5}$$

and

$$\big(\boldsymbol{v}(z,\boldsymbol{\Phi})\big)_k = \frac{1}{M}\operatorname{tr}\big[\boldsymbol{R}_k\boldsymbol{T}\boldsymbol{\Phi}\boldsymbol{T}\big]. \tag{A.6}$$

A.1 Proof of Prop. 1

Let us define the matrices

$$\boldsymbol{\Omega}_k = \left(\big(\boldsymbol{M}^{(s)}\circ\hat{\boldsymbol{G}}_k\big)\boldsymbol{P}\big(\boldsymbol{M}^{(s)}\circ\hat{\boldsymbol{G}}_k\big)^* - \big(\boldsymbol{m}_k^{(s)}\circ\hat{\boldsymbol{g}}_k\big)\big(\boldsymbol{m}_k^{(s)}\circ\hat{\boldsymbol{g}}_k\big)^* p + \boldsymbol{\Sigma}_k\right)^{-1} \tag{A.7}$$

and $\boldsymbol{\Omega}_k^{'} = |\mathcal{F}_k|, \boldsymbol{\Omega}_k$. Then, (2.23) can be written as

$$\mathrm{SINR}_k = \hat{\boldsymbol{g}}_k^* \boldsymbol{\Omega}_k \hat{\boldsymbol{g}}_k \, p \tag{A.8}$$

$$= \frac{p}{|\mathcal{F}_k|} \operatorname{tr}\left[\hat{\boldsymbol{g}}_k \hat{\boldsymbol{g}}_k^* \boldsymbol{\Omega}_k^{'}\right]. \tag{A.9}$$

For $|\mathcal{F}_k|$,$|\mathcal{U}_m| \to \infty \; \forall \, k, m$, using [82, Lemma 4] and Theorem 7, we have

$$\frac{p}{|\mathcal{F}_k|} \operatorname{tr}\left[\hat{\boldsymbol{g}}_k \hat{\boldsymbol{g}}_k^* \boldsymbol{\Omega}_k^{'}\right] - \frac{p}{|\mathcal{F}_k|} \operatorname{tr}[\boldsymbol{\Gamma}_k \boldsymbol{T}_k] \overset{\text{a.s.}}{\to} 0. \tag{A.10}$$

In our case, the role of $\left(\boldsymbol{H}\boldsymbol{H}^* + \boldsymbol{S} + z\boldsymbol{I}_M\right)^{-1}$ in Theorem 7 is played by $\boldsymbol{\Omega}_k^{'}$. There is a direct mapping between the terms in the aforementioned theorem and our problem, namely (i) $\boldsymbol{D} = \boldsymbol{\Gamma}_k \, p$, (ii) $\boldsymbol{R}_j = \boldsymbol{\Gamma}_j \, p$, and (iii) $\boldsymbol{S} + z\boldsymbol{I}_M = \frac{1}{|\mathcal{F}_k|}\boldsymbol{\Sigma}_k$ with matrix $\boldsymbol{T}_k$ following the structure of $\boldsymbol{T}$ in Theorem 7, i.e.,

$$\boldsymbol{T}_k = \left(\frac{1}{|\mathcal{F}_k|} \sum_{j \neq k}^{K} \frac{\boldsymbol{\Gamma}_j}{1 + e_{j,k}} \, p + \frac{1}{|\mathcal{F}_k|} \boldsymbol{\Sigma}_k\right)^{-1}, \tag{A.11}$$

where the contribution of GU k is removed as in $\boldsymbol{\Omega}_k^{'}$. The necessary coefficients can be calculated as $e_{j,k} = \lim_{n \to \infty} e_{j,k}^{(n)}$ with

$$e_{j,k}^{(n)} = p \operatorname{tr}\left[\boldsymbol{\Gamma}_j \left(\sum_{i \neq k}^{K} \frac{\boldsymbol{\Gamma}_i}{1 + e_{i,k}^{(n-1)}} \, p + \boldsymbol{\Sigma}_k\right)^{-1}\right] \tag{A.12}$$

$$= \sum_{m \in \mathcal{F}_k} \frac{m_{k,j,m}^{(s)} \gamma_{j,m}}{\sum\limits_{\substack{i \in \mathcal{U}_m \\ i \neq k}} \frac{\gamma_{i,m}}{1 + e_{i,k}^{(n-1)}} + \sum\limits_{i \in \mathcal{U}_m} c_{i,m} + \sum\limits_{i \notin \mathcal{U}_m} r_{i,m} + \frac{\sigma^2}{p}}, \tag{A.13}$$

where $m_{k,j,m}^{(s)} = \nVdash\{j \in \mathcal{U}_m \text{ for } m \in \mathcal{F}_k\}$. The fixed-point algorithm can be used to compute $e_{j,k}^{(n)}$ and has been proved to converge [82]. Finally, since matrices $\boldsymbol{\Gamma}_k$ and $\boldsymbol{T}_k$ are diagonal,

(A.34) can be written as

$$\frac{p}{|\mathcal{F}_k|}\mathrm{tr}[\boldsymbol{\Gamma}_k\boldsymbol{T}_k] = p\,\mathrm{tr}\left[\boldsymbol{\Gamma}_k\Big(\sum_{j\neq k}^{K}\frac{\boldsymbol{\Gamma}_j}{1+e_{j,k}}\,p+\boldsymbol{\Sigma}\Big)^{-1}\right] \tag{A.14}$$

which yields the expression in Prop. 1 after some straightforward algebra.

A.2 Proof of Prop. 2

From (2.24), we can compute the deterministic equivalence of each term in the numerator and denominator separately, letting $|\mathcal{F}_k|, |\mathcal{U}_m| \to \infty$. We divide each by $1/|\mathcal{F}_k|^2$ and recall that $\hat{g}_{k,m} = \overline{g}_{k,m} + \zeta_{k,m}$ with the distribution of $\zeta_{k,m}$ derived in (2.10). Then, the signal satisfies

$$\frac{1}{|\mathcal{F}_k|}\hat{\boldsymbol{g}}_k^*\hat{\boldsymbol{g}}_k - \frac{1}{|\mathcal{F}_k|}\big(\boldsymbol{g}_k^*\boldsymbol{g}_k + \mathrm{tr}\boldsymbol{\Gamma}_k\big) \overset{\text{a.s.}}{\to} 0, \tag{A.15}$$

as a consequence of [82, Lemma 4]. Therefore, from the continuous mapping theorem,

$$\left(\frac{1}{|\mathcal{F}_k|}\hat{\boldsymbol{g}}_k^*\hat{\boldsymbol{g}}_k\right)^2 - \left(\frac{1}{|\mathcal{F}_k|}\big(\boldsymbol{g}_k^*\boldsymbol{g}_k + \mathrm{tr}\boldsymbol{\Gamma}_k\big)\right)^2 \overset{\text{a.s.}}{\to} 0. \tag{A.16}$$

For the interfering terms, a similar procedure is used. Applying [82, Lemma 4] twice, the interfering terms follow

$$\begin{aligned}\frac{1}{|\mathcal{F}_k|^2}&|(\boldsymbol{m}_k^{(s)}\circ\hat{\boldsymbol{g}}_k)^*(\boldsymbol{m}_{k,i}^{(s)}\circ\hat{\boldsymbol{g}}_i)|^2 \\ &- \frac{1}{|\mathcal{F}_k|^2}\mathrm{tr}\Big[\big((\boldsymbol{m}_{k,i}^{(s)}\circ\overline{\boldsymbol{g}}_i)(\boldsymbol{m}_{k,i}^{(s)}\circ\overline{\boldsymbol{g}}_i)^* + \boldsymbol{\Gamma}_i\big)(\overline{\boldsymbol{g}}_k\overline{\boldsymbol{g}}_k^* + \boldsymbol{\Gamma}_k)\Big] \overset{\text{a.s.}}{\to} 0.\end{aligned} \tag{A.17}$$

Finally, following a similar approach for the noise,

$$\frac{1}{|\mathcal{F}_k|^2}\hat{\boldsymbol{g}}_k^*\boldsymbol{\Sigma}_k\hat{\boldsymbol{g}}_k - \frac{1}{|\mathcal{F}_k|^2}\mathrm{tr}\big[(\overline{\boldsymbol{g}}_k\overline{\boldsymbol{g}}_k^* + \boldsymbol{\Gamma}_k)\boldsymbol{\Sigma}_k\big] \overset{\text{a.s.}}{\to} 0. \tag{A.18}$$

Plugging the derived deterministic equivalents into (2.24), Prop. 2 is obtained.

A.3 Gradient Computations

We first calculate some partial derivatives needed for (2.37). We do so with respect to x_m, and a similar procedure can be used for y_m. First, note that (2.3) can be written as $\overline{g}_{k,m} = \big(b_m \frac{1}{d_{k,m}^{\alpha_m+\kappa}} \frac{K_{k,m}}{K_{k,m}+1}\big)^b$ with $b_m = (\alpha_m + 1)\,\beta_0\,H^{\alpha_m}$ and $b = 0.5$. Then,

$$\frac{\partial \overline{g}_{k,m}}{\partial x_m} = b\overline{g}_{k,m}\frac{d_{k,m}^{\alpha_m+\kappa}\,\frac{\partial K_{k,m}}{\partial x_m} - K_{k,m}\frac{\partial d_{k,m}^{\alpha_m+\kappa}}{\partial x_m}(1+K_{k,m})}{d_{k,m}^{\alpha_m+\kappa}K_{k,m}(1+K_{k,m})}, \tag{A.19}$$

with

$$\frac{\partial K_{k,m}}{\partial x_m} = \frac{-A_2 H K_{k,m}}{d_{k,m}^2\sqrt{1-\frac{H^2}{d_{k,m}^2}}} \cdot \frac{x_m - \mathrm{x}_k}{d_{k,m}}, \tag{A.20}$$

and

$$\frac{\partial d_{k,m}^{\alpha_m+\kappa}}{\partial x_m} = (\alpha_m+\kappa)d_{k,m}^{\alpha_m+\kappa-2} \cdot \frac{x_m - \mathrm{x}_k}{d_{k,m}}. \tag{A.21}$$

Similar to (A.19), we can obtain $\frac{\partial r_{k,m}}{\partial x_m}$ with $\frac{\partial \gamma_{k,m}}{\partial x_m}$ as reported in [10]. Note that Num_m can be expressed as $\mathrm{Num}_m = \big(\sum_{m\in\mathcal{F}_k}|\overline{g}_{k,m}|^2 + \gamma_{k,m}\big)^2$ with derivative

$$\frac{\partial \mathrm{Num}_m}{\partial x_m} = 2\left(\sum_{m\in\mathcal{F}_k}|\overline{g}_{k,m}|^2 + \gamma_{k,m}\right)\left(2\overline{g}_{k,m}\frac{\partial \overline{g}_{k,m}}{\partial x_m} + \frac{\partial \gamma_{k,m}}{\partial x_m}\right). \tag{A.22}$$

To calculate the derivative of Den_m, we first need to manipulate the interference and noise terms. A given interference term can be decomposed as

$$\mathrm{tr}\Big[\big((\boldsymbol{m}_{k,i}^{(s)} \circ \overline{\boldsymbol{g}}_i)(\boldsymbol{m}_{k,i}^{(s)} \circ \overline{\boldsymbol{g}}_i)^* + \boldsymbol{\Gamma}_i\big)(\overline{\boldsymbol{g}}_k \overline{\boldsymbol{g}}_k^* + \boldsymbol{\Gamma}_k)\Big] = \underbrace{\left(\sum_{m\in\mathcal{F}_k} m_{k,i,m}^{(s)} \overline{g}_{i,m} \overline{g}_{k,m}\right)^2}_{\mathrm{T1}}$$
$$+ \underbrace{\sum_{m\in\mathcal{F}_k} m_{k,i,m}^{(s)} |\overline{g}_{k,m}|^2 \gamma_{i,m}}_{\mathrm{T2}} + \underbrace{\sum_{m\in\mathcal{F}_k} m_{k,i,m}^{(s)} |\overline{g}_{i,m}|^2 \gamma_{k,m}}_{\mathrm{T3}} + \underbrace{\sum_{m\in\mathcal{F}_k} m_{k,i,m}^{(s)} \gamma_{i,m}\gamma_{k,m}}_{\mathrm{T4}}. \quad \text{(A.23)}$$

The calculation of the derivatives for each term is straightforward, involving only polynomial terms that depend on $\overline{g}_{i,m}$ and $\gamma_{i,m}$, whose derivatives are derived in (A.19) and [10], respectively. After some algebra, the derivative of each interfering term is obtained. Finally, the noise can be decomposed as

$$\mathrm{tr}\left[(\overline{\boldsymbol{g}}_k \overline{\boldsymbol{g}}_k^* + \boldsymbol{\Gamma}_k)\frac{\boldsymbol{\Sigma}_k}{p}\right] = \underbrace{\sum_{i=1}^{K}\Big(\sum_{m\in\mathcal{F}_k} m_{k,i,m}^{(i)} \overline{g}_{i,m} \overline{g}_{k,m}\Big)^2}_{\mathrm{T1}} + \underbrace{\sum_{m\in\mathcal{F}_k} (|\overline{g}_{k,m}|^2 + \gamma_{k,m})\Big(\sum_{i\in\mathcal{U}_m} c_{i,m}\Big)}_{\mathrm{T2}}$$
$$+ \underbrace{\sum_{m\in\mathcal{F}_k} (|\overline{g}_{k,m}|^2 + \gamma_{k,m})\Big(\sum_{i\notin\mathcal{U}_m} r_{i,m}\Big)}_{\mathrm{T3}} + \underbrace{\sum_{m\in\mathcal{F}_k} (|\overline{g}_{k,m}|^2 + \gamma_{k,m})\frac{\sigma^2}{p}}_{\mathrm{T4}}. \quad \text{(A.24)}$$

As for the interference terms, obtaining the derivative w.r.t. the FBS locations from (A.24) is tedious but straightforward. Combining the derivatives obtained from (A.23) and (A.24), the partial derivative w.r.t. x_m of $\mathrm{Den_m}$ arises. A similar procedure can be followed to obtain the derivative w.r.t. y_m and, with that, the overall gradient.

A.4 Proof of Prop. 3

In PC networks, only the FBSs have access to the channel estimates. Thus, the C-RAN regards $\boldsymbol{v}_k^* \mathbb{E}\{\boldsymbol{\mu}_{k,k}\}$ as the true channel and the signal model therein is

$$\hat{x}_k = \underbrace{\boldsymbol{v}_k^* \mathbb{E}\{\boldsymbol{\mu}_{k,k}\} x_k}_{\text{signal}} + \underbrace{(\boldsymbol{v}_k^* \boldsymbol{\mu}_{k,k} - \boldsymbol{v}_k^* \mathbb{E}\{\boldsymbol{\mu}_{k,k}\}) x_k}_{\text{channel uncertainty}} + \underbrace{\sum_{i \neq k} \boldsymbol{v}_k^* (\boldsymbol{m}_{k,i}^{(s)} \circ \boldsymbol{\mu}_{k,i}) x_i}_{\text{interference}} + \underbrace{\boldsymbol{v}_k^* \boldsymbol{n}_m'}_{\text{noise}}. \quad \text{(A.25)}$$

The second, third, and fourth terms, which are uncorrelated, are pooled as effective noise. As uncorrelated Gaussian noise represents the worst case in terms of the achievable spectral efficiency, we obtain the lower bound in Prop. 3.

A.5 Proof of Prop. 4

As for the centralized MRC, we compute the equivalents for each of the terms in (2.43). Again, we first divide both numerator and denominator by $1/|\mathcal{F}_k|^2$. Then, the signal term satisfies

$$\frac{1}{|\mathcal{F}_k|^2} |\boldsymbol{v}_k^* \mathbb{E}\{\boldsymbol{\mu}_{k,k}\}|^2 = \frac{1}{|\mathcal{F}_k|^2} |\mathbb{E}\{\hat{\boldsymbol{g}}_k^* \hat{\boldsymbol{g}}_k\}|^2 \quad \text{(A.26)}$$

$$= \frac{1}{|\mathcal{F}_k|^2} \left(\sum_{m \in \mathcal{F}_k} \gamma_{k,m} \right)^2. \quad \text{(A.27)}$$

For the interference terms, when $i \neq k$

$$\frac{1}{|\mathcal{F}_k|^2}\mathbb{E}\{|\boldsymbol{v}_k^*(\boldsymbol{m}_{k,i}^{(s)} \circ \boldsymbol{\mu}_{k,i})|^2\} =$$
$$\frac{1}{|\mathcal{F}_k|^2}\mathbb{E}\{|(\boldsymbol{m}_k^{(s)} \circ \hat{\boldsymbol{g}}_k^*)(\boldsymbol{m}_i^{(s)} \circ \boldsymbol{g}_i)|^2\} =$$
$$\frac{1}{|\mathcal{F}_k|^2}\mathrm{tr}\big[(\overline{\boldsymbol{g}}_k\overline{\boldsymbol{g}}_k^* + \boldsymbol{\Gamma}_k)((\boldsymbol{m}_{k,i}^{(s)} \circ \overline{\boldsymbol{g}}_i)(\boldsymbol{m}_{k,i}^{(s)} \circ \overline{\boldsymbol{g}}_i)^* + \boldsymbol{R}_i)\big], \quad \text{(A.28)}$$

where $\boldsymbol{R}_i = \mathrm{diag}\{m_{i,m}^{(s)} r_{i,m} \ \forall m\}$. For the variance term,

$$\frac{1}{|\mathcal{F}_k|^2}\mathrm{var}\{\boldsymbol{v}_k^*\boldsymbol{\mu}_{k,k}\} = \frac{1}{|\mathcal{F}_k|^2}\mathrm{var}\{\hat{\boldsymbol{g}}_k^*\boldsymbol{g}_k\}$$
$$= \frac{1}{|\mathcal{F}_k|^2}\mathrm{var}\{\hat{\boldsymbol{g}}_k^*\hat{\boldsymbol{g}}_k\} + \frac{1}{|\mathcal{F}_k|^2}\mathrm{var}\{\hat{\boldsymbol{g}}_k^*\tilde{\boldsymbol{g}}_k\}$$
$$= \mathbb{E}\left\{ \left| \frac{1}{|\mathcal{F}_k|}\hat{\boldsymbol{g}}_k^*\hat{\boldsymbol{g}}_k - \frac{1}{|\mathcal{F}_k|}(\overline{\boldsymbol{g}}_k^*\overline{\boldsymbol{g}}_k + \mathrm{tr}\boldsymbol{\Gamma}_k) \right|^2 \right\} + \mathbb{E}\left\{ \left| \frac{1}{|\mathcal{F}_k|}\hat{\boldsymbol{g}}_k^*\tilde{\boldsymbol{g}}_k \right|^2 \right\} \overset{\text{a.s.}}{\to} 0, \quad \text{(A.29)}$$

as a consequence of [82, Lemma 3]. A similar procedure can be used to derive the contribution of the noise,

$$\frac{1}{|\mathcal{F}_k|^2}\boldsymbol{v}_k^*\boldsymbol{\Sigma}_k^{'}\boldsymbol{v}_k = \frac{1}{|\mathcal{F}_k|}\mathbb{E}\left\{ \frac{1}{|\mathcal{F}_k|}(\hat{\boldsymbol{g}}_k^*\boldsymbol{\Sigma}^{''}\hat{\boldsymbol{g}}_k \right\} - \frac{1}{|\mathcal{F}_k|^2}\mathrm{tr}\big[(\overline{\boldsymbol{g}}_k\overline{\boldsymbol{g}}_k^* + \boldsymbol{\Gamma}_k)\boldsymbol{\Sigma}^{''}\big] \overset{\text{a.s.}}{\to} 0, \quad \text{(A.30)}$$

where $\boldsymbol{\Sigma}_k^{''} = (\boldsymbol{M}^{(i)}{}_k \circ \overline{\boldsymbol{G}}_k)(\boldsymbol{M}^{(i)} \circ \overline{\boldsymbol{G}}_k)^* p + \boldsymbol{D}_2 + \sigma^2\boldsymbol{I}$. After plugging the various terms into (2.43), Prop. 4 is obtained.

A.6 Proof of Theorem 1

Define the matrix

$$\boldsymbol{\Omega}_k = \left(\left(\boldsymbol{c}_k \circ \boldsymbol{M}^{(s)} \circ \hat{\boldsymbol{G}}_k\right) \boldsymbol{P} \left(\boldsymbol{c}_k \circ \boldsymbol{M}^{(s)} \circ \hat{\boldsymbol{G}}_k\right)^* - \left(\boldsymbol{c}_k \circ \hat{\boldsymbol{g}}_k\right)\left(\boldsymbol{c}_k \circ \hat{\boldsymbol{g}}_k\right)^* p_k + \boldsymbol{\Sigma}_k \right)^{-1}, \tag{A.31}$$

where $\boldsymbol{P} = \mathrm{diag}\{p_k \,\forall\, k\}$ and $\boldsymbol{\Omega}'_k = |\mathcal{F}_k|\boldsymbol{\Omega}_k$. Then, (3.26) can be written as

$$\begin{aligned} \mathrm{SINR}_k &= \left(\boldsymbol{c}_k \circ \hat{\boldsymbol{g}}_k\right)^* \boldsymbol{\Omega}_k \left(\boldsymbol{c}_k \circ \hat{\boldsymbol{g}}_k\right) p_k & \text{(A.32)} \\ &= \frac{p_k}{|\mathcal{F}_k|} \mathrm{tr}\left[\left(\boldsymbol{c}_k \circ \hat{\boldsymbol{g}}_k\right)\left(\boldsymbol{c}_k \circ \hat{\boldsymbol{g}}_k\right)^* \boldsymbol{\Omega}'_k\right]. & \text{(A.33)} \end{aligned}$$

For $|\mathcal{F}_k|,|\mathcal{U}_m| \to \infty \,\forall\, k, m$, using [82, Lemma 4] and Thm. 7,

$$\frac{p_k}{|\mathcal{F}_k|} \mathrm{tr}\left[\left(\boldsymbol{c}_k \circ \hat{\boldsymbol{g}}_k\right)\left(\boldsymbol{c}_k \circ \hat{\boldsymbol{g}}_k\right)^* \boldsymbol{\Omega}'_k\right] - \frac{p_k}{|\mathcal{F}_k|} \mathrm{tr}[\boldsymbol{\Phi}\boldsymbol{\Gamma}_k \boldsymbol{T}_k] \overset{\text{a.s.}}{\to} 0. \tag{A.34}$$

In our case, the role of $\left(\boldsymbol{H}\boldsymbol{H}^* + \boldsymbol{S} + z\boldsymbol{I}_M\right)^{-1}$ in Thm. 7 is played by $\boldsymbol{\Omega}'_k$. There is a direct mapping between the terms in the aforementioned theorem and our problem, namely (i) $\boldsymbol{D} = \boldsymbol{\Phi}\boldsymbol{\Gamma}_k\, p_k$, (ii) $\boldsymbol{R}_j = \boldsymbol{\Phi}\boldsymbol{\Gamma}_j\, p_j$, and (iii) $\boldsymbol{S} + z\boldsymbol{I}_M = \frac{1}{|\mathcal{F}_k|}\boldsymbol{\Sigma}_k$ with matrix $\boldsymbol{T}_k$ following the structure of $\boldsymbol{T}$ in Thm. 7, namely

$$\boldsymbol{T}_k = \left(\frac{1}{|\mathcal{F}_k|} \sum_{j \neq k}^{K} \frac{\boldsymbol{\Phi}\boldsymbol{\Gamma}_j}{1 + e_j}\, p_j + \frac{1}{|\mathcal{F}_k|}\boldsymbol{\Sigma}_k \right)^{-1}. \tag{A.35}$$

The necessary coefficients can be calculated as $e_j = \lim_{n\to\infty} e_j^{(n)}$ with

$$e_j^{(n)} = p_j \,\mathrm{tr}\left[\boldsymbol{\Phi}\boldsymbol{\Gamma}_j \left(\sum_{i \neq j}^{K} \frac{\boldsymbol{\Phi}\boldsymbol{\Gamma}_i}{1 + e_i^{(n-1)}}\, p_i + \boldsymbol{\Sigma}_j \right)^{-1} \right]. \tag{A.36}$$

$$\overline{\mathrm{SINR}}_k^{\mu} = \sum_{m\in\mathcal{F}_k} \frac{\gamma_{k,m}p_k}{\sum\limits_{\substack{i\in\mathcal{U}_m\\ i\neq k}} \frac{\gamma_{i,m}}{1+e_{i,k}}p_i + \frac{r_m}{\phi_m}\big(\sum\limits_{\forall i} r_{i,m}p_i + \frac{\sigma^2}{\mu}\big) - \sum\limits_{i\in\mathcal{U}_m}\gamma_{i,m}p_i + \frac{\sum\limits_{i=1}^{K} r_i p_i + \frac{\sigma^2}{\mu}}{\phi_m \mathrm{p}_m}\sigma^2}. \quad \text{(A.39)}$$

The fixed-point algorithm can be used to compute $e_j^{(n)}$ and has been proved to converge [82]. Finally, since matrices $\boldsymbol{\Gamma}_k$ and $\boldsymbol{T}_k$ are diagonal, (A.34) can be written as

$$\frac{p_k}{|\mathcal{F}_k|}\mathrm{tr}[\boldsymbol{\Phi}\boldsymbol{\Gamma}_k\boldsymbol{T}_k] = p_k\,\mathrm{tr}\left[\boldsymbol{\Phi}\boldsymbol{\Gamma}_k\left(\sum_{i\neq k}^{K}\frac{\boldsymbol{\Phi}\boldsymbol{\Gamma}_i}{1+e_i}\,p_i + \boldsymbol{\Sigma}_k\right)^{-1}\right], \quad \text{(A.37)}$$

and, with some straightforward algebra, the expression in Thm. 1 is obtained.

A.7 Proof of Prop. 5

The definition of competitive utility functions and monotonic constraints are available at [104, Assumptions 1 and 2]. In our case, the utility function of user k is given in Thm. 1. It satisfies positivity because each $\overline{\mathrm{SINR}}_{k,m}$ in (3.31) is positive. Then, to verify competitiveness, it is enough to show that a function of the type $\sum\limits_{m\in\mathcal{F}_k}\frac{a_{k,m}p_k}{c_{k,m}+d_{k,m}p_k}$ is always increasing for $a_{k,m}, c_{k,m}, d_{k,m} > 0$. Indeed,

$$\frac{d}{d\,p_k}\sum_{m\in\mathcal{F}_k}\frac{a_{k,m}p_k}{c_{k,m}+d_{k,m}p_k} = \sum_{m\in\mathcal{F}_k}\frac{a_{k,m}c_{k,m}}{(c_{k,m}+d_{k,m}p_k)^2} > 0. \quad \text{(A.38)}$$

Similarly, it can be shown that $\overline{\mathrm{SINR}}_k$ is decreasing with respect to p_i for $i \neq k$. Finally, to show directional monotonicity, we substitute p_i by $\mu p_i\ \forall\, i$ and define the new SINR by $\overline{\mathrm{SINR}}_k^{\mu}$. After some straightforward algebraic manipulations, $\overline{\mathrm{SINR}}_k^{\mu}$ is included in (A.39). All the terms in $\overline{\mathrm{SINR}}_k^{\mu}$ are as in $\overline{\mathrm{SINR}}_k$ except for the ones that depend on σ^2, which are divided by μ. Provided that $\mu > 1$, each of the denominators is smaller in $\overline{\mathrm{SINR}}_k^{\mu}$ and

thus $\overline{\mathrm{SINR}}_k^{\mu} \geq \overline{\mathrm{SINR}}_k$ for $\mu > 1$. Finally, it is easy to show that $p_k \leq p_{\max}$ are monotonic constraints.

A.8 Proof of Theorem 2

Given a ZF fronthaul combiner $\boldsymbol{U}$, we can make construct $\boldsymbol{U}^{(\epsilon)} = \hat{\boldsymbol{H}}(\hat{\boldsymbol{H}}^* \hat{\boldsymbol{H}} + \epsilon \boldsymbol{I})^{-1}$ satisfying $\boldsymbol{U} = \lim_{\epsilon \to 0} \boldsymbol{U}^{(\epsilon)}$. For ease of exposition, we define $\boldsymbol{\Omega} = (\frac{1}{N}\hat{\boldsymbol{H}}\hat{\boldsymbol{H}}^* + \frac{\epsilon}{N}\boldsymbol{I})^{-1}$ while $\boldsymbol{\Omega}_m$ equals $\boldsymbol{\Omega}$ without the contribution of the mth channel. Note that $\boldsymbol{U}^{(\epsilon)} = \frac{1}{N}\hat{\boldsymbol{H}}\boldsymbol{\Omega}$. In addition,

$$\boldsymbol{U}^{(\epsilon)} = \hat{\boldsymbol{H}}(\hat{\boldsymbol{H}}^* \hat{\boldsymbol{H}} + \epsilon \boldsymbol{I})^{-1} = (\hat{\boldsymbol{H}}\hat{\boldsymbol{H}}^* + \epsilon \boldsymbol{I})^{-1}\hat{\boldsymbol{H}} = \frac{1}{N}\boldsymbol{\Omega}\hat{\boldsymbol{H}}. \tag{A.40}$$

As a consequence, $\mathbb{E}\{\boldsymbol{u}_m^{(\epsilon)*}\boldsymbol{Q}\boldsymbol{u}_m^{(\epsilon)}\}$ can be written as

$$\mathbb{E}\{\boldsymbol{u}_m^{(\epsilon)*}\boldsymbol{Q}\boldsymbol{u}_m^{(\epsilon)}\} = \mathbb{E}\left\{\frac{1}{N^2}\hat{\boldsymbol{h}}_m^* \boldsymbol{\Omega}\boldsymbol{Q}\boldsymbol{\Omega}\hat{\boldsymbol{h}}_m\right\}. \tag{A.41}$$

The term inside the expectation satisfies

$$\frac{1}{N^2}\hat{\boldsymbol{h}}_m^* \boldsymbol{\Omega}\boldsymbol{Q}\boldsymbol{\Omega}\hat{\boldsymbol{h}}_m = \frac{\frac{1}{N^2}\hat{\boldsymbol{h}}_m^* \boldsymbol{\Omega}_m\boldsymbol{Q}\boldsymbol{\Omega}_m\hat{\boldsymbol{h}}_m}{(1 + \frac{1}{N}\hat{\boldsymbol{h}}_m^* \boldsymbol{\Omega}_m\hat{\boldsymbol{h}}_m)^2} \tag{A.42}$$

$$= \frac{\mathrm{N}_m}{(1 + \mathrm{D}_m)^2}. \tag{A.43}$$

Note that N_m converges a.s. to

$$\mathrm{N}_m \overset{\text{a.s.}}{\to} \frac{1}{N^2}\mathrm{tr}\left(\boldsymbol{\Phi}_m\boldsymbol{\Omega}_m\boldsymbol{Q}\boldsymbol{\Omega}_m\right) \tag{A.44}$$

$$\overset{\text{a.s.}}{\to} \frac{1}{N^2}\mathrm{tr}\left(\boldsymbol{\Phi}_m\boldsymbol{T}'(\epsilon, \boldsymbol{Q})\right), \tag{A.45}$$

where $\boldsymbol{T}'(\epsilon, \boldsymbol{Q})$ is provided in Thm. 8 for $\boldsymbol{D} = \boldsymbol{\Phi}_m$, $\boldsymbol{\Phi} = \boldsymbol{Q}$, $\boldsymbol{S} = \boldsymbol{0}$, $z = \epsilon$, $\boldsymbol{R}_k = \boldsymbol{\Phi}_k$ and by substituting $M = N$. For the term in the denominator, applying Thm. 7,

$$\mathrm{D}_m \xrightarrow{\text{a.s.}} \frac{1}{N}\mathrm{tr}\left(\boldsymbol{\Phi}_m \boldsymbol{\Omega}_m\right) \tag{A.46}$$

$$\xrightarrow{\text{a.s.}} \frac{1}{N}\mathrm{tr}\left(\boldsymbol{\Phi}_m \boldsymbol{T}\right), \tag{A.47}$$

where the same substitutions used to obtain $\boldsymbol{T}'(\epsilon, \boldsymbol{Q})$ are made in Thm. 7 to acquire $\boldsymbol{T}$. Applying the continuous mapping theorem,

$$\mathbb{E}\{\boldsymbol{u}_m^{(\epsilon)*}\boldsymbol{Q}\boldsymbol{u}_m^{(\epsilon)}\} - \frac{\frac{1}{N^2}\mathrm{tr}\left(\boldsymbol{\Phi}_m \boldsymbol{T}'(\epsilon, \boldsymbol{Q})\right)}{\left(1 + \frac{1}{N}\mathrm{tr}\left(\boldsymbol{\Phi}_m \boldsymbol{T}\right)\right)^2} \xrightarrow{\text{a.s.}} 0. \tag{A.48}$$

Taking the limit when $\epsilon \to 0$ in both terms results in the convergence stated in Thm. 2.

A.9 Proof of Prop. 6

Given that we have added three slack variables, we analyze their behavior individually. First, $||\boldsymbol{\ell}(n+1) - \boldsymbol{\ell}(n)|| \leq \Delta(n)$ will be met with equality at the optimal, otherwise the value of $\Delta(n)$ can be reduced to achieve a better cost function. A similar argument is used to show that $y(n)^2 = \sqrt{\delta(n)^4 + \frac{\Delta(n)^4}{4v_0^4}} - \frac{\Delta(n)^2}{2v_0^2}$, otherwise the value of $y(n)$ can be decreased to reduce the cost function. Finally, at the optimal, Eq. (5.28) is met, or else the value of $\beta(n)$ can be increased.

A.10 Proof of Prop. 7

We first define $x = 1 + e^{-(B_1 + B_2 u_B(n))}$ and $y = d_B^2(n)$. Also, let us define $C_3 = C_1 \frac{\beta_0 p(n)}{\sigma^2}$ and $C_4 = C_2 \frac{\beta_0 p(n)}{\sigma^2}$, both strictly positive for $p(n) > 0$. Substituting the previous terms

$$\nabla^2 f(x,y) = \begin{pmatrix} \frac{\partial^2 f(x,y)}{\partial x^2} & \frac{\partial^2 f(x,y)}{\partial x \partial y} \\ \frac{\partial^2 f(x,y)}{\partial x \partial y} & \frac{\partial^2 f(x,y)}{\partial y^2} \end{pmatrix} = \begin{pmatrix} \frac{C_4(2xy^{\frac{\kappa}{2}}+2C_3x+C_4)}{x^2(xy^{\frac{\kappa}{2}}+C_3x+C_4)^2} & \frac{\frac{\kappa}{2}C_4y^{\frac{\kappa}{2}-1}}{xy^{\frac{\kappa}{2}}+C_3x+C_4)^2} \\ \frac{\frac{\kappa}{2}C_4y^{\frac{\kappa}{2}-1}}{xy^{\frac{\kappa}{2}}+C_3x+C_4)^2} & \frac{\frac{\kappa}{2}(C_3x+C_4)\Big((1+\frac{\kappa}{2})xy^{\frac{\kappa}{2}}+C_3x+C_4\Big)}{y^2(xy^{\frac{\kappa}{2}}+C_3x+C_4)^2} \end{pmatrix}. \tag{A.49}$$

$$\boldsymbol{s}^{\mathrm{T}}\nabla^2 f(x,y)\boldsymbol{s} = \frac{C_4 s_1^2 y^2(xy^{\frac{\kappa}{2}}+2C_3x+C_4) + s_2^2x^2\big((2C_3x+C_4)xy^{\frac{\kappa}{2}}+(C_3x+C_4)^2\big) + C_4xy^{\frac{\kappa}{2}}(s_1y+s_2x)^2}{x^2y^2(xy^{\frac{\kappa}{2}}+C_3x+C_4)^2}.$$

into $R(n)$ and dropping the time index n, we obtain $R = f(x.y)\log_2(e)$ where $f(x,y) = \log\left(1+\big(C_3+\frac{C_4}{x}\big)\frac{1}{y^{\frac{\kappa}{2}}}\right)$. The Hessian matrix of $f(x,y)$ is included in Eq. (A.49). For any vector $\boldsymbol{s} = (s_1, s_2)^{\mathrm{T}}$, it can be shown that $\boldsymbol{s}^{\mathrm{T}}\nabla^2 f(x,y)\boldsymbol{s} \geq 0$ as included in Eq. (A.50). Therefore, $f(x,y)$ is convex with respect to x, y, which ultimately implies the convexity of $R(n)$ as well.

A.11 Proof of Lemma 1

By exploiting the convexity of $R(n)$ as shown in Prop. 7, such a term accepts a lower bound of the type:

$$R(n) \geq R^{lb}(n) \tag{A.50}$$

$$= \overline{R}(n) - \phi(n)\big(e^{-\lambda(n)} - e^{-\overline{\lambda}(n)}\big) -$$

$$\zeta(n)\big(||\boldsymbol{q}(n) - \boldsymbol{q}_B||^2 - ||\overline{\boldsymbol{q}}(n) - \boldsymbol{q}_B||^2\big), \tag{A.51}$$

where coefficients $\phi(n)$ and $\zeta(n)$ are provided in Eqs. (A.52) and (A.54), respectively.

$$\phi(n) = \frac{\partial R(n)}{\partial(1+e^{-\lambda(n)})} \tag{A.52}$$

$$= \frac{\frac{\beta_0}{\sigma^2}p(n)C_2}{\log(2)(1+e^{-\overline{\lambda}(n)})\big(\frac{\beta_0}{\sigma^2}p(n)[C_1(1+e^{-\overline{\lambda}(n)})+C_2]+(1+e^{-\overline{\lambda}(n)})d_B^{\kappa}(n)\big)} \tag{A.53}$$

$$\zeta(n) = \frac{\partial R(n)}{\partial d_B^2(n)} = \frac{\frac{\beta_0}{\sigma^2}p(n)\kappa[C_1(1+e^{-\overline{\lambda}(n)})+C_2]}{2\log(2)d_B^2(n)\big(\frac{\beta_0}{\sigma^2}p(n)[C_1(1+e^{-\overline{\lambda}(n)})+C_2]+(1+e^{-\overline{\lambda}(n)})d_B^{\kappa}(n)\big)} \tag{A.54}$$

A.12 Proof of Lemma 2

A similar procedure as the one followed in App. A.10 is considered to show the convexity of $u_B(n) = \frac{(H(n)-H_B)}{d_B(n)}$ with respect to $||\boldsymbol{q}(n)-\boldsymbol{q}_B||^2$. As a consequence, $u_B(n)$ accepts the following lower bound:

$$u_B(n) \geq u_B^{lb}(n) \tag{A.55}$$

$$= \overline{u}_B(n) - \psi(n)\big(||\boldsymbol{q}(n)-\boldsymbol{q}_B||^2 - ||\overline{\boldsymbol{q}}(n)-\boldsymbol{q}_B||^2\big), \tag{A.56}$$

where the value of $\psi(n)$ is

$$\psi(n) = \frac{\partial u_B(n)}{\partial||\boldsymbol{q}(n)-\boldsymbol{q}_B||^2} \tag{A.57}$$

$$= \frac{H(n)-H_B}{2(||\overline{\boldsymbol{q}}(n)-\boldsymbol{q}_B||^2+(H(n)-H_B)^2)^{\frac{3}{2}}} \tag{A.58}$$

A.13 Proof of Prop. 8

To solve the power allocation problem, we first formulate the Lagrangian as in (A.59). Defining $w_m(n) = \sum_{i=1}^{n-D}\lambda_{i,m}\alpha_{s_m}(i)$, the Lagrangian can be rewritten as presented in (A.60)

$$\mathcal{L}(p(n),\boldsymbol{\lambda}) = \sum_{n=1}^{N}\delta(n)p(n) \; + \sum_{n=1}^{N-D}\sum_{m=1}^{M}\lambda_{n,m}\Big[\alpha_{s_m}(n)\sum_{i=n+D}^{N}\delta(i)R(i) + B\big(1-\alpha_m(n)\big) - \sum_{l=m}^{M}C_{s_l}\Big] + \sum_{n=D+1}^{N}\sum_{k=1}^{K}\lambda_{n,k}^{'}\big(I_{th} - p(n)K_k(n)\big) + \sum_{n=D+1}^{N}\lambda_{n}^{''}\big(p_{max} - p(n)\big). \tag{A.59}$$

$$\mathcal{L}(p(n),\boldsymbol{\lambda}) = \sum_{n=1}^{N}\delta(n)p(n) \; + \sum_{n=D+1}^{N}\sum_{m=1}^{M}w_m(n)\delta(n)R(n) - \sum_{n=D+1}^{N}\sum_{k=1}^{K}\lambda_{n,k}^{'}p(n)K_k(n) - \sum_{n=D+1}^{N}\lambda_{n}^{''}p(n) + K_{ct}, \tag{A.60}$$

where K_{ct} is a constant term that does not depend on $p(n)$. Taking the derivative with respect to the optimization variables, $p(n)$, for fixed values of the multipliers, we obtain

$$\frac{d\mathcal{L}(p(n),\boldsymbol{\lambda})}{dp(n)} = \delta(n) + \sum_{m=1}^{M}\frac{w_m(n)\delta(n)}{\ln(2)}\frac{K_B(n)}{1+p(n)K_B(n)} - \sum_{k=1}^{K}\lambda_{n,k}^{'}K_k(n) - \lambda_{n,m}^{''} = 0. \tag{A.61}$$

Solving the previous equation for $p(n)$, we obtain:

$$p(n) = \left[\frac{\sum_{m=1}^{M}\frac{w_m(n)\delta(n)}{\ln(2)}}{\sum_{k=1}^{K}\lambda_{n,k}^{'}K_k(n) + \lambda_{n}^{''} - \delta(n)} - \frac{1}{K_B(n)}\right]^{+}, \tag{A.62}$$

for all $n = D+1,\ldots,N$. The value of $a(n)$ in Eq. (5.44) is therefore:

$$a(n) = \frac{\sum_{m=1}^{M}\frac{w_m(n)\delta(n)}{\ln(2)}}{\sum_{k=1}^{K}\lambda_{n,k}^{'}K_k(n) + \lambda_{n}^{''} - \delta(n)}. \tag{A.63}$$

To obtain the Lagrangian multipliers, we aim to solve the dual problem after obtaining the solution to $p(n)$:

$$\begin{aligned} \max_{\boldsymbol{\lambda}} \quad & \mathcal{L}(p(n), \boldsymbol{\lambda}) \\ \text{s.t.} \quad & \boldsymbol{\lambda} \geq \mathbf{0}, \end{aligned} \tag{A.64}$$

which can be efficiently solved by gradient methods and therefore obtain the optimal multipliers $\boldsymbol{\lambda}^*$ for fixed $p^*(n)$. Particularly, the ellipsoid method is an enticing solution [106] where obtaining the subgradient of $\mathcal{L}(p^*(n), \boldsymbol{\lambda})$ with respect to $\boldsymbol{\lambda}$ for fixed $p^*(n)$ is a straightforward calculation given (A.60). Consequently, we use an iterative process between the solutions obtained in (A.62) and (A.64).

A.14 Proof of Prop. 9

Define the iteration number and the corresponding cost function by j and $\eta^{(j)}$, respectively. Next, we get rid of the time index n and represent the cost as a function of the optimization variables, i.e., $\eta\big(\alpha_m^{(j)}, \boldsymbol{q}^{(j)}, \delta^{(j)}, H^{(j)}, p^{(j)}\big)$. The BCD approach followed in Alg. 4 provides the following inequalities: (i) $\eta\big(\alpha_m^{(j)}, \boldsymbol{q}^{(j)}, \delta^{(j)}, H^{(j)}, p^{(j)}\big) \geq \eta\big(\alpha_m^{(j+1)}, \boldsymbol{q}^{(j)}, \delta^{(j)}, H^{(j)}, p^{(j)}\big)$ by solving (5.23); (ii) applying the SCA technique to the 2D-UAV path and UAV altitude subproblems presented in (5.30) and (5.42), respectively, results in $\eta\big(\alpha_m^{(j+1)}, \boldsymbol{q}^{(j)}, \delta^{(j)}, H^{(j)}, p^{(j)}\big) \geq \eta\big(\alpha_m^{(j+1)}, \boldsymbol{q}^{(j+1)}, \delta^{(j+1)}, H^{(j+1)}, p^{(j)}\big)$, and (iii) optimizing the transmit power in (5.43) results in $\eta\big(\alpha_m^{(j+1)}, \boldsymbol{q}^{(j+1)}, \delta^{(j+1)}, H^{(j+1)}, p^{(j)}\big) \geq \eta\big(\alpha_m^{(j+1)}, \boldsymbol{q}^{(j+1)}, \delta^{(j+1)}, H^{(j+1)}, p^{(j+1)}\big)$. As a result, Alg. 4 provides a non-increasing sequence: $\eta^{(0)} \geq \eta^{(1)} \geq \cdots \geq \eta^{(*)}$ where for simplicity $\eta^{(*)}$ is the objective function after convergence. Since the cost function is lower-bounded by a value of zero, the BCD approach followed by Alg. 4 will converge.