



Politecnico di Bari

Repository Istituzionale dei Prodotti della Ricerca del Politecnico di Bari

Advanced modeling and design techniques for the internet of drones

This is a PhD Thesis

Original Citation:

Advanced modeling and design techniques for the internet of drones / Iacovelli, Giovanni. - ELETTRONICO. - (2022).
[10.60576/poliba/iris/iacovelli-giovanni_phd2022]

Availability:

This version is available at <http://hdl.handle.net/11589/245821> since: 2022-12-14

Published version

DOI:10.60576/poliba/iris/iacovelli-giovanni_phd2022

Publisher: Politecnico di Bari

Terms of use:

(Article begins on next page)



Politecnico
di Bari

Department of Electrical and Information Engineering
Electrical and Information Engineering

Ph.D. Program

SSD: ING-INF/03– Telecommunications

Final Dissertation

Advanced Modeling and Design Techniques for the Internet of Drones

by

Giovanni Iacovelli

Supervisors:

Prof. Luigi Alfredo Grieco

Coordinator of Ph.D. Program:

Prof. Mario Carpentieri

Course n°35, 01/11/2019-31/10/2022



LIBERATORIA PER L'ARCHIVIAZIONE DELLA TESI DI DOTTORATO

Al Magnifico Rettore
del Politecnico di Bari

Il sottoscritto Giovanni Iacovelli nato a Bari (BA) il 05/10/1995 residente a Bitetto (BA) in via Sante Cirrotti 29 e-mail giacovelli2k@gmail.com iscritto al 3° anno di Corso di Dottorato di Ricerca in Electrical and Information Engineering ciclo XXXV ed essendo stato ammesso a sostenere l'esame finale con la prevista discussione della tesi dal titolo:

Advanced Modeling and Design Techniques for the Internet of Drones

DICHIARA

- 1) di essere consapevole che, ai sensi del D.P.R. n. 445 del 28.12.2000, le dichiarazioni mendaci, la falsità negli atti e l'uso di atti falsi sono puniti ai sensi del codice penale e delle Leggi speciali in materia, e che nel caso ricorressero dette ipotesi, decade fin dall'inizio e senza necessità di nessuna formalità dai benefici conseguenti al provvedimento emanato sulla base di tali dichiarazioni;
- 2) di essere iscritto al Corso di Dottorato di ricerca Electrical and Information Engineering ciclo XXXV, corso attivato ai sensi del "Regolamento dei Corsi di Dottorato di ricerca del Politecnico di Bari", emanato con D.R. n.286 del 01.07.2013;
- 3) di essere pienamente a conoscenza delle disposizioni contenute nel predetto Regolamento in merito alla procedura di deposito, pubblicazione e autoarchiviazione della tesi di dottorato nell'Archivio Istituzionale ad accesso aperto alla letteratura scientifica;
- 4) di essere consapevole che attraverso l'autoarchiviazione delle tesi nell'Archivio Istituzionale ad accesso aperto alla letteratura scientifica del Politecnico di Bari (IRIS-POLIBA), l'Ateneo archiverà e renderà consultabile in rete (nel rispetto della Policy di Ateneo di cui al D.R. 642 del 13.11.2015) il testo completo della tesi di dottorato, fatta salva la possibilità di sottoscrizione di apposite licenze per le relative condizioni di utilizzo (di cui al sito <http://www.creativecommons.it/Licenze>), e fatte salve, altresì, le eventuali esigenze di "embargo", legate a strette considerazioni sulla tutelabilità e sfruttamento industriale/commerciale dei contenuti della tesi, da rappresentarsi mediante compilazione e sottoscrizione del modulo in calce (Richiesta di embargo);
- 5) che la tesi da depositare in IRIS-POLIBA, in formato digitale (PDF/A) sarà del tutto identica a quelle **consegnate**/inviata/da inviarsi ai componenti della commissione per l'esame finale e a qualsiasi altra copia depositata presso gli Uffici del Politecnico di Bari in forma cartacea o digitale, ovvero a quella da discutere in sede di esame finale, a quella da depositare, a cura dell'Ateneo, presso le Biblioteche Nazionali Centrali di Roma e Firenze e presso tutti gli Uffici competenti per legge al momento del deposito stesso, e che di conseguenza va esclusa qualsiasi responsabilità del Politecnico di Bari per quanto riguarda eventuali errori, imprecisioni o omissioni nei contenuti della tesi;
- 6) che il contenuto e l'organizzazione della tesi è opera originale realizzata dal sottoscritto e non compromette in alcun modo i diritti di terzi, ivi compresi quelli relativi alla sicurezza dei dati personali; che pertanto il Politecnico di Bari ed i suoi funzionari sono in ogni caso esenti da responsabilità di qualsivoglia natura: civile, amministrativa e penale e saranno dal sottoscritto tenuti indenni da qualsiasi richiesta o rivendicazione da parte di terzi;
- 7) che il contenuto della tesi non infrange in alcun modo il diritto d'Autore né gli obblighi connessi alla salvaguardia di diritti morali od economici di altri autori o di altri aventi diritto, sia per testi, immagini, foto, tabelle, o altre parti di cui la tesi è composta.

Bari, 23/08/2022

Firma _____

Il/La sottoscritto, con l'autoarchiviazione della propria tesi di dottorato nell'Archivio Istituzionale ad accesso aperto del Politecnico di Bari (POLIBA-IRIS), pur mantenendo su di essa tutti i diritti d'autore, morali ed economici, ai sensi della normativa vigente (Legge 633/1941 e ss.mm.ii.),

CONCEDE

- al Politecnico di Bari il permesso di trasferire l'opera su qualsiasi supporto e di convertirla in qualsiasi formato al fine di una corretta conservazione nel tempo. Il Politecnico di Bari garantisce che non verrà effettuata alcuna modifica al contenuto e alla struttura dell'opera.
- al Politecnico di Bari la possibilità di riprodurre l'opera in più di una copia per fini di sicurezza, back-up e conservazione.

Bari, 23/08/2022

Firma _____



Politecnico
di Bari

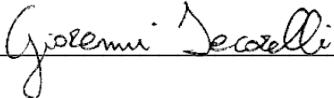
Department of Electrical and Information Engineering
Electrical and Information Engineering
Ph.D. Program
SSD: ING-INF/03– Telecommunications

Final Dissertation

Advanced Modeling and Design Techniques for the Internet of Drones

by

Giovanni Iacovelli



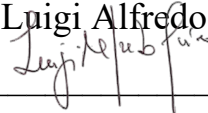
Referees:

Prof. Claudia Campolo

Dott.ssa Anna Maria Vegni

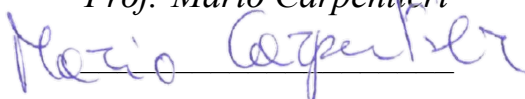
Supervisors:

Prof. Luigi Alfredo Grieco



Coordinator of Ph.D Program:

Prof. Mario Carpentieri



«Everyone says “Well, you should be harmless, virtuous, you shouldn’t do anyone any harm, you should sheath your competitive instinct, you shouldn’t try to be too assertive, you want to take a back seat”. NO. Wrong. You should be a monster. An absolute monster, and then you should learn how to control it.»

«Because if you are not capable of cruelty, you are absolutely a victim to anyone who is. And so, part of the reason that people go watch anti-heroes and villains is because there’s a part of them crying out for the incorporation of the monster within them, which is what gives them strength of character and self-respect. Because it’s impossible to respect yourself until you grow teeth, and if you grow teeth then you realize that you’re somewhat dangerous, [...] and then you might be more willing to demand that you treat yourself with respect and other people do the same thing. And so, that doesn’t mean that being cruel is better than not being cruel. What it means is that being able to be cruel, and then not being cruel, is better than not being able to be cruel. Because in the first case you’re not but weak and naïve, and in the second case you’re dangerous but you have it under control. A lot of martial arts concentrate on exactly that as part of their philosophy of training. It’s like “We’re not training you to fight, we’re training you to be peaceful, and awake, and avoid fights”. [...]

The strength that you develop in your monstrosity is actually the best guarantee of peace and that’s partly why Jung believed that it was necessary for people to integrate their shadow. And he said that was a terrible thing for people to attempt because the human shadow, which is all those things about yourself that you don’t want to realize, reaches all the way to hell. And what he meant by that was it’s through an analysis of your own shadow that you can come to understand why other people are capable, and you as well, of the sorts of terrible atrocities that characterize, let’s say, the 20th century. And without that understanding there’s no possibility of bringing it under control. When you study nazi Germany, for example, or you study the Soviet Union, particularly under Stalin, and you’re asking yourself “Well, what are these perpetrators like?”, [...] the answer is “They’re just like you”. If you don’t know that, that just means that you don’t know anything about people, including yourself. It also means that you have to discover why they’re just like you, and believe me that’s no picnic.

So that’s enough to traumatize people and that’s partly why they don’t do it. And it’s also partly why the path to enlightenment and wisdom is seldom trod upon, because if it was all a matter of following your bliss and doing what made you happy then everyone in the world would be a paragon of wisdom. But it’s not that at all, it’s a matter of facing the thing you least want to face.»

Jordan B. Peterson

«Being aggressive. That means you are ready to attack. As I always point out, this doesn’t mean you walk around with your chest puffed out, ready to bang heads with everyone around you. It doesn’t mean you confront people physically or mentally head-on without a tactically superior plan. NO. What it does mean is that you are going to get after it. You are going to move fast. You are going to think fast. You are going to outthink and outmaneuver the enemy. If I think the enemy is going to attack me, I’m going to attack them first. If I think they are going to seize a piece of terrain, I am going to be there waiting for them. If I think the enemy is going to flank me, too late I’m already flanking them. I view aggression as a fire in your mind that says “I am going to win”. I am going to battle. And I am going to fight. And I am going to use every tool I have to crush my enemy. And that is what aggression is to me. The unstoppable fighting spirit. The drive. The burning desire to achieve mission success using every possible tool, asset, and strategy, and tactic to bring about victory. It is the will to win. And if that kind of internal relentless aggression is your default mode, you will win.»

John Gretton "Jocko" Willink

POLITECNICO DI BARI

Abstract

Department of Electrical and Information Engineering

Doctor of Philosophy

Advanced Modeling and Design Techniques for the Internet of Drones

by Giovanni IACOVELLI

Internet of Drones (IoD) is a major leap in telecommunications. In fact, due to their inherently versatility, Unmanned Aerial Vehicles (UAVs) introduce several degrees of freedom in the reference system, which can be employed to enhance the communication quality. Therefore, drones are going to be a cornerstone of the future wireless technologies. Inevitably, this comes with new challenges, which require sophisticated and well-thought-out strategies, to be overtaken. Starting from an architectural standpoint, this work proposes and analyzes mathematical models which are then employed to derive optimized solutions. Moreover, to further corroborate the theoretical findings and to speed up the prototyping phase of an IoD system, a thorough simulation platform is presented.

Contents

List of Figures	viii
List of Tables	ix
List of Acronyms	xi
Scientific Contributions	xv
Introduction to the Internet of Drones	1
1 On the Design of the Drone Control Layer	3
1.1 Introduction	3
1.2 Reference Scenario	4
1.3 Interfaces Characterization	5
1.4 Core Characterization	5
2 On the Interplay between Energy and Memory Constraints in Optimized UAV Communications	7
2.1 Introduction	7
2.2 Problem Formulation	8
2.3 Proposed solution	10
2.4 Performance evaluation	11
2.4.1 Parameter settings	11
2.4.2 Discussion on results	12
3 An Optimization Approach to Energy-Efficient UAV Communications in Cellular Networks	15
3.1 Introduction	15
3.2 Reference Scenario	16
3.3 System Model	16
3.3.1 UAV transmission	17
3.3.2 Trajectories in the area of interest	18
3.3.3 Storage and Data model	19
3.3.4 UAV power consumption model	19
3.4 Problem Formulation	21
3.5 Proposed Solution	22
3.5.1 Sub-Problem 1: Gathered/Transmitted Data Optimization	22
3.5.2 Sub-Problem 2: Trajectory-related Parameters Optimization	22
3.5.3 Sub-Problem 3: Base Stations Optimization	24
3.5.4 Overall iterative algorithm	25
3.6 Numerical results	25
3.6.1 Parameter settings	25
3.6.2 Optimization Framework	26

4	Drone Swarm as Mobile Relaying System: A Hybrid Optimization Approach	29
4.1	Introduction	29
4.2	System Model	30
4.3	Problem Formulation	32
4.4	Proposed Solution	32
4.4.1	Sub-Problem 1: Trajectory optimization	32
4.4.2	Sub-Problem 2: Scheduling Optimization	33
4.5	Numerical results	35
5	Channel Gain Lower Bound for IRS-Assisted UAV-Aided Communications	39
5.1	Introduction	39
5.2	System Model	40
5.3	Channel Modeling Approximation	43
5.4	Numerical Results and Discussion	46
6	Multi-UAV IRS-assisted Communications: Multi-User Channel Modeling and Fair Sum-Rate Optimization via Deep Reinforcement Learning	49
6.1	Introduction	49
6.2	System Model	52
6.3	Proposed Channel Modeling	53
6.4	Trajectory and Phase Shift Matrix Optimization	60
6.4.1	Problem formulation	60
6.4.2	Proposed Solution based on Reinforcement Learning	61
6.5	Numerical Results and Discussion	62
6.5.1	Channel Model Analysis	63
6.5.2	Results on trajectory and phase shift optimization	64
7	Internet of Drones Simulator: Design, Implementation, and Performance Evaluation	67
7.1	Introduction	67
7.2	Architectural Overview	68
7.3	Underlying Platform	69
7.4	Core of IoD-Sim	69
7.4.1	World Definition	69
7.4.2	Drones	70
7.4.3	Other Simulation Entities: ZSPs and Remotes	71
7.4.4	Mobility	72
7.4.5	Applications	73
7.4.6	Scenario Configuration Interface	74
7.5	Simulation Development Platform	76
7.6	Simulation Campaign	78
7.6.1	Scenario Design	79
7.6.2	Scenario #1 - Telemetry	81
7.6.3	Scenario #2 - Multimedia Signals Acquisition	83
7.6.4	Scenario #3 - Smart Cities	84
7.6.5	Performance Evaluations	86
	Conclusions and Future Works	89
	Bibliography	91

List of Figures

1.1	Schematic representation of the envisioned scenario.	4
1.2	High level overview of DCL Interfaces and their types of applicability.	5
1.3	High level architecture of the DCL.	6
2.1	Drone acquiring multimedia signal while offloading to the BS.	8
2.2	Power consumption in the energy-bounded configuration.	12
2.3	Datarates in all the configurations.	12
2.4	Memory availability trend over the mission.	13
3.1	Reference Scenario.	16
3.2	Trajectories and Base Station (BS) association for different K-factors.	26
3.3	Speed profiles over time.	26
3.4	Data rate comparison over time.	27
3.5	Transmission power comparison over time.	27
3.6	Convergence of the overall algorithm.	27
4.1	Trajectories followed by drones and association with Sensor Nodes (SNs) in the first scenario.	35
4.2	Data rate in the first scenario.	35
4.3	Trajectories and total relayed data in the second scenario.	36
4.4	Convergence in the first two scenarios.	36
4.5	Trajectories and total relayed data in the third scenario.	37
4.6	Average data rate of swarms and convergence in the third scenario.	37
4.7	Trajectory plans comparison in the fourth scenario.	38
5.1	Reference scenario.	40
5.2	Comparison between approximation and actual Complementary CDF for different values of δ_X and δ_Y	46
5.3	$F_{ G_{k,i} ^2}^{-1}$ for different K-factors and number of PRUs.	47
5.4	Relative errors between exact and approximate curves with K-factors, outage probabilities, and number of PRUs.	47
6.1	Reference scenario.	52
6.2	(A)pproximation vs Monte Carlo (MC).	63
6.3	$\overline{\text{SNR}}$ with respect to number of PRUs.	63
6.4	$\overline{\text{SNR}}$ for $P_u = 2$ patches divided horizontally (left) and vertically (right).	64
6.5	Drones' trajectories and normalized Signal-to-Noise Ratio (SNR) for $G = 20$, $U = 3$, $P_u = 4$, and $\varpi = 5 \cdot 10^{-4}$	65
6.6	Total amount of data received by nodes, with $P_u = 4$	65
6.7	Total amount of data received by nodes, with $P_u = 9$	66
6.8	Comparison among data transmitted amounts for random (Rnd) and optimized (Opt) scheduling. The black stars represent the mean value.	66
7.1	Overview of Internet of Drones Simulator (IoD-Sim) Architecture.	68

7.2	A set of trajectories, generated with (7.6), with different Interest Levels (from 1 to 10, incrementally) for Points of Interest (PpoI) 1, 3, 5, and 7. The other points have constant Interest Level set to 1.	72
7.3	Finite State Machine (FSM) of the Drone Client and Server Application.	73
7.4	An excerpt of scenario configuration with an overlay of the models associated to the analyzed parts.	75
7.5	Logical flow to initialise and configure a scenario in IoD Sim.	76
7.6	<i>Simulation Development Platform</i> high-level schema.	76
7.7	Airflow Architectural Design.	77
7.8	An overview of the configuration of a generic model in Airflow.	78
7.9	A simple scenario with one drone and Zone Service Provider (ZSP) designed from scratch in Airflow.	79
7.10	Scenario #1.	80
7.11	Power consumption and peripheral state for each drone, in the first scenario.	80
7.12	Drones' trajectories with their power consumption, in the first scenario.	81
7.13	Received Signal Strength Indicator (RSSI) of each drone in the first scenario.	81
7.14	Trajectory design and eNB attachment for each drone, in the second scenario.	82
7.15	Drones' throughput, in the second scenario.	82
7.16	Memory occupancy for each drone, in the second scenario.	83
7.17	Scenario #3 simulation environment.	84
7.18	Ground Users (GUs) application latency of link combined by Wi-Fi, relay drone, and Long-Term Evolution (LTE).	85
7.19	GUs application latency over LTE-only link.	85
7.20	GUs application Packet Loss Ratio (PLR) for Scenario #3.	86
7.21	Performance evaluation of the different simulated scenarios.	87

List of Tables

2.1	Summary of notation.	9
3.1	Main notation.	17
3.2	Parameter settings.	25
4.1	Parameter settings.	34
6.1	Main notation adopted in this work.	51
6.2	Parameter set used in the simulations.	62
7.1	ns3::Drone properties in IoD-Sim.	70
7.2	Drone Peripherals Properties.	70
7.3	ns3::ConstantAccelerationDrone MobilityModel TypeId attributes.	73
7.4	ns3::ParametricSpeedDrone MobilityModel TypeId attributes.	73
7.5	Configuration parameters for <i>Telemetry Applications</i>	74
7.6	Configuration parameters for <i>Generic Traffic Applications</i>	74
7.7	Comparison of the total number of events, the real time taken to execute, and the simulated time of each scenario.	86

List of Acronyms

A2G	Air-to-Ground
ACO	Ant Colony Optimization
AoA	Angles of Arrival
AoD	Angles of Departure
AoV	Angles of View
API	Application Programming Interface
ATC	Air Traffic Control
BCD	Block Coordinate Descendent
BS	Base Station
CDF	Cumulative Distribution Function
CSI	Channel State Information
CSMA	Carrier Sense Multiple Access
D2D	Device-to-Device
DCL	Drone Control Layer
DOC	Drone Orchestration Center
DRL	Deep Reinforcement Learning
EbI	Eastbound Interface
FBS	Flying Base Station
FoVs	Fields of View
FoV	Field of View
FR1	Frequency Range 1
FSM	Finite State Machine
GA	Genetic Algorithm
GSL	GNU Scientific Library
GUI	Graphical User Interface
GU	Ground User
IMU	Inertial Measurement Unit

IoD-Sim Internet of Drones Simulator
IoD Internet of Drones
IoT Internet of Things
IRS Intelligent Reflective Surface
JSON JavaScript Object Notation
KPI Key Performance Index
LoS Line of Sight
LTE Long-Term Evolution
MCC Mission Control Center
MDP Markov Decision Process
MIMO Multiple-Input-Multiple-Output
MINLFP Mixed-Integer Non-Linear Fractional Programming
MINLP Mixed-Integer Non-Linear Programming
NAT Network Address Translation
NbI Northbound Interface
NLFP Non-Linear Fractional Programming
NLoS Non Line of Sight
NLoS Non-Line of Sight
NR-U 5G NR in Unlicensed spectrum
ns-3 Network Simulator 3
NT No-Transmission
OFDMA Orthogonal Frequency Multiple Access
OoS Out-of-Service
OSI Open Systems Interconnection
PaaS Platform as a Service
PDF Probability Density Function
PLR Packet Loss Ratio
PpoI Points of Interest
PPO Proximal Policy Optimization
PRU Passive Reflective Unit
QoS Quality of Service

RIS Reconfigurable Intelligent Surface
RoI Region of Interest
RSSI Received Signal Strength Indicator
RV Random Variable
SbI Southbound Interface
SCA Successive Convex Approximation
SINR Signal-to-Interference-plus-Noise Ratio
SNR Signal-to-Noise Ratio
SN Sensor Node
SSID Service Set Identifier
TCP Transmission Control Protocol
UAV Unmanned Aerial Vehicle
UDP User Datagram Protocol
VLC Visible Light Communications
VLC Visible Light Communication
WbI Westbound Interface
XML Extensible Markup Language
ZSP Zone Service Provider

Scientific Contributions

All the scientific contributions produced during the doctoral course are listed below.

International Journals:

- G. Iacovelli, P. Boccadoro and L. A. Grieco, "On the Interplay Between Energy and Memory Constraints in Optimized UAV Communications," in *IEEE Networking Letters*, vol. 2, no. 4, pp. 203-206, Dec. 2020.
- G. Iacovelli and L. A. Grieco, "Drone Swarm as Mobile Relaying System: A Hybrid Optimization Approach," in *IEEE Transactions on Vehicular Technology*, vol. 70, no. 11, pp. 12272-12277, Nov. 2021.
- G. Iacovelli, A. Coluccia and L. A. Grieco, "Channel Gain Lower Bound for IRS-Assisted UAV-Aided Communications," in *IEEE Communications Letters*, vol. 25, no. 12, pp. 3805-3809, Dec. 2021.
- G. Iacovelli and C. Iacovelli, "Representing Logic Gates Over Euclidean Space via Heaviside Step Function," in *Scientific Reports*, vol. 8009, no. 12, May 2022.
- G. Grieco, G. Iacovelli, P. Boccadoro and L. A. Grieco, "Internet of Drones Simulator: Design, Implementation, and Performance Evaluation," in *IEEE Internet of Things Journal*, 2022. (Accepted)

International Conferences:

- G. Iacovelli, P. Boccadoro and L. A. Grieco, "An Iterative Stochastic Approach to Constrained Drones' Communications," 2020 IEEE/ACM 24th International Symposium on Distributed Simulation and Real Time Applications (DS-RT), 2020, pp. 1-8.
- F. Vista, G. Iacovelli, and L.A. Grieco, "Quantum scheduling optimization for UAV-enabled IoT networks," In *Proceedings of the CoNEXT Student Workshop (CoNEXT-SW '21)*, 2021. <https://doi.org/10.1145/3488658.3493783>
- G. Grieco, G. Iacovelli, P. Boccadoro and L. A. Grieco, "On the Design of the Drone Control Layer," *European Wireless 2021; 26th European Wireless Conference*, 2021, pp. 1-7.
- G. Iacovelli, P. Boccadoro and L. A. Grieco, "An Optimization Approach to Energy-Efficient UAV Communications in Cellular Networks," *European Wireless 2021; 26th European Wireless Conference*, 2021, pp. 1-8.
- G. Iacovelli, P. Boccadoro and L. A. Grieco, "On Optimizing Drones' Communications Under Different Radio Coverage Conditions," 2021 29th Mediterranean Conference on Control and Automation (MED), 2021, pp. 324-329.

Under Review:

- G. Iacovelli, A. Coluccia and L. A. Grieco, "Multi-UAV IRS-assisted Communications: Multi-User Channel Modeling and Fair Sum-Rate Optimization via Deep Reinforcement Learning," 2022. Submitted to *IEEE Transaction on Vehicular Technologies*.

Introduction to the Internet of Drones

Internet of Drones (IoD) [1] is a disruptive paradigm which is expected to play a key role in upcoming 6G communications. IoD is conceived to ease mission design aspects, such as trajectory planning, mission design, flight control, resource optimization, and swarm management. The inherent versatility of drones, also named Unmanned Aerial Vehicles (UAVs), enables a vast plethora of applications [2]: sensing, delivery, surveying, patrolling, payload transportation, and Flying Base Stations (FBSs) [3], to name a few.

However, the heterogeneous technological landscape characterizing the IoD leads to interoperability issues especially in case of complex missions. Moreover, drone infrastructures require highly qualified personnel with expertise spanning over multiple aspects of hardware, mechatronics, network, and software engineering. As a matter of fact, the huge number of possible applications rely on specific hardware and software environments which hinder UAV employment at scale. In order to avoid such dependencies, a de-verticalizing architectural solution is needed to advance the deployment of drones in a structured manner. A flexible middleware would ease the management of multiple subsystems requirements [4].

One of the most interesting applications which have been recently investigated by the scientific community stem from the possibility to equip UAVs with Intelligent Reflective Surfaces (IRSs), also known as Reconfigurable Intelligent Surfaces (RISs). These metasurfaces are composed by a matrix of elements, namely Passive Reflective Units (PRUs), and have demonstrated superior capabilities of controlling the radio environment. In particular, IRSs are able to shift incident electromagnetic waves by a programmable phase, thus yielding beam-forming when optimally set. This unique property is at the basis of the emerging concept of Smart Radio Environments [5]. IRS-assisted wireless systems grant significant performance improvement, enhancing the channel quality perceived by communicating nodes. The high mobility of drones allows to change the IRS location to obtain a better Line of Sight (LoS) link and a lower pathloss. Although this combination grants a huge improvement of the channel quality, it comes with new challenges [6], [7] and in particular with the necessity of a dedicated channel model.

Despite the huge enhancement granted by IoD in several scenarios, UAVs are constrained devices which require sophisticated optimization strategies to be practically employed. The many facets of drones' limitations have been widely analyzed by the scientific community from different standpoints: (i) energy consumption [8]–[10], (ii) mission duration [11], [12], (iii) path and trajectory design [8], [10], [12]–[23], (iv) communication links and reliability [18], [24], [25], (v) achievable data rates [9], [10], [12], [15]–[17], [19], [25], [26], (vi) fine tuning on transmission power [10], [13], [15]–[17], [19], [20], [27], [28], and (vii) optimal data gathering [9], [29]. To properly optimize these aspects, it is necessary to encode the reference scenario into a problem of minimization/maximization of an objective function with respect to some parameters, subject to a set of constraints. Usually, the formulated optimization problem is non-convex and, hence, challenging to solve. In order to address this issue, two approaches are usually adopted:

- The first aims at the convexification of the problem. Indeed, in terms of tractability, convex programming brings mainly two huge advantages: (i) every local minimum is a global minimum and (ii) if the objective function is strictly convex, then the problem has at most one optimal point. Moreover, the optimal set is convex, which allows the

employment of common gradient-based algorithms. To achieve this objective, i.e. convexify the problem, several techniques can be applied, spanning from the computation of reasonable approximations to the introduction of iterative methods aiming at split the problem into multiple subproblems.

- The second approach is adopted when an accurate convex approximation cannot be obtained. In this case, heuristic methodologies are adopted, which however do not always guarantee that the optimal solution is retrieved. These methods include the well-know Genetic Algorithm (GA), the Ant Colony Optimization (ACO) algorithm, and many others. Machine Learning comprises several different methodologies, mostly data-driven, which cannot be applied in case of cutting-edge use cases, due to the lack of real measurements. Therefore, the Reinforcement Learning is usually employed to use the theoretical communication models.

To further facilitate the design and the prototyping phases of IoD systems, simulations are widely conceived as a useful aid. Indeed, a large scale adoption of UAVs can be time consuming and may require unfeasible costs, considering that IoD scenarios are characterized by different network topologies, communication technologies, drones' equipment, and software applications. Moreover, transmitted/received signals can experience different propagation conditions which depends on the environmental surrounding. To tackle this problem, simulators are an essential tool to ease the testing phase and state the readiness for real world exploitation. At the same time, simulators can be a learning tool for young professionals, engineering students and researchers to improve their knowledge and explore scenarios never considered before.

This thesis attempts to address the aforementioned technological issues in seven Chapters. In particular, Chapter 1 proposes a middleware solution, located between the transport and application layers, to cope with the intrinsic heterogeneity brought by IoD and hence to scale out the entire IoD ecosystem. Chapters 2, 3, and 4 discuss different optimization strategies to solve challenging problems related to different IoD scenarios. Some of the optimized aspects include kinetics-related parameters, transmission power, and scheduling plans. Chapter 5 provides a preliminary channel model for IRS-Assisted UAV-Aided Communications which is further generalized, and hence improved, in 6 where a Reinforcement Learning approach is employed to optimize an entire multi-UAV multi-user mission. Chapter 7 presents a comprehensive and open-source simulation platform for the IoD relying on accurate modeling techniques able to characterize all the features and limitations of UAVs. Finally, the last Chapter concludes the work and draws future research perspectives.

Chapter 1

On the Design of the Drone Control Layer

A key challenge for the IoD is represented by the technological fragmentation of its components. In this Chapter, a middleware solution, namely Drone Control Layer (DCL), is proposed to enable complex tasks which involve heterogeneous swarms. DCL is located between the transport and application layers and it is built on top of some basic properties: transparency, modularity, reprogrammability, and resiliency. These principles are crucial to scale out the entire IoD ecosystem, thus enabling complex mission design and development with different drones as actors, each one with proper and unique capabilities.

1.1 Introduction

UAV technological landscape is extremely heterogeneous, therefore interoperability issues could arise in complex missions. As a matter of fact, current available applications strongly depend on specific hardware and software environments which threatens/slows down drones' employment at scale. To boost the deployment of the IoD, such dependencies should be avoided thanks to a de-verticalizing platform that grants portability. Furthermore, drone infrastructures require highly qualified personnel with expertise spanning over multiple aspects of hardware, mechatronics, network, and software engineering. A middleware with a flexible software interface eases the management of multiple subsystems and their requirements to be satisfied [4].

Thanks to the recent advancements in cloud and edge computing, the use of a middleware in the IoD enables complex mission design, off-the-shelf software-defined components, integrated service provisioning, and management at a glance. Mission plan can be envisioned as a composition of different containers in a micro-service development environment [30]. On these bases, this work designs a middleware solution, namely DCL, that is located between the transport and application layers. It helps to define the underlying platform to abstract mission planning from drone peculiarities, while providing a safe and unified control structure. Meanwhile, the DCL grants the management of drones by means of a common set of interfaces with predictable responses. This facilitates applications' portability. The DCL identifies each elementary component of a given mission plan in order to further assign them to dedicated, yet specific, core modules, which cooperate towards mission accomplishment. The middleware has four interfaces which enable (i) the abstraction of underlying drivers and hardware, (ii) the use of common primitives for application development, (iii) the communication between drone and other logical entities, and (iv) core functionality extensibility. Inspired by cloud and edge practices, DCL-enabled drones working together with other logical entities in the IoD (e.g., Unmanned Traffic Management, Air Traffic Control (ATC) services, weather stations, and recharging stations) can be envisioned as resources, thus naturally becoming Platform as a Service (PaaS). Differently from the current state of the art [31]–[36], which envisions

drone as standalone entities, the DCL allows seamless management and coordination among swarms of heterogeneous drones. Moreover, the DCL enables the deployment across multiple unmanned systems. The proposed solution allows a wide applicability in several scenarios of interest.

1.2 Reference Scenario

The reference scenario (Figure 1.1) envisions swarms of drones assigned to different missions. Each mission is composed by a specific set of operations. Drones are grouped to form a

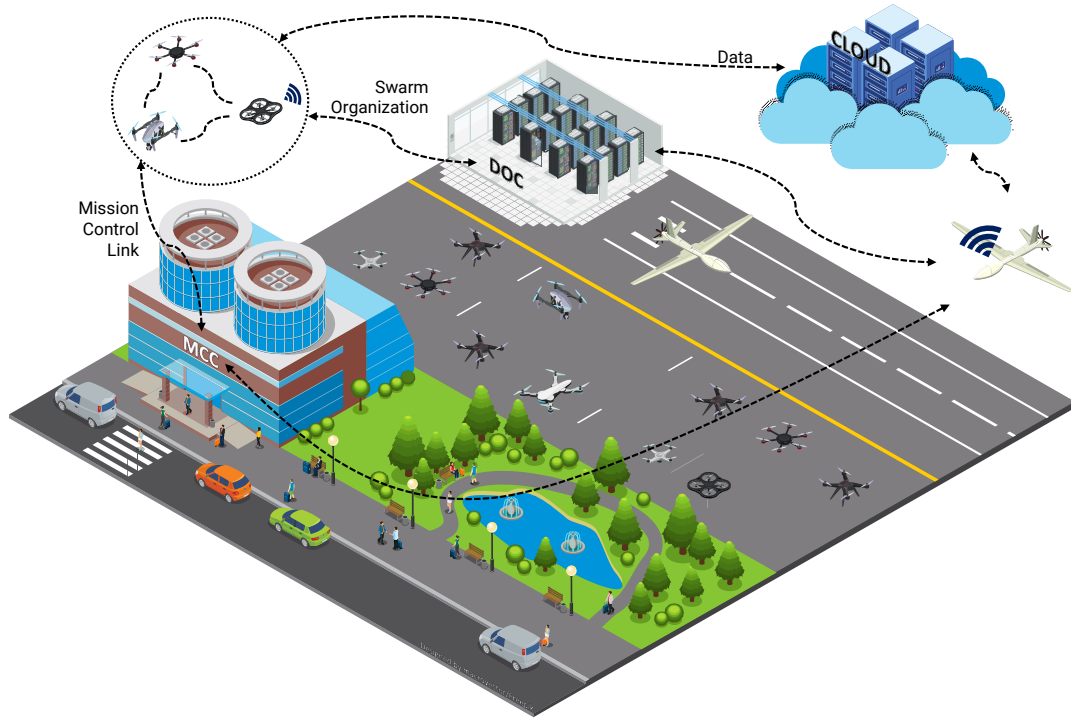


FIGURE 1.1: Schematic representation of the envisioned scenario.

swarm based on their on-board equipment and capabilities. Mission assignment and drones' enrollment are handled by dedicated logical entities:

- Mission Control Center (MCC): ground control infrastructure that designs the mission and monitors the swarm.
- Drone Orchestration Center (DOC): a central hub used to enroll drones in a mission by matching their characteristics with the operations to be done.
- Cloud-based Application Services: high-level functionalities for data mining and analysis related to the mission.

Drones are assigned to a specific mission, based on their capabilities, before take off. Once the swarm formation setup is completed, drones can start exchanging data with the MCC and execute the mission tasks.

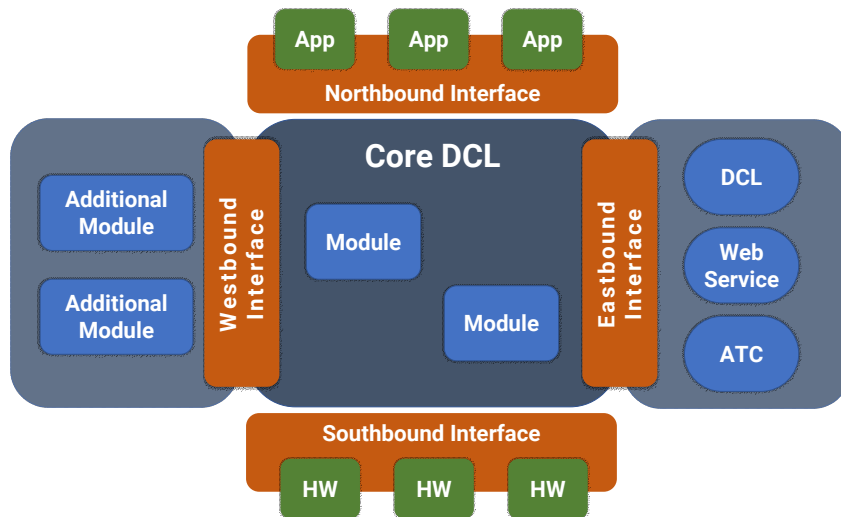


FIGURE 1.2: High level overview of DCL Interfaces and their types of applicability.

1.3 Interfaces Characterization

The architecture of the DCL is graphically introduced in Figure 1.2.

The DCL includes four main interfaces:

- The Northbound Interface (NbI) supports the development of high-level applications on top of the DCL to ease drone's control, communications and mission planning. As illustrated in Figure 1.3, the NbI provides an event-driven notification mechanism that eases information exchanges towards upper-level applications. For instance, drones in a swarm are able to react to updates and dynamically adjust their configuration to newer roles, if needed.
- The Southbound Interface (SbI) enables the interactions among the on-board resources and all the upper layers. Such a component solves the problems arising from vendor-related dependencies. In fact, this interface recognizes and supports multiple protocol stacks, radio interfaces, flight control primitives, and hardware resources in order to provide a set of software abstractions.
- The Westbound Interface (WbI) enables the development of DCL integration with additional modules and customized algorithms in order to optimize operations and support emerging applications. Extensions can cover a wide range of further developments, such as energy optimization.
- The Eastbound Interface (EbI) allows horizontal logical communications with other DCL-enabled entities to synchronize information, e.g., tasks, missions, and establish swarm networks. It also allows to establish direct connections between the drone and the outer world, i.e., Internet.

1.4 Core Characterization

The architectural components of the DCL, and its organization (Figure 1.3), can be considered as the base fabric of all primitives that assist the planning of a mission. These elements characterize the kernel of a general purpose drone.

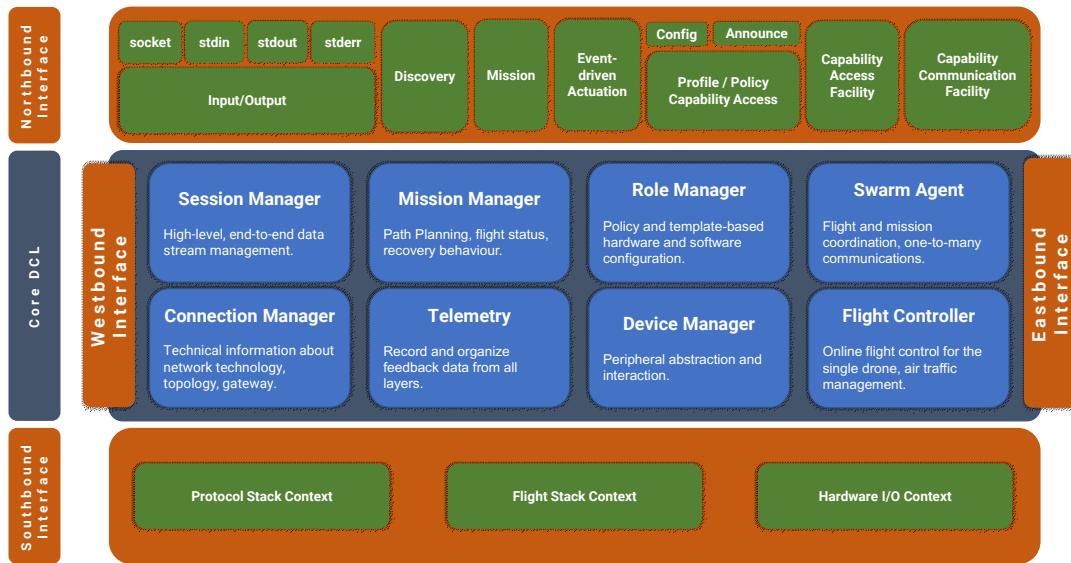


FIGURE 1.3: High level architecture of the DCL.

The Device Manager provides handlers to orchestrate drone’s hardware in terms of its capabilities. As a consequence, the Device Manager can be used to validate the suitability of the drone for the specific mission plan. Each hardware component is managed by a specific *Device Driver*. The latter exposes multiple high-level capability objects to communicate with such hardware component.

Leveraging the high-level description provided by the Device Manager, the Role Manager is responsible for matching drones to a mission plan. To this aim, the specific mission requirements are encoded in a Policy-based structure, which allows to detect the optimal *Role* for the drone.

The Telemetry module aggregates and elaborates drone data in an uniform and declarative way. Information derived from the SbI, e.g., IMU, state, and diagnostic data, converge into this module and are provided to applications through the NbI or external entities via the EbI.

The Connection Manager is focused on connection abstraction, which decouples data exchange from the protocol stack in use, thus yielding to context independent communications. The module maintains links with remote hosts using multiple radio and networking technologies according to drone *Role* and application requirements.

Leveraging the functionalities exposed by the Connection Manager, the Session Manager handles simultaneous logical links to ease context-based information exchange to/from external entities. The module provides a simple interface to open/close connections and send/receive end-to-end, broadcast and multicast information.

The Flight Controller provides a common set of commands that allows flight maneuvers, addressing hardware and software complexity. It has the responsibility of accepting such commands and controlling drone movements using its specific functionalities and drivers.

The Swarm Agent is specifically designed to handle information regarding mutual coordination among drones forming a swarm. Such messages allow swarm cooperation in order to achieve mission tasks.

Finally, the Mission Manager is focused on monitoring the drone status and on planning the sequence of tasks to be completed. Specifically, it orchestrates DCL modules by disseminating the information derived from the *Mission Plan*. Task scheduling and operational strategies can differ among missions or be limited by environmental restrictions.

Chapter 2

On the Interplay between Energy and Memory Constraints in Optimized UAV Communications

UAVs are key enablers in many emerging verticals but their resources are quite constrained, thus requiring sophisticated optimization approaches to prolong mission lifetime. The majority of scientific literature focuses on energy constraints, but when UAVs gather high-resolution multimedia signals, memory constraints become critical too. In this Chapter, a lean convex optimization framework is proposed to maximize the acquired/uploaded data, subject to joint energy and memory constraints.

2.1 Introduction

Drones are battery-supplied systems, on-board available energy is limited, which suggests that properly designed optimization routines are needed to maximize the lifetime of a mission. At the same time, it is worth remarking that energy is not the only constrained resource on board of a drone: memory availability is usually very limited and could hinder the development of services that gather high-resolution multimedia signals. For instance, considering a real drone, the onboard available memory can be quickly filled up if high-resolution video signals are acquired (e.g., 4k RAW at ~ 1 Gbps). Unfortunately, to the best of authors' knowledge, this aspect has been neglected by the majority of the scientific literature, with the sole exceptions of [9], [37]. In particular, in [37] beamforming techniques are discussed in the context of content provisioning by multiple UAVs to users requesting specific contents of interest. This reference deals with content management, assuming that any content is simplified to a unity dimension, and envisions storage capacity as the maximum amount of contents that can be stored by each drone as a constraint for the optimization problem. In [9], instead, the problem formulation considers a drone providing video signals to a multi-hop ground infrastructure based on Visible Light Communication (VLC): an algorithm has been proposed to manage on board resources without demonstrating its optimality. However, channel modelling is neglected.

The present contribution proposes an optimization framework that accounts for both energy and memory constraints in high-resolution multimedia acquisition services. In particular, the reference scenario (described in Sec. 2.2) involves a drone acquiring multimedia signals via onboard camera following a given trajectory. With the aim of offloading the memory, while increasing the responsiveness of the system, a low-resolution version of gathered data is uploaded leveraging air-to-ground communications (see Fig. 2.1). Here, uplink datastreams deserve particular attention, while downlink streams-related problems are not a major

concern*.

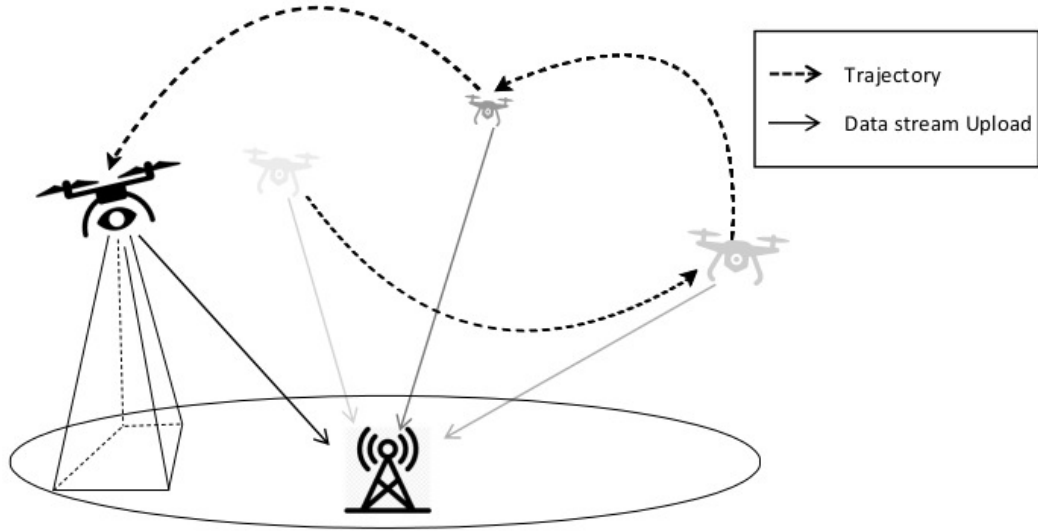


FIGURE 2.1: Drone acquiring multimedia signal while offloading to the BS.

The optimization framework (proposed in Sec. 2.3) aims at maximizing the amount of gathered and transmitted data, subject to contrasting bounds on available energy and memory. Clearly, continuous transmissions help avoid memory overflow but waste energy. Hence, finding the optimal amount of data to acquire from the camera and transmit to the ground motivates this contribution. Unfortunately, the formulation of this non-convex optimization problem makes the solution hard to be derived. To face this issue, an equivalent convex optimization formulation has been defined thanks to the introduction of a set of slack variables. Simulation results (presented in Sec. 2.4) demonstrate the effectiveness of the proposed approach in a realistic 5G scenario under several settings.

2.2 Problem Formulation

The total duration of the mission T is split into N time intervals with duration δ_t , each. Before the mission starts, the available memory and energy on-board are M_0 and E_0 , respectively. The amount of transmission power spent by the drone during the k -th time-slot for uploading operations is defined as P_k . Therefore, when the drone is offloading data, the channel capacity can be calculated according to the Shannon's equation:

$$R_k = B \log_2 \left(1 + \frac{P_k h_k}{\sigma^2} \right), \quad (2.1)$$

where $\sigma^2 = N_0 B$ is the noise power at the receiver's side, B is the available bandwidth, and h_k is the channel power gain.

The energy spent in the acquisition operations can be expressed as $E_c = \delta_t N P_c$, where P_c is the constant power consumed by the camera. To guarantee a uniform quality of gathered multimedia signals, the amount of acquired data over each time step is i_0 . The amount of uploaded data, instead, is equal to $o_k = \delta_t R_k$, during the k -th time step. When the UAV has

*Signalling and control data exchanges between the drone and the ground BS are neglected because herein assumed to be less demanding than the bandwidth needed to transfer multimedia signals.

completed its mission, the remaining amount of energy E_F can be expressed as:

$$E_F = E_0 - E_m - \delta_t \sum_{k=1}^N P_k - E_c. \quad (2.2)$$

being E_m is the total mechanical energy spent throughout the mission. As discussed in [9], during the mission the drone has to continuously control its operations based on surrounding physical conditions, so that the energy spent cannot be exactly known in advance: therefore, a random variable has to be considered to model E_m . Without loss of generality, it is assumed that E_m 's distribution is non-normal Gaussian where μ_m and σ_m are its mean and standard deviation, respectively. Nevertheless, the following considerations can be easily referred to other cases. Hence, it is possible to state that the probability that the leftover energy E_F is

Symbol	Description
N	Time intervals the mission is composed by [#].
δ_t	Duration of a time interval [s].
P_k	Transmission power at time interval k [W].
R_k	Datarate [Bps].
E_m	Overall mechanical energy consumption [J].
E_0/E_F	Initial/Final onboard available energies [J].
i_0	Acquired data at any time interval [B].
o_k	Uploaded data at time interval k [B].
M_0	Initial onboard available memory [B].
\mathcal{P}	Set of Transmission Power values [W]
\mathcal{W}	Set of slack variables.
γ	Mechanical energy required to accomplish the mission [J].
E_c	Onboard camera energy consumption [J].
ε	Out-of-Service probability.

TABLE 2.1: Summary of notation.

less than zero must be at most ε , i.e. Out-of-Service probability: $P_r(E_F < 0) \leq \varepsilon$ which is equivalent to

$$P_r(E_m > E_0 - \delta_t \sum_{k=1}^N P_k - E_c) \leq \varepsilon.$$

With a simple change of notation, the function that models the tail of E_m is $Q_1(x) = Q(\frac{x-\mu_m}{\sigma_m})$, from which it results $Q_1^{-1}(\varepsilon) = \gamma$. Therefore, it is possible to write:

$$E_c + \gamma + \delta_t \sum_{k=1}^N P_k \leq E_0. \quad (2.3)$$

The main focus of the present work is to solve (P1), that is formulated as follows:

$$(P1) : \max_{i_0, \mathcal{P}} N i_0 + \delta_t \sum_{k=1}^N R_k \quad \mathbf{s.t.}$$

$$E_c + \gamma + \delta_t \sum_{k=1}^N P_k \leq E_0 \quad (2.4)$$

$$k i_0 - \delta_t \sum_{j=1}^k R_j \leq M_0, \quad \forall k : 1 \dots N \quad (2.5)$$

$$\delta_t \sum_{j=1}^k R_k \leq k i_0, \quad \forall k : 1 \dots N \quad (2.6)$$

$$0 \leq P_k \leq P_{MAX}, \quad \forall k : 1 \dots N \quad (2.7)$$

$$i_0 \geq 0 \quad (2.8)$$

Problem (P1) aims at maximizing the amount of acquired and offloaded data subject to several constraints on available memory and energy, where \mathcal{P} is the set of $P_k \forall k$. In particular, in (2.4) it is explicitly stated that there is an upper-bound to the maximum amount of energy that the drone can use at any time during the mission. Similarly, in (2.5), memory limitation is presented. Respecting constraint (2.6) implies that the amount of data to be offloaded cannot exceed the available one, at any time during the mission. Finally, Equations (2.7) and (2.8) clarify the bounds for power consumption and acquired data, respectively. The main notation is summarized in Tab. 2.1

2.3 Proposed solution

As clearly results from its formulation, (P1) is a non-convex problem with reference to constraint (2.6). To tackle this issue, slack variables $\mathcal{W} = \{w_k \geq 0, \forall k\}$ can be introduced. Hence, (P1) can be reformulated as:

$$(P2) : \max_{i_0, \mathcal{P}, \mathcal{W}} N i_0 + \delta_t \sum_{k=1}^N w_k \quad \mathbf{s.t.}$$

$$(2.4), (2.7), (2.8),$$

$$k i_0 - \delta_t \sum_{j=1}^k w_j \leq M_0, \quad \forall k : 1 \dots N \quad (2.9)$$

$$R_k \geq w_k, \quad \forall k : 1 \dots N \quad (2.10)$$

$$\delta_t \sum_{j=1}^k w_j \leq k i_0, \quad \forall k : 1 \dots N. \quad (2.11)$$

Theorem 1. Solving problem (P1) is equivalent to solving problem (P2), for a sufficiently large M_0 .

Proof. The constraints in (2.10) can be used to reach the optimal solution of problem (P1) when the equality holds. Since, R_k represents an upper-bound for w_k , even when the equality is not respected, the condition will always be verified for increasing values of w_k , until equality holds again. Because of constrains (2.9) and (2.11), the value of w_k can be increased up to R_k if and only if M_0 is sufficiently large. ■

2.4 Performance evaluation

A simulation campaign has been carried out to evaluate, through MATLAB R2020a, (i) power consumption, (ii) achievable datarates, and (iii) memory occupation over time in a realistic 5G scenario. To this aim, two different settings were considered: the first configures an energy-bounded scenario, whereas the second is a memory-constrained one. Their main difference is the available energy E_0 before the mission starts. In the first case, the energy availability is far more than sufficient for completing the mission while offloading data with the maximum transmission power. In the second one, instead, E_0 value is restricted, thus requiring an optimized tuning of the transmission power. Therefore, the memory constraint is not dominating.

2.4.1 Parameter settings

Without loss of generality, the mean μ_m of the mechanical energy consumption E_m has been modeled as proposed in [38]. The mechanical power $P(V)$ spent by a drone, flying at a fixed quota H , is the same in every δ_t as it travels at an optimal constant cruise speed V [39]. For what concerns the channel model, instead, h_k has been defined as proposed in [16]. Starting from real, high-profile and low-profile drones, the two scenarios will envision $E_0 = \{213, 100\}$ kJ, respectively. A reference area of interest is monitored by a drone flying over a specific path, described by the well-known Theodorus Spiral, which is composed of right triangles, placed edge-to-edge [40]. As a property, for each and every point \mathbf{q}_k , $\forall k : 1 \dots N$ composing the spiral, the distance is constant. The latter property perfectly suites the constant velocity envisioned by the model. Once the drone takes-off from the starting point \mathbf{q}_0 , it reaches the quota at the point \mathbf{q}_1 and proceeds to acquire and upload data along its spiral path. The ground BS is placed in \mathbf{q}_b . The mission ends when the drone reverts back to \mathbf{q}_0 . Assuming a 16mm lens, it can be derived the equation set related to the Field of View (FoV)'s base and height as follows:

$$b = 2H \sec(\theta/2) \tan(\phi/2), \quad (2.12)$$

$$h = 2H \sec(\phi/2) \tan(\theta/2), \quad (2.13)$$

where θ and ϕ are the Angles of View (AoV) of b and h , respectively. It is worth noting that the spiral has been sized in order to obtain non-overlapping adjacent FoVs.

It is herein assumed that the reference scenario involves 5G networking technologies. In particular, $B = 20$ MHz bandwidth around 3 GHz in the 5G NR in Unlicensed spectrum (NR-U) is accounted for usage by the drone. According to specifications [41] and [42], such values are justified when referring to Frequency Range 1 (FR1). Further, in order to achieve a sensible throughput increment, the communication infrastructure implements Multiple-Input-Multiple-Output (MIMO) with space-division multiplexing and carrier aggregation. Since E_m is a Gaussian random variable (see Section 2.2), two values of the confidence interval $U_m = 3\sigma_m$ have been chosen: 5% and 10%. In other words, $\sigma_m = [0.016, 0.033] \cdot \mu_m$.

The configuration setting also includes: $M_0 = 8$ GB, $P_c = 10$ W, $\delta_t = 3$ s, $N = 199^{\ddagger\ddagger}$, $\varepsilon = 0.01$, $V = 16$ m/s, $\theta = 70.2^\circ$, $\phi = 43.3^\circ$, $H = 100$ m, $\mathbf{q}_b = [0 \ 0 \ 10]$, $\alpha = 2$, $\beta_0 = -60$ dB, $P_{MAX} = 1$ W, $N_0 = -174$ dBm/Hz and hence $\sigma^2 \approx -101$ dBm. Simulation parameters regarding the physical characterization of the drone are summarized in [38].

^{‡‡} N is sized to have a 4 complete rounds trajectory over a 1.5 km² area.

2.4.2 Discussion on results

Figure 2.2 shows the power trend over time in the energy-bounded cases obtained by solving (P2). Those values are obtained from the Shannon's equation, being $w_k = R_k$ the datarate[†]. These results show that a lower σ_m , i.e., a lower variability on the mechanical energy spent during the mission, implies a lower γ in the constraint (2.4), and, as a consequence, a larger amount of available power that can be used for transmission tasks. The results in the memory-bounded cases are omitted since the available energy is more than sufficient for completing the mission and, hence, the considered values are always equal to P_{MAX} . Figure 2.3 shows

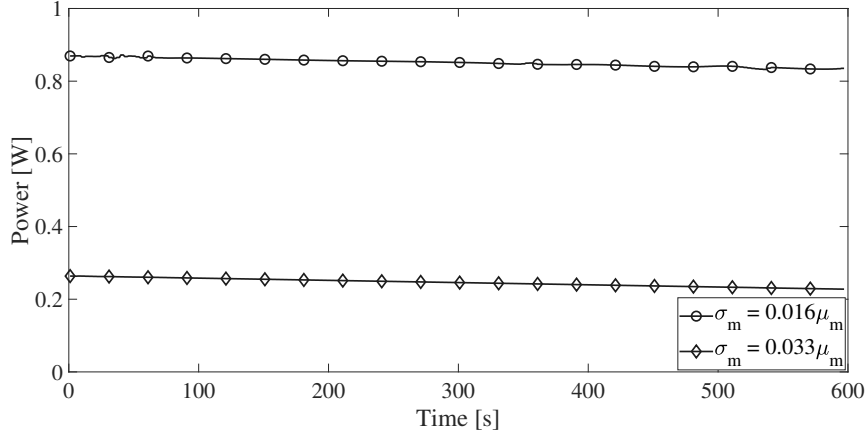


FIGURE 2.2: Power consumption in the energy-bounded configuration.

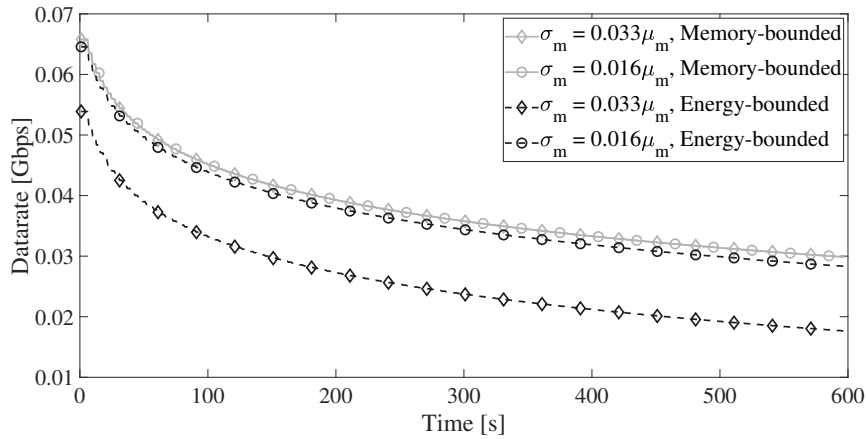


FIGURE 2.3: Datarates in all the configurations.

the optimal datarates obtained from (P2) in all considered settings: clearly σ_m does not affect w_k (or equivalently R_k) in memory-bounded scenarios because in those scenarios the energy limitations are less severe than memory ones. On the other hand, different datarates are obtained for different values of σ_m in energy-bounded scenarios: the higher σ_m , the higher γ in (2.4), the smaller the amount of energy that can be allocated to communication tasks (i.e., the lower w_k). Similar results hold for i_0 : being $\sigma_m = 0.033\mu_m$, in the energy-bounded configuration, 0.185 Gbps were acquired, whereas, with $\sigma_m = 0.016\mu_m$, the amount of acquired data grew up to 0.217 Gbps. In the memory-bounded ones, instead, the result was 0.221 Gbps for both values of σ_m . Figure 2.4 shows the available memory over time resulting from the

[†]It is worth to note that in all simulated settings, we verified that the optimal solution of (P2) always provides $w_k = R_k$.

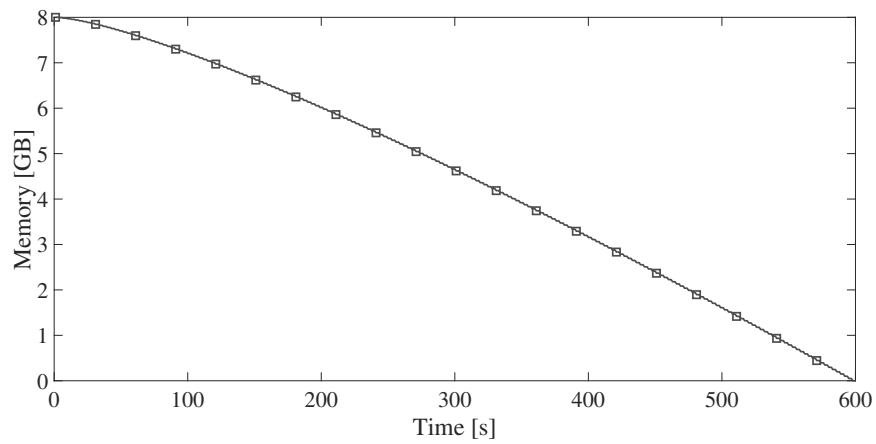


FIGURE 2.4: Memory availability trend over the mission.

solution of ($P2$). The results are identical in all configurations, since the difference between acquired and uploaded data does not change, regardless of the involved statistical fluctuations.

Chapter 3

An Optimization Approach to Energy-Efficient UAV Communications in Cellular Networks

In the IoD architecture, both drone-to-drone and drone-to-ground communications may take place, depending on the application scenario. In this Chapter it is presented a solution which designs an energy-efficient mission plan for a UAV in charge of surveying specific areas to acquire high-resolution video signals. Before the mission ends, it delivers gathered data to a ground control infrastructure, through BSs deployed in the region. The proposed framework takes into account constraints related to energy, kinematics, and connectivity, which leads to a Mixed-Integer Non-Linear Fractional Programming (MINLFP) problem, usually hard to solve.

3.1 Introduction

Communications among drones and ground infrastructure play a central role in several missions. Unfortunately, Unmanned Aerial Vehicles (UAVs) are affected by intrinsic limitations, involving energy efficiency, and mission endurance. Moreover, communication reliability is threatened by frequent topology changes, that may require adaptive routing strategies [43]. This holds even more when those constraints start to interplay, requiring sophisticated optimization strategies. Chapter 1 introduces the problem of a UAV involved in a video data gathering/provisioning scenario. Unfortunately, both trajectory and speed are not optimized, and the cellular network is simplified to a single BS.

This Chapter, instead, studies a reference scenario involving a drone acquiring and transmitting high-resolution video signals in specific areas of interest. This work considers Rician fading as channel model and several constraints, related to energy, kinematics and connectivity. The formulation proposed hereby leads to a Mixed-Integer Non-Linear Fractional Programming (MINLFP) problem which is challenging to solve. To tackle this issue, three sub-problems have been derived thanks to Block Coordinate Descendent (BCD) technique [44]. Each of them resulted to be non-convex. Therefore, they have been approached using Successive Convex Approximation (SCA) algorithm together with slack variables, thus leading to a quasi-optimal solution. The resulting optimization framework aims at maximizing energy efficiency, defined as the ratio between the achievable data rate and transmission energy. The goal is reached by deriving the maximum amount of data to gather from areas of interest, while optimizing trajectory to follow. In addition, both mission accomplishment and transmission reliability are modeled through Out-of-Service (OoS) probabilities. These

guarantee that the drone has enough energy to successfully complete the mission taking advantage of the maximum achievable channel quality. A simulation campaign has been carried out to demonstrate the effectiveness of the proposed framework, when multiple parameters are involved. In particular, it is shown the possible design of optimized trajectories that satisfy several constraints in different channel settings with multiple BSs and areas of interest.

3.2 Reference Scenario

The reference scenario (Figure 3.1) involves a UAV that takes-off from a starting point and has to reach specific areas of interest to acquire multimedia signals, via an onboard camera. Once the drone approaches the first area of interest, it starts to acquire and transmit data to a ground control infrastructure. The end-to-end communication leverages the BSs of the cellular network deployed in the region. In order to guarantee a proper quality of the gathered videos, the amount of stored information is uniform over time. Moving from one area to the next one, the UAV follows a trajectory that maximizes the energy efficiency, defined as the ratio between data rate and transmission energy expenditure. Further, the mission is subject to different constraints related to energy, kinematics and connectivity. Moreover, acquired data has to be transmitted before the mission ends, to provide information in a reasonable time and hence raising the system responsiveness. Clearly, in such scenario, uplink data streams is the major concern while downlink flow is less demanding in terms of bandwidth, and hence neglected. Finally, once the mission is completed, the drone heads towards the initial point.

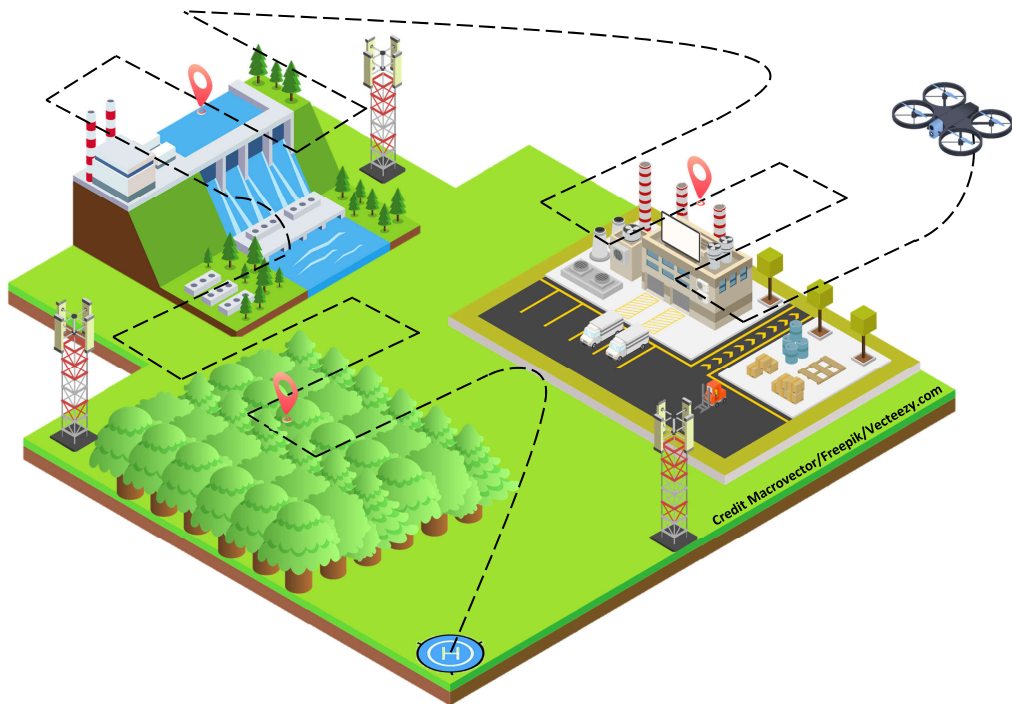


FIGURE 3.1: Reference Scenario.

3.3 System Model

The total duration of the mission T is split into N time intervals with duration δ_t . The trajectory pursued by the drone is also discretized in N points, i.e. $\mathbf{q}_k \in \mathbb{R}^2, k : 1 \dots N$. Before the mission starts, the available onboard energy is E_0 . Once the UAV takes-off, it has to reach

Param.	Description	Param.	Description
N	Number of time intervals [#]	β_0	Reference channel power gain [dB]
Z	Number of BSs [#]	R_k	Achievable data rate [$\frac{\text{bits}}{\text{s}}$]
L	Number of areas [#]	E_m	Mechanical energy consumption [J]
δ_t	Duration of a time interval [s]	E_0/E_F	Initial/Final available energies [J]
P_k	Transmission power [W]	i_k	Acquired data [bits]
E_0	Available onboard energy [kJ]	α	Pathloss coefficient [#]
ζ	Data rate OoS probability [#]	\mathbf{q}_k	Position vector [m]
ε	Energy OoS probability [#]	\mathbf{v}_k	Velocity vector [$\frac{\text{m}}{\text{s}}$]
K	K-factor [dB]	P_c	Camera power consumption [W]
N_0	Noise power density [$\frac{\text{dBm}}{\text{Hz}}$]	$\mathbf{\Omega}$	Matrix of BS positions [m]
B	Bandwidth [MHz]	d_k	Distance between UAV and BS [m]

TABLE 3.1: Main notation.

the areas of interest $\mathcal{S} = \{S_i, i : 1 \dots L\}$ and, in order to provide video data before the mission ends, transmit gathered information to the control center. Throughout its mission, the drone has to decide to which of the $Z \in \mathbb{N}$ BSs it is worth to attach. This issue is accounted through a binary vector $\mathbf{b}_k \in \{0, 1\}^Z$ that describes to which BS the UAV is transmitting in each time interval k . The components of this vectors are defined as $\mathbf{b}_{k,z}$ with $z : 1 \dots Z$. The trajectories to survey designated areas $\mathcal{U} = \{U_i, i : 1 \dots L\}$ are fixed and designed to maximize the total covered area with minimum overlapping. Their points are defined as $U_{i,\ell}, \ell : 1 \dots J_i$ with J_i as the number of points of which U_i is made. The main notation is summarized in Table 3.1.

3.3.1 UAV transmission

Let g_k be the Air-to-Ground (A2G) channel gain between the UAV and the BS. In every time interval k the channel gain can be expressed as

$$g_k = \sqrt{\mu_k} h_k, \quad (3.1)$$

where μ_k accounts for pathloss, while h_k ($\mathbb{E}(|h_k|^2) = 1$) takes care of small-scale effects through the Rician model. Hence, it is possible to define:

$$\mu_k = \beta_0 d_k^{-\alpha} \quad (3.2)$$

$$d_k = \sqrt{H^2 + \|\mathbf{q}_k - \mathbf{b}_k^T \mathbf{\Omega}\|^2} \quad (3.3)$$

with β_0 being the reference channel power gain, α the pathloss coefficient, and d_k the euclidean distance between the UAV and BS. Clearly, d_k depends on the position of the drone \mathbf{q}_k and on the BS to which it is transmitting, located at $\mathbf{b}_k^T \mathbf{\Omega}$. It is worth noting that $\mathbf{\Omega}$ is the matrix that has as rows the BSs' positions. Moreover, the Rician fading model is expressed by

$$h_k = \sqrt{\frac{K}{1+K}} \bar{h}_k + \sqrt{\frac{1}{1+K}} \tilde{h}_k, \quad (3.4)$$

where K is the Rician factor. $\bar{h}_k \in \mathbb{C}$ describes the deterministic component ($|\bar{h}_k| = 1$) while \tilde{h}_k , accounting for the Rayleigh fading, is assumed to follow a complex Gaussian distribution $\mathcal{CN}(0, 1)$. The transmission power consumed by the drone during the mission for uploading operations is defined as P_k . Therefore, according to Shannon's equation, it is possible to

compute the channel capacity as

$$\bar{R}_k = B \log_2 \left(1 + \frac{P_k \beta_0 |h_k|^2}{\sigma^2 d_k^\alpha} \right), \quad (3.5)$$

being $\sigma^2 = N_0 B$ the noise power at the receiver's side and B the available bandwidth. However, \bar{R}_k is not exactly known in each instant k , due to stochastic component in (3.5). Data transmission from the UAV requires a data rate R_k at each instant k , so that $\bar{R}_k \geq R_k$. Therefore, it is necessary to guarantee that the OoS probability p is below or equal to a threshold ζ :

$$\begin{aligned} p &= \mathbb{P}(\bar{R}_k < R_k) \\ &= \mathbb{P} \left(|h_k|^2 < \frac{\sigma^2 d_k^\alpha (2^{\frac{R_k}{B}} - 1)}{P_k \beta_0} \right) \\ &= F \left(\frac{\sigma^2 d_k^\alpha (2^{\frac{R_k}{B}} - 1)}{P_k \beta_0} \right) \\ &= F(u) \leq \zeta, \end{aligned} \quad (3.6)$$

where F is the Cumulative Distribution Function of $|h_k|^2$. With the aim of maximizing the total data received by every BS in a reliable manner, it is considered $p = \zeta$ i.e. the maximum tolerable OoS probability. F can be explicitly described as [45]

$$F(u) = 1 - Q_m \left(\underbrace{\sqrt{2K}}_{\lambda_1}, \underbrace{\sqrt{2(K+1)u}}_{\lambda_2} \right) = \zeta, \quad (3.7)$$

with Q_m denoting the Marcum Q-function. The correspondent inverse expression, with respect to λ_2 is:

$$\lambda_2 = Q_m^{-1}(\lambda_1, 1 - \zeta). \quad (3.8)$$

Unfortunately, the inverse Marcum Q-function has no closed-form expression and can only be computed with numeric techniques [46]. To tackle this issue, an approximated formulation can be used as in [45, Eq. 17]. Finally, reminding that

$$u = \frac{\sigma^2 d_k^\alpha (2^{\frac{R_k}{B}} - 1)}{P_k \beta_0} \simeq \frac{\lambda_2^2}{2(K+1)},$$

the expression of the achievable data rate can be derived as

$$R_k = B \log_2 \left(1 + \frac{P_k \beta_0 \lambda_2^2}{\sigma^2 d_k^\alpha 2(K+1)} \right) \quad (3.9)$$

$$= B \log_2 \left(1 + \frac{P_k \beta_0 f}{\sigma^2 d_k^\alpha} \right), \quad (3.10)$$

which satisfies requirement described in (3.6), where $f = \frac{\lambda_2^2}{2(K+1)}$.

3.3.2 Trajectories in the area of interest

The envisioned scenario is composed by a certain number L of areas of interest S_i for the surveying activity that are not contiguous. Each surveying area, that the drone has to inspect, is assumed to be circular with center $C_i = [\hat{x}_i \ \hat{y}_i]^T$ and radius R_i . The drone has to cover the circular areas entirely, and fly from one area, S_i , to the following one, S_{i+1} .

To ensure a thorough coverage of the areas of interest, thus maximizing the amount of gathered data, a snake-like trajectory U_i is involved. Each point of the trajectory is defined as $U_{i,\ell} = [x_\ell \ y_\ell]$. As a further constraint, the trajectory is designed in order to minimize the overlap between two adjacent Fields of View (FoVs). The FoV is defined as the projection of the lens aperture to the ground.

Therefore, the equation set related to the FoV's base and height is:

$$b_{FOV} = 2H \sec\left(\frac{\theta}{2}\right) \tan\left(\frac{\phi}{2}\right), \quad (3.11)$$

$$h_{FOV} = 2H \sec\left(\frac{\phi}{2}\right) \tan\left(\frac{\theta}{2}\right), \quad (3.12)$$

where θ and ϕ are the AoV of b_{FOV} and h_{FOV} , respectively. Drone's speed within each S_i is denoted by v^* and assumed to be constant to ensure a homogeneous data acquisition and minimum energy expenditure. Each inner trajectory U_i is designed accordingly.

3.3.3 Storage and Data model

Let i_k be the amount of data acquired during the mission, in each time interval k . The acquisition of video signals takes place only when the UAV is surveying an interest area:

$$i_k = \begin{cases} i_0 & \forall k \ni \mathbf{q}_k \in \mathcal{U} \\ 0 & \text{otherwise} \end{cases} \quad (3.13)$$

where i_0 guarantees a uniform quality of gathered multimedia signals. Indeed, the amount of acquired data is equal in each time interval k . Moreover, it is not possible to transmit more data than the acquired amount, and it is equivalent to impose

$$\sum_{j=1}^k \delta_t R_j \leq \sum_{j=1}^k i_j, \quad \forall k : 1 \dots N. \quad (3.14)$$

Since gathered data must be transmitted before the mission ends, it is required that:

$$\sum_{k=1}^N i_k - \sum_{k=1}^N \delta_t R_k = 0. \quad (3.15)$$

Finally, it is herein assumed that the drone has always enough room in memory to store acquired data.

3.3.4 UAV power consumption model

The mechanical energy E_m consumed by a drone flying at speed \mathbf{v}_k [20][22] is expressed as:

$$E_m = \sum_{k=1}^N \delta_t \left(\hat{P}_0 + \frac{3\hat{P}_0 \|\mathbf{v}_k\|^2}{U_{tip}^2} + \frac{1}{2} d_0 \hat{\rho} s A \|\mathbf{v}_k\|^3 \right) + \sum_{k=1}^N \delta_t \hat{P}_i \left(\sqrt{1 + \frac{\|\mathbf{v}_k\|^4}{4v_0^4}} - \frac{\|\mathbf{v}_k\|^2}{2v_0^2} \right)^{\frac{1}{2}}. \quad (3.16)$$

Here, \hat{P}_0 and \hat{P}_i are, respectively, the blade profile and the induced power while the drone is in hovering status. Further, U_{tip} is the rotor blade's tip speed, v_0 is the mean rotor induced

velocity when hovering, d_0 is the fuselage drag ratio and s is the rotor solidity. Moreover, A and $\hat{\rho}$ represent the rotor disc area and air density, respectively. The energy spent in the acquisition operations, in the interest areas, can be expressed as

$$E_c = P_c \sum_{k \ni \mathbf{q}_k \in \mathcal{U}} \delta_t,$$

in which P_c is the constant power consumed by the camera. When the UAV has completed its mission, the remaining amount of energy E_F can be expressed as:

$$E_F = E_0 - E_m - \sum_{k=1}^N \delta_t P_k - E_c. \quad (3.17)$$

However, the hypothesis concerning this value being deterministic is weak. In fact, it is unreasonable to assume that the exact value of the energy spent can be known in advance[9], [47]. This is motivated by the fact that during the mission the drone has to continuously control its operations based on surrounding physical conditions. Therefore, E_m has to be considered as a random variable, thus ensuring a stochastic generalized formulation. Without loss of generality, it is assumed that E_m 's distribution is non-normal Gaussian where \bar{E}_m and \tilde{E}_m are its mean and standard deviation, respectively. In particular, the average mechanical energy spent is assumed to follow the expression in (3.16), while \tilde{E}_m is defined as a certain percentage of the mean [9], [47].

Hence, it is possible to state that the probability that the leftover energy E_F is less than zero must be at most ε , i.e. OoS probability:

$$\mathbb{P}(E_F < 0) \leq \varepsilon, \quad (3.18)$$

which is equivalent to

$$\mathbb{P}(E_m > E_0 - \sum_{k=1}^N \delta_t P_k - E_c) \leq \varepsilon.$$

With a simple change of notation, the function that models the tail of E_m is $Q(x) = Q(\frac{\gamma - \bar{E}_m}{\tilde{E}_m})$, from which it results:

$$\gamma = \bar{E}_m + Q^{-1}(\varepsilon) \tilde{E}_m, \quad (3.19)$$

being $Q(\cdot)$ the Q-function and γ the energy required to accomplish the mission with a maximum OoS probability ε . Finally, it is possible to write:

$$E_c + \gamma + \sum_{k=1}^N \delta_t P_k \leq E_0. \quad (3.20)$$

3.4 Problem Formulation

Let be $\mathcal{Q} = \{\mathbf{q}_k\}$, $\mathcal{V} = \{\mathbf{v}_k\}$, $\mathcal{P} = \{P_k\}$, $\mathcal{I} = \{i_k\}$ and $\mathcal{B} = \{\mathbf{b}_k\}$. The main focus of the present work is to solve the following problem, which is formulated as follows:

$$(P1) : \max_{\mathcal{Q}, \mathcal{V}, \mathcal{P}, \mathcal{B}, \mathcal{I}} \frac{\sum_{k=1}^N R_k}{\sum_{k=1}^N \delta_t P_k} \quad \text{s.t.} \quad \mathbf{b}_k \in \{0, 1\}^Z \quad \forall k : 1 \dots N \quad (3.21)$$

$$\sum_{z=1}^Z \mathbf{b}_{kz} = 1, \quad \forall k : 1 \dots N \quad (3.22)$$

$$E_c + \gamma + \sum_{k=1}^N \delta_t P_k \leq E_0 \quad (3.23)$$

$$\sum_{k=1}^N i_k - \sum_{k=1}^N \delta_t R_k = 0 \quad (3.24)$$

$$\sum_{j=1}^k \delta_t R_j \leq \sum_{j=1}^k i_j, \quad \forall k : 1 \dots N \quad (3.25)$$

$$0 \leq P_k \leq P_{MAX}, \quad \forall k : 1 \dots N \quad (3.26)$$

$$i_k \geq 0 \quad \forall k : 1 \dots N \quad (3.27)$$

$$i_k = 0 \quad \forall k \ni' \mathbf{q}_k \notin \mathcal{U} \quad (3.28)$$

$$i_{k+1} = i_k \quad \forall k : 1 \dots (N-1) \ni' \{\mathbf{q}_k, \mathbf{q}_{k+1}\} \in \mathcal{U} \quad (3.29)$$

$$\mathbf{q}_{k+1} = \mathbf{q}_k + \delta_t \mathbf{v}_k \quad \forall k : 1 \dots (N-1) \quad (3.30)$$

$$\mathbf{q}_k = U_{i,\ell} \quad \forall \mathbf{q}_k \in \mathcal{U}, \forall i : 1 \dots L, \forall k : 1 \dots N, \forall \ell : 1 \dots J_i, \quad (3.31)$$

$$\mathbf{q}_0 = \mathbf{q}_N \quad (3.32)$$

$$\|\mathbf{v}_k\| \leq v_{MAX} \quad \forall k : 1 \dots N \quad (3.33)$$

$$\frac{\|\mathbf{v}_{k+1} - \mathbf{v}_k\|}{\delta_t} \leq a_{MAX} \quad \forall k : 1 \dots N \quad (3.34)$$

Problem (P1) aims at maximizing the amount of acquired data, through data rate R_k previously defined in (3.10), and at the same time minimizing the total transmission energy expenditure. In particular, (3.22) states that the UAV cannot be attached to multiple BSs at the same time. In (3.23) it is explicitly stated that there is an upper-bound to the maximum amount of energy that the drone can use at any time during the mission. (3.24) imposes that the gathered data has to be transmitted before the landing. Respecting constraint (3.25) implies that the amount of data to be offloaded cannot exceed the available one, at any time during the mission. Equations (3.26) and (3.27) clarify the bounds for power consumption and acquired data, respectively. As stated in (3.13), constraints (3.28) and (3.29) ensure that (i) when the UAV moves among interest areas no acquisition is performed (ii) otherwise, the amount of acquired data is equal in each step. (3.31) guarantees that within these areas the survey trajectory is fixed. Equation (3.30) denotes the 2D movement laws of the drone and (3.32) sets a correspondences between start and end points. Finally, constraints (3.33) and (3.34) state the upper bounds of speed and acceleration.

3.5 Proposed Solution

Problem (P1) is a MINLFP problem, which is hard to solve. To tackle this issue, it can be solved by applying BCD technique[44]. Therefore, (P1) is split into three sub-problems, iteratively solved in sequence until a quasi-optimum solution is reached.

3.5.1 Sub-Problem 1: Gathered/Transmitted Data Optimization

The optimization procedure is limited to the transmission and the acquisition of data. Therefore, position \mathcal{Q} , velocity \mathcal{V} , and reference BS \mathcal{B} are assumed to be fixed. Hence, the new sub-problem (P2) is

$$(P2) : \max_{\mathcal{P}, \mathcal{I}} \frac{\sum_{k=1}^N R_k}{\sum_{k=1}^N \delta_t P_k} \quad \text{s.t.} \quad (3.23) - (3.29).$$

Unfortunately, this is a non-convex optimization problem due to non-affine constraint (3.24) and concave constraint (3.25). Similarly to [47], it is possible to reformulate this problem in a convex formulation. Consider a set of slack variables $\mathcal{W} = \{w_k \geq 0, \forall k\}$. Hence, (P2) can be reformulated as:

$$(P2.1) : \max_{\mathcal{P}, \mathcal{I}, \mathcal{W}} \frac{\sum_{k=1}^N w_k}{\sum_{k=1}^N \delta_t P_k} \quad \text{s.t.} \quad (3.23), (3.26) - (3.29)$$

$$R_k \geq w_k, \quad \forall k : 1 \dots N \quad (3.35)$$

$$\sum_{k=1}^N i_k - \sum_{k=1}^N \delta_t w_k = 0 \quad (3.36)$$

$$\sum_{j=1}^k \delta_t w_j \leq \sum_{j=1}^k i_j, \quad \forall k : 1 \dots N. \quad (3.37)$$

Solving problem (P2) is equivalent to solving problem (P2.1). In fact, when the equality is not respected, the condition will always be verified for increasing values of w_k , until equality holds. Even if (3.36) or constraints in (3.37) are satisfied for strict inequality in (3.35), P_k can always be decreased to obtain a higher objective function value, until equality holds again. It results that problem (P2.1) is now a standard Non-Linear Fractional Programming (NLFP) problem, with a concave numerator and a convex denominator, which is guaranteed to converge to a prescribed accuracy $\xi < 0$. It can be solved thanks to different algorithms, such as Dinkelbach's Algorithm [48].

3.5.2 Sub-Problem 2: Trajectory-related Parameters Optimization

Given the previously optimized variables, i.e. \mathcal{P} and \mathcal{I} , and an initial guess for \mathcal{B} , this resolution stage aims at retrieving movement-related parameters. Therefore the envisioned sub-problem is:

$$(P3) : \max_{\mathcal{Q}, \mathcal{V}} \sum_{k=1}^N R_k \quad \text{s.t.} \quad (3.23), (3.30) - (3.34),$$

Unfortunately, (P3) is a NLFP problem, which is hard to solve. In fact, the objective function numerator is neither convex nor concave with respect to \mathcal{Q} . Moreover, the denominator and (3.23) are non-convex functions due to the presence of γ and in particular of \bar{E}_m . First, to tackle this issue, a new set of slack variables $\mathcal{T} = \{\tau_k \geq 0, k : 1 \dots N\}$ is introduced. Hence, the reformulated problem, with the new expression of \bar{E}_m , i.e. \hat{E}_m , is:

$$(P3.1) : \max_{\mathcal{Q}, \mathcal{V}, \mathcal{T}} \sum_{k=1}^N R_k \quad \text{s.t.} \quad (3.30) - (3.34), \quad (3.38)$$

$$E_c + \hat{\gamma} + \sum_{k=1}^N \delta_t P_k \leq E_0 \quad (3.39)$$

$$\tau_k^2 + \frac{\|\mathbf{v}_k\|^2}{v_0^2} \geq \frac{1}{\tau_k}, \quad \forall k : 1 \dots N \quad (3.40)$$

where γ has been replaced by $\hat{\gamma} = \hat{E}_m + Q^{-1}(\varepsilon)\tilde{E}_m$ and

$$\tau_k = \left(\sqrt{1 + \frac{\|\mathbf{v}_k\|^4}{4v_0^4}} - \frac{\|\mathbf{v}_k\|^2}{2v_0^2} \right)^{\frac{1}{2}} \quad (3.41)$$

$$\hat{E}_m = \sum_{k=1}^N \delta_t \left(\hat{P}_0 + \frac{3\hat{P}_0 \|\mathbf{v}_k\|^2}{U_{tip}^2} + \frac{1}{2} d_0 \hat{\rho} s A \|\mathbf{v}_k\|^3 + \hat{P}_i \tau_k \right) \quad (3.42)$$

It can be verified that \hat{E}_m is now convex with respect to \mathcal{V} and \mathcal{T} . Clearly, (P3.1) is equivalent to (P3) when equality holds in (3.40). Otherwise, it is always possible to decrease τ_k until the equality holds again.

Further, regarding the numerator of the cost function, it worth knowing that first-order Taylor expansion is a global underestimator [16] for convex functions. Therefore, for any local point \mathbf{q}_k^η , it is possible to lower-bound R_k as follows:

$$R_k \geq \hat{R}_k = A_k^\eta - I_k^\eta (\|\mathbf{q}_k - \mathbf{b}_k^T \boldsymbol{\Omega}\|^2 - \|\mathbf{q}_k^\eta - \mathbf{b}_k^T \boldsymbol{\Omega}\|^2), \quad (3.43)$$

where

$$A_k^\eta = B \log_2 \left(1 + \frac{\Gamma_k}{d_k^{\eta\alpha/2}} \right), \quad (3.44)$$

$$I_k^\eta = \frac{B \log_2 e (\alpha/2) \Gamma_k}{d_k^{\eta\alpha/2} (d_k^{\eta\alpha/2} + \Gamma)}, \quad (3.45)$$

$$\Gamma_k = \frac{P_k \beta_0 f}{\sigma^2}, \quad (3.46)$$

$$d_k^\eta = H^2 + \|\mathbf{q}_k^\eta - \mathbf{b}_k^T \boldsymbol{\Omega}\|^2. \quad (3.47)$$

In the same way, the new non-convex constraint (3.40) can be lower-bounded by its first-order Taylor approximation, for any local point $(\mathbf{v}_k^\eta, \tau_k^\eta)$, thus obtaining the following new constraint:

$$\tau_k^{\eta 2} + 2\tau_k^\eta (\tau_k - \tau_k^\eta) - \frac{\|\mathbf{v}_k^\eta\|^2}{v_0^2} + \frac{2}{v_0^2} \mathbf{v}_k^{\eta T} \mathbf{v}_k \geq \frac{1}{\tau_k}. \quad (3.48)$$

As a consequence, the new formulation of the equivalent problem is:

$$(P3.2) : \max_{\mathcal{Q}, \mathcal{V}, \mathcal{T}} \sum_{k=1}^N \hat{R}_k \quad \mathbf{s.t.}$$

$$(3.30) - (3.32), (3.33), (3.34), (3.39), (3.48).$$

Problem (P3.2) is convex and can be solved by applying SCA technique.

3.5.3 Sub-Problem 3: Base Stations Optimization

Finally, given $\{\mathcal{P}, \mathcal{I}, \mathcal{Q}, \mathcal{V}\}$, the BS-related set \mathcal{B} is optimized. This contribution is two-folded: it is of relevance when studying trajectory, and, at the same time, it must be related to transmissions. The related problem is stated as follows:

$$(P4) : \max_{\mathcal{B}} \sum_{k=1}^N R_k \quad \mathbf{s.t.}$$

$$(3.21).$$

Problem (P4) is a Mixed-Integer Non-Linear Programming (MINLP) which is challenging to solve. In fact, (3.21) is a integer constraint and R_k presents the same issue observe in (P3). First, it is necessary to relax (3.21), by substituting it with:

$$0 \leq \mathbf{b}_{kz} \leq 1, \quad \forall k : 1 \dots N, \forall z : 1 \dots Z \quad (3.49)$$

Furthermore, thanks to the analysis and substitutions done in (3.43), similar consideration can be done once again for local point \mathbf{b}_k^η :

$$R_k \geq \tilde{R}_k = A_k^\eta - I_k^\eta (\|\mathbf{q}_k - \mathbf{b}_k^T \boldsymbol{\Omega}\|^2 - \|\mathbf{q}_k - \mathbf{b}_k^{\eta T} \boldsymbol{\Omega}\|^2), \quad (3.50)$$

and hence it is possible to reformulate (P4) as:

$$(P4.1) : \max_{\mathcal{B}} \sum_{k=1}^N \tilde{R}_k \quad \mathbf{s.t.}$$

$$\sum_{z=1}^Z \mathbf{b}_{kz} = 1, \quad \forall k : 1 \dots N \quad (3.51)$$

$$0 \leq \mathbf{b}_{kz} \leq 1, \quad \forall k : 1 \dots N, \forall z : 1 \dots Z \quad (3.52)$$

Problem (P4.1) can be efficiently solved through the SCA technique. After the optimization process, the relaxed values of \mathcal{B} are processed in order to identify the best BS. This corresponds to replace the maximum value of \mathbf{b}_k with 1 and remaining values with 0, for each k .

It is worth specifying that isolating this sub-problem is necessary, because the relaxation of (3.21) implies a simultaneous connection to multiple BSs. Even though this is mathematically verified, it is forbidden in real cellular networks. Therefore, it may lead to wrong results if \mathcal{B} was optimized together with other variables, such as \mathcal{I} . Indeed, this will lead to wrong maximization results in terms of R_k and hence of acquired data.

3.5.4 Overall iterative algorithm

Based on above discussion, a quasi-optimal solution of problem (P1) can be found by iteratively solving (P2.1), (P3.2) and (P4.1). Specifically, the optimization variables are divided into three groups: the first for gathered/transmitted data, i.e. $\{\mathcal{P}, \mathcal{I}\}$, the second are movement-related, i.e. $\{\mathcal{Q}, \mathcal{V}\}$, and the third deals with the handover procedure, i.e. \mathcal{B} . Note that the BCD technique convergence is not guaranteed due to the presence of Dinkelbach's Algorithm in (P2.1). Nonetheless, numerical results will show that convergence is always achieved.

3.6 Numerical results

This section proposes an evaluation of the envisioned solution. To this aim, parameters settings are herein introduced and motivated. The results of the conducted simulation campaign are discussed in details.

3.6.1 Parameter settings

The UAV flies in a reference area $900 \times 800 \text{ m}^2$ wide. The starting point of the mission, and the ending one, are the same, i.e., $[0 \ 400]^T$. The mission time T has been divided in $N = 250$ instants of $\delta_t = 1 \text{ s}$ each. Since E_m is a Gaussian random variable, the confidence interval $U_m = 3\tilde{E}_m$ has been chosen, so that $\tilde{E}_m = 0.016 \cdot \bar{E}_m$. Without loss of generality, three BSs are randomly deployed in the reference area, and assumed to be located in $[100 \ 100]^T$, $[700 \ 700]^T$ and $[600 \ 100]^T$, respectively. The drone has to follow a mission plan that includes two interest areas placed in $[300 \ 700]^T$ and $[800 \ 300]^T$ with a radius $R_1 = R_2 = 100 \text{ m}$. The inner trajectories have been designed according to specifications discussed in Subsection 3.3.2. Specifically, given the parameters of the energy consumption model, $v^* = 10 \text{ m/s}$ has been derived as the minimum of \bar{E}_m . As for the Rician factor K , two values have been considered, i.e., 5 dB and 10 dB. The remaining parameters are summarized in Table 3.2.

Param.	Value	Param.	Value
N	250 [#]	β_0	-60 [dB]
Z	3 [#]	ϕ	70.2 [°]
L	2 [#]	θ	43.3 [°]
δ_t	1 [s]	H	50 [m]
P_{MAX}	1 [W]	α	2 [#]
E_0	150 [kJ]	v_{MAX}	50 [$\frac{\text{m}}{\text{s}}$]
v^*	10 [$\frac{\text{m}}{\text{s}}$]	a_{MAX}	5 [$\frac{\text{m}}{\text{s}^2}$]
ξ	0.5 [#]	P_c	10 [W]
ζ	0.01 [#]	\hat{P}_i	88.6279 [#]
ε	0.01 [#]	\hat{P}_0	79.8563 [#]
K	5/10 [dB]	d_0	0.6 [#]
N_0	-174 [$\frac{\text{dBm}}{\text{Hz}}$]	v_0	4.03 [m/s]
B	20 [MHz]	U_{tip}	120 [$\frac{\text{m}}{\text{s}}$]
Ω	{100 100}	A	0.503 [m^2]
	{700 700}	$\hat{\rho}$	1.225 [$\frac{\text{kg}}{\text{m}^3}$]
	{600 100} [m]	s	0.05 [#]

TABLE 3.2: Parameter settings.

3.6.2 Optimization Framework

The OoS probability ζ is reasonably assumed to be as low as 0.01. Therefore, the chosen two values for K are addressed by the second case of [45, Eq. 17]. Figure 3.2 shows the

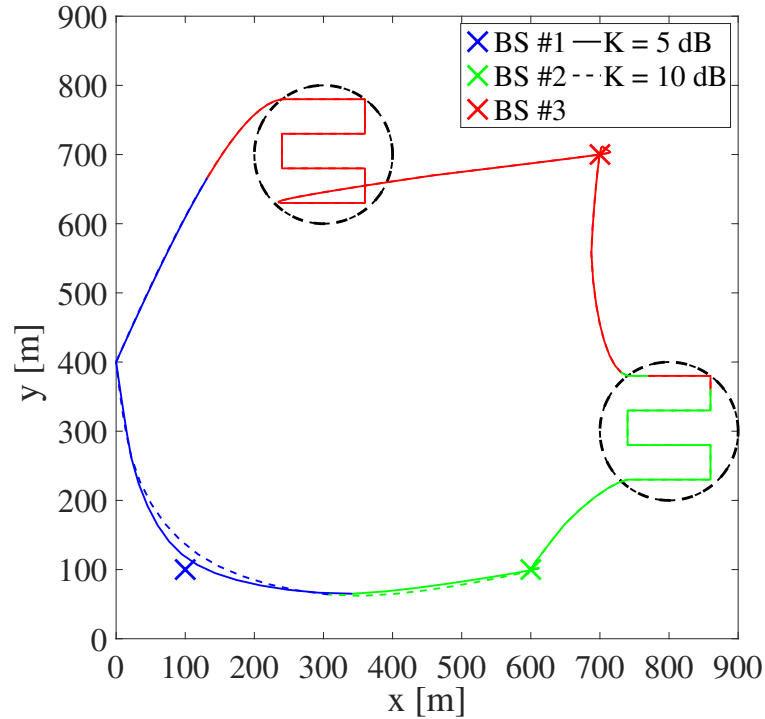


FIGURE 3.2: Trajectories and BS association for different K-factors.

trajectories followed by the UAV during missions characterized by two different K-factors. It is worth noting that the drone, after each survey, and in both cases, approaches the BS in order to maximize the total transmitted and, hence, acquired data. As expected, the drone is always attached to the BS which grants the best channel quality. Even if the trajectory plans are very similar, in case of lower LoS connectivity the UAV travels closer to BSs #1. The

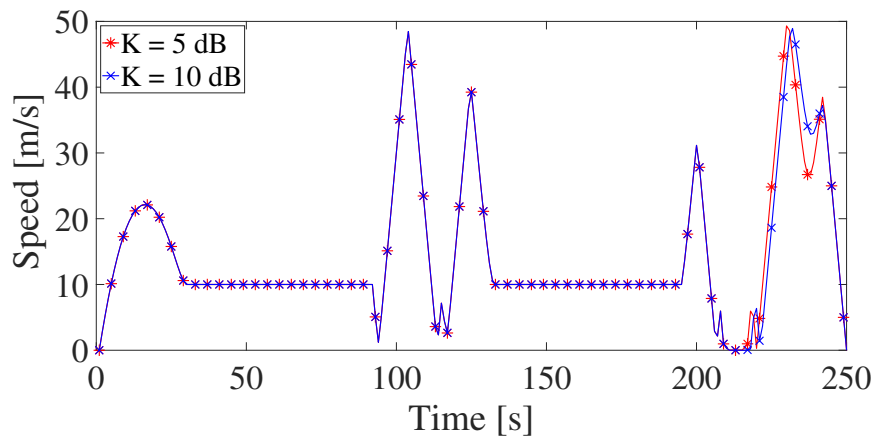


FIGURE 3.3: Speed profiles over time.

speed trends, represented in Figure 3.3, are very similar too. In particular, when the first area is reached, the speed drops down to the fixed value $v^* = 10$ m/s, at which the UAV is able to guarantee a uniform high resolution video acquisition and minimum energy consumption. The same rationale holds for S_2 . It is worth noting that, a little difference is observed in the

last part of the mission, which corresponds to the difference highlighted for trajectory plans. Figures 3.4 and 3.5 show the data rate and the transmission power over the mission in the

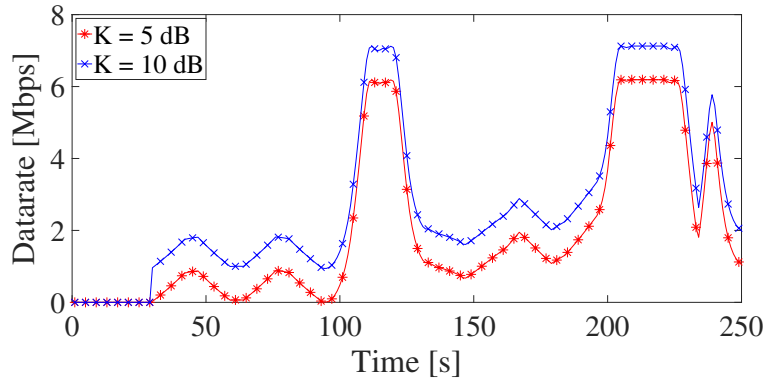


FIGURE 3.4: Data rate comparison over time.

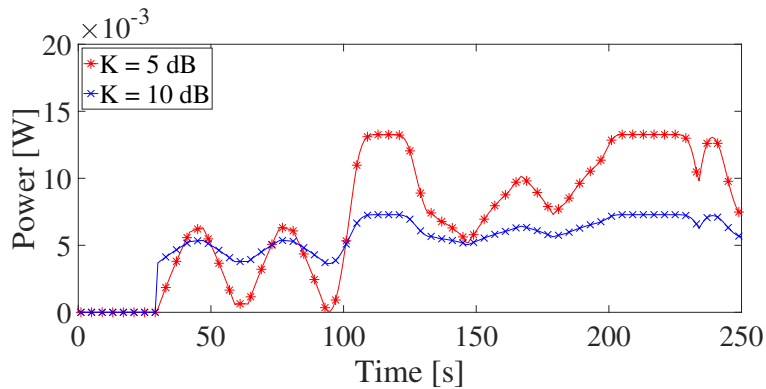


FIGURE 3.5: Transmission power comparison over time.

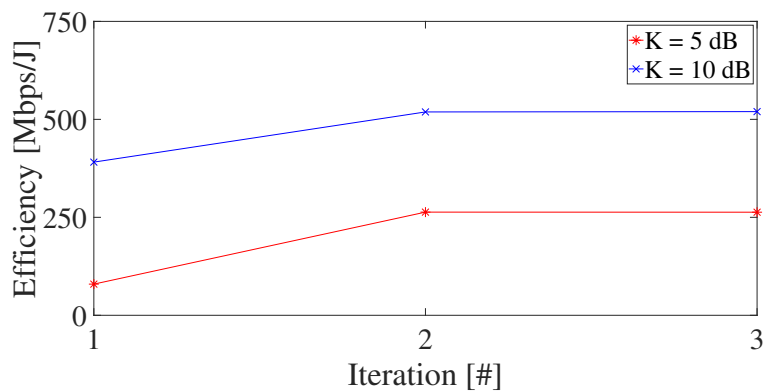


FIGURE 3.6: Convergence of the overall algorithm.

two configurations, respectively. Clearly, with a lower K , i.e., 5 dB, the channel capacity is limited with respect to the higher value, i.e., 10 dB. In fact, for a mission that lasts 250 s, the average data rates are 1.935 Mbps and 2.762 Mbps, respectively.

At the beginning, the data rate value is as low as zero since no data have been acquired, yet. The same holds for the transmission power. As the mission goes by, it acquires data from the areas of interest. The overall amounts for $K = 5$ dB and $K = 10$ dB are 483.76 Mbit and 690.67 Mbit, respectively. As a matter of fact, a larger K -factor implies a higher data rate

due to the better LoS condition. Finally, Figure 3.6 shows two major contributions. First, the proposed combination of BCD, SCA and Dinkelbach's Algorithm monotonically converges to a quasi-optimal solution. A formal mathematical proof will be investigated in future works. Second, it is demonstrated the non-negligible effect of K on energy efficiency. Simulation results shown that moving from $K = 5$ dB to $K = 10$ dB, increases energy efficiency from 269.39 Mbps/J to 532.23 Mbps/J, which corresponds to an increase of 98%.

Chapter 4

Drone Swarm as Mobile Relaying System: A Hybrid Optimization Approach

This Chapter investigates a scenario in which a swarm of UAVs enables the communication between a set of SNs and a control center. Considering a general fading channel model, a MINLP problem is formulated to maximize the overall amount of relayed data by jointly optimizing trajectory and scheduling plan of each drone. Combining convex optimization and ACO algorithm, a quasi-optimal solution is obtained.

4.1 Introduction

The possibility to organize UAVs in swarms makes cooperative Device-to-Device (D2D) communications play a central role. In such a context, the allocation of network resources becomes a challenging, yet fundamental, aspect. For instance, [49] studies a scenario in which cellular networks and relays improve communications among devices. A network coding aided cooperative diversity scheme is designed from which the system data rate expression is derived considering interference among nodes. A distributed low-complexity algorithm is developed to solve a coalition formation game, thus jointly optimizing the allocation of spectrum resources and the relay selection.

Drone swarms are also employed for service provisioning, thus demanding specific solutions to optimize data processing and dissemination. To this aim, [50] proposes a holistic middleware, which employs reinforcement learning to dynamically balance the broadcast rate and knowledge loss rate. Moreover, a cooperative dissemination method is designed to fine-tune storage and energy allocation among drones.

Besides their great potential, drones are constrained devices which require sophisticated optimization strategies to finely tune on-board resources (e.g., energy and memory). From this point of view, several contributions jointly analyze different aspects such as energy-efficiency, trajectory design, achievable data rates, memory occupancy, and scheduling planning. In this regard, in [51] multiple sources and destinations communicate through a UAV-enabled relaying system. The contribution aims at maximizing the minimum throughput of all links and, at the same time, optimizing the UAVs' trajectories and transmission power levels. However, the considered channel model only accounts for LoS link. Moreover, it is assumed that for each source-destination couple a dedicated drone is deployed, which cannot always be realized.

[52] studies a scenario in which pairs of transceivers need the support of drones to communicate, thus acting as relays. The aim is to minimize the total service time, consisting of communication time and flight duration. Although interesting, the article assumes that for each pair of nodes the communication is enabled by only one drone in the whole service time.

Besides, the time spent flying between two consecutive locations is not employed to serve more nodes.

This work overtakes limitations highlighted above by envisioning a scenario consisting of a variable number of UAVs and Sensor Nodes (SNs), deployed in a reference area. Throughout the mission, the same SN can be served by different drones that continuously relay signals to a control center, through a BS. Each SN is equipped with a wake-up receiver which allows (i) to recover from sleep state, thus saving energy, and (ii) to identify the associated relaying UAV. Communications reliability has also been considered by imposing a low OoS probability.

The present contribution aims at maximizing the total amount of relayed data, while optimizing trajectory and scheduling plan of each drone of the swarm, considering a general fading channel model. A Mixed-Integer Non-Linear Programming (MINLP) problem stems from the derived mathematical formulation, which is challenging to solve. Hence, a quasi-optimal solution is achieved by leveraging BCD and SCA techniques combined with the Ant Colony Optimization (ACO) algorithm[53]. To the best of authors' knowledge the combination of these techniques has never been employed before. Simulation results demonstrate the validity of the proposed solution in different parameter configurations and with respect to a benchmark scheme derived from [52]. Moreover, even if convergence cannot be mathematically proved, it is numerically verified.

The rest of the work is organized as follows: Section 4.2 describes the adopted system model. Sections 4.3 and 4.4 discuss the problem formulation and the proposed solution. Section 4.5 presents the obtained numerical results.

4.2 System Model

The total mission time T is split into N intervals with duration δ_t . The swarm is composed by D drones having the same hardware and capabilities, each one following a trajectory discretized into N points $\mathbf{q}_{k,z} \in \mathbb{R}^2, k : 1 \dots N, z : 1 \dots D$ and flying at speed $\mathbf{v}_{k,z} \in \mathbb{R}^2, k : 1 \dots N, z : 1 \dots D$, at constant altitude H . In particular, $\mathbf{Q}_k = \{\mathbf{q}_{k,z} \forall z\}$ and $\mathbf{V}_k = \{\mathbf{v}_{k,z} \forall z\}$ refer to the position and velocity matrices of all drones in the k -th timeslot, respectively. UAVs are in charge of relaying data from S SNs placed in $\mathbf{u}_j \in \mathbb{R}^2, j : 1 \dots S$ to a control center, through a BS located at \mathbf{q}_b . Without loss of generality, when the swarm flies over the area of interest, the generic j -th SN has already generated an amount of sensed data o_j . Throughout their mission, each UAV has to select which SN to serve, in each timeslot. This scheduling plan is described through a binary 3D matrix $\mathbf{X} \in \{0, 1\}^{N \times D \times S}$ which is composed by vectors, i.e., $\mathbf{X} = \{\mathbf{x}_{k,j} \forall k, j\}$, or equivalently by 2D matrices, i.e., $\mathbf{X} = \{\mathbf{X}_j \forall j\}$, that can be obtained through indexes. It is assumed that each drone has two dedicated antennas for SN-UAV and UAV-BS links, while SNs have just one. As a consequence, it is necessary to guarantee that (i) each SN is served by only one drone in each timeslot k and (ii) a UAV can communicate just with one SN per time interval. Moreover, the transmission power of j -th SN and z -th UAV are defined as P_j^S and $P_{k,z}^D$, respectively. Indeed, it is assumed that drones adopt a power control mechanism such that $0 \leq P_{k,z}^D \leq P_{\text{MAX}}^D$, whereas SNs can only transmit with a fixed power level. Besides, the whole system employs a modulation scheme such that interference among SNs, UAVs, and BS is avoided. Let $g_{k,j}$ be the gain of the quasi-static flat-fading channel between the relaying UAV and a SN j . In each time interval k , $g_{k,j} = \sqrt{\mu_{k,j}} h_{k,j}$ where $\mu_{k,j}$ accounts for pathloss, while $h_{k,j}$ ($\mathbb{E}(|h_{k,j}|^2) = 1$) is a random variable describing a generic channel model coefficient, which remains unchanged in each k

but may vary among timeslots [17]. Therefore, it results that:

$$\mu_{k,j} = \beta_0 d_{k,j}^{-\alpha}, \quad (4.1)$$

$$d_{k,j} = \sqrt{H^2 + \|\mathbf{x}_{k,j} \mathbf{Q}_k^\top - \mathbf{u}_j\|^2}, \quad (4.2)$$

where β_0 is the reference channel power gain, α is the pathloss coefficient, and $d_{k,j}$ is the euclidean distance of the UAV-SN link. Thanks to Shannon's equation, the channel capacity is defined as:

$$c_{k,j} = B_j \log_2 \left(1 + \frac{P_j^s \beta_0 |h_{k,j}|^2}{\sigma_j^2 d_{k,j}^\alpha} \right), \quad (4.3)$$

with $\sigma_j^2 = N_0 B_j$ as the noise power and B_j as the available SN's bandwidth. However, $c_{k,j}$ cannot be exactly known in each instant k since the instantaneous channel coefficient $|h_{k,j}|$ is stochastic. Therefore, to guarantee that the OoS probability $p_{k,j}$ remains below or equal to a threshold ζ , it is necessary to impose the following:

$$\begin{aligned} p_{k,j} &= \mathbb{P}(c_{k,j} < r_{k,j}^s) \\ &= \mathbb{P} \left(|h_{k,j}|^2 < \frac{\sigma_j^2 d_{k,j}^\alpha (2^{\frac{r_{k,j}^s}{B_j}} - 1)}{P_j^s \beta_0} \right) \\ &= F \left(\frac{\sigma_j^2 d_{k,j}^\alpha (2^{\frac{r_{k,j}^s}{B_j}} - 1)}{P_j^s \beta_0} \right) \leq \zeta, \quad \forall \begin{array}{l} z : 1 \dots D - 1, \\ j : 1 \dots S, \end{array} \end{aligned} \quad (4.4)$$

being $F(\cdot)$ the Cumulative Distribution Function (CDF) of $|h_{k,j}|^2$. In order to ensure a reliable transmission, the maximum tolerable OoS probability is considered i.e. $p_{k,j} = \zeta \forall k, s$. Therefore, the maximum achievable data rate is

$$r_{k,j}^s = B_j \log_2 \left(1 + \frac{P_j^s \beta_0 F^{-1}(\zeta)}{\sigma_j^2 d_{k,j}^\alpha} \right), \quad (4.5)$$

where $F^{-1}(\cdot)$ denotes the inverse CDF. With the same rationale, it is possible to define the BS-UAV channel model and, hence, the data rate as:

$$r_{k,z}^D = B_z \log_2 \left(1 + \frac{P_{k,z}^D \beta_0 F^{-1}(\zeta)}{\sigma_z^2 d_{k,z}^\alpha} \right), \quad (4.6)$$

with $d_{k,z} = \sqrt{H^2 + \|\mathbf{q}_{k,z} - \mathbf{q}_b\|^2}$. For the sake of notation, define the data rate vectors of sensors as $\mathbf{r}_j^s = \{r_{k,j}^s, \forall k\}$ and $\mathbf{r}_k^D = \{r_{k,j}^s, \forall j\}$.

It is worth specifying that, in this work, the exchange of signaling and control data is neglected because assumed to be less demanding than data transmission in terms of time and bandwidth[47].

4.3 Problem Formulation

Let be $\mathbf{Q} = \{\mathbf{Q}_k \forall k\}$ and $\mathbf{V} = \{\mathbf{V}_k \forall k\}$. The main focus of the present work is to solve the following problem, which is formulated as follows:

$$(P1) : \max_{\mathbf{X}, \mathbf{Q}, \mathbf{V}} \sum_{j=1}^S \sum_{z=1}^D (\mathbf{X}_j^T \mathbf{r}_j^S)_z \quad \text{s.t.}$$

$$\mathbf{x}_{k,z} \mathbf{r}_k^S \leq \hat{r}_{k,z}^D, \quad \forall k : 1 \dots N, z : 1 \dots D, \quad (4.7)$$

$$\delta_t \sum_{z=1}^D (\mathbf{X}_j^T \mathbf{r}_j^S)_z \leq o_j, \quad \forall j : 1 \dots S, \quad (4.8)$$

$$\mathbf{q}_{k+1,z} = \mathbf{q}_{k,z} + \delta_t \mathbf{v}_{k,z}, \quad \forall k : 1 \dots (N-1), \quad (4.9)$$

$$\mathbf{q}_{1,z} = \mathbf{q}_{N,z}, \quad \forall z : 1 \dots D, \quad (4.10)$$

$$\|\mathbf{v}_{k,z}\| \leq v_{\text{MAX}}, \quad \forall k : 1 \dots N, z : 1 \dots D, \quad (4.11)$$

$$\frac{\|\mathbf{v}_{k+1,z} - \mathbf{v}_{k,z}\|}{\delta_t} \leq a_{\text{MAX}}, \quad \forall k : 1 \dots (N-1), z : 1 \dots D, \quad (4.12)$$

$$\mathbf{v}_{1,z} = \mathbf{v}_{N,z} = \mathbf{0}, \quad \forall z : 1 \dots D, \quad (4.13)$$

$$\|\mathbf{x}_{k,z}\| = 1, \quad \forall k : 1 \dots N, z : 1 \dots D, \quad (4.14)$$

$$\|\mathbf{x}_{k,j}\| = 1, \quad \forall k : 1 \dots N, j : 1 \dots S. \quad (4.15)$$

Problem (P1) aims at maximizing the total amount of transmitted data from SNs to BS through every drone of the swarm by jointly optimizing their scheduling plan \mathbf{X} , trajectory \mathbf{Q} and speed \mathbf{V} . In particular, (4.7) states that SNs' data rate cannot be higher than $\hat{r}_{k,z}^D$ i.e. maximum achievable relaying UAVs' data rate. Constraint (4.8) implies that, for each j , transmitted sensing data must be lower than the acquired. Equations (4.9) and (4.10) describe the 2D movement of UAVs and the correspondence between start/end point of the trajectory. (4.13) imposes the initial/final speed of drones. Constraints (4.11) and (4.12) denote the speed and acceleration upper-bounds, respectively. Finally, (4.14) and (4.15) guarantee that a drone serves just one SN and viceversa.

4.4 Proposed Solution

Problem (P1) is a MINLP problem, which is hard to solve. To tackle this issue, BCD technique is applied. Therefore, (P1) is divided into two sub-problems, which are more tractable, and alternately solved until convergence to a quasi-optimal solution is achieved.

4.4.1 Sub-Problem 1: Trajectory optimization

The first sub-problem aims at optimizing the trajectory-related parameters \mathbf{Q} and \mathbf{V} . Therefore, \mathbf{X} is initialized or assumed to be known. The envisioned sub-problem is:

$$(P2) : \max_{\mathbf{Q}, \mathbf{V}} \sum_{j=1}^S \sum_{z=1}^D (\mathbf{X}_j^T \mathbf{r}_j^S)_z \quad \text{s.t.}$$

$$(4.7) - (4.13).$$

Unfortunately, (P2) is a non-convex problem. In fact, the objective function is neither convex nor concave with respect to \mathbf{Q} , as well as constraints (4.7) and (4.8). However, they are convex whereas $\|\mathbf{q}_{k,z} - \mathbf{q}_b\|$ and $\|\mathbf{x}_{k,j} \mathbf{Q}_k - \mathbf{u}_j\|^T$ are considered. To tackle this issue, SCA technique

can be employed in order to obtain an approximate optimum solution. Indeed, reminding that first-order Taylor expansion is a global underestimator for convex functions, it is possible to lower-bound $\hat{r}_{k,z}^D$ for each local point $\mathbf{q}_{k,z}^r$ as:

$$\hat{r}_{k,z}^D \geq \tilde{r}_{k,z}^D = A_{k,z}^r - I_{k,z}^r (\|\mathbf{q}_{k,z} - \mathbf{q}_b\|^2 - \|\mathbf{q}_{k,z}^r - \mathbf{q}_b\|^2), \quad (4.16)$$

where $\Gamma = \frac{P_{\text{MAX}}^D \beta_0 F^{-1}(\zeta)}{\sigma_d^2}$, $I_{k,z}^r = \frac{B_z \log_2 e(\alpha/2)\Gamma}{d_{k,z}^r \alpha/2 (d_{k,z}^r \alpha/2 + \Gamma)}$, and $A_{k,z}^r = B_z \log_2 \left(1 + \frac{\Gamma}{d_{k,z}^r \alpha/2}\right)$. Clearly, the same rationale can be used for $r_{k,j}^S$:

$$r_{k,j}^S \geq \tilde{r}_{k,j}^S = A_{k,j}^r - I_{k,j}^r (\|\mathbf{x}_{k,j} \mathbf{Q}_k\|^T - \mathbf{u}_j^2 - \|\mathbf{x}_{k,j} \mathbf{q}_k^r\|^T - \mathbf{u}_j\|^2). \quad (4.17)$$

However, it can be verified that (4.17) is still a concave function, raising an issue with respect to constraints (4.7) and (4.8). Therefore, introducing a set of slack variables $\mathcal{W} = \{w_{k,j} \geq 0, \forall k, j\}$, the sub-problem is reformulated as follows:

$$(P2.1) : \max_{\mathbf{Q}, \mathbf{V}, \mathcal{W}} \sum_{j=1}^S \sum_{z=1}^D (\mathbf{X}_j^T w_j)_z \quad \text{s.t.}$$

$$\tilde{r}_{k,j}^S \geq w_{k,j}, \quad \forall j : 1 \dots S, \quad (4.18)$$

$$\mathbf{x}_{k,z} w_k^T \leq \tilde{r}_{k,z}^D, \quad \forall k : 1 \dots N, z : 1 \dots D, \quad (4.19)$$

$$\sum_{z=1}^D (\mathbf{X}_j^T w_j)_z \leq o_j, \quad \forall j : 1 \dots S, \quad (4.20)$$

$$(4.9) - (4.13). \quad (4.21)$$

Solving problem (P2.1) is equivalent to solve (P2). In fact, $w_{k,z}$ can always be increased until equality holds in (4.18). Even if there exists a constraint in (4.19) or (4.20) satisfied for strict inequality in (4.18), there will always be $\mathbf{x}_{k,z}$ or \mathbf{X}_j^T such that (4.7) and (4.8) hold. Furthermore, (P2.1) is a standard convex optimization problem, which can be solved by several tools such as CVX[54]. Since the objective function is lower-bounded by a finite value it is guaranteed to converge.

4.4.2 Sub-Problem 2: Scheduling Optimization

Given $\{\mathbf{Q}, \mathbf{V}\}$, the scheduling plan \mathbf{X} is hereby optimized. The related problem is stated as follows:

$$(P3) : \max_{\mathbf{X}} \sum_{j=1}^S \sum_{z=1}^D (\mathbf{X}_j^T \mathbf{r}_j^S)_z \quad \text{s.t.}$$

$$(4.7), (4.8), (4.14), (4.15).$$

(P3) is a MINLP problem and due to its combinatorial nature is challenging to solve. Therefore, the well-known ACO algorithm is employed, which has been proved to converge to optimality [53], [55]. (P3) is a large-scale optimization problem since the number of possible states exponentially grows with N and D , i.e., $(S+1)^{ND}$. To face this issue, problem (P3) is solved for each k and z , where it is put in place a colony of L ants which can move among the possible $S+1$ states, described by a matrix $\mathbf{M}^{S \times (S+1)} = [\mathbf{0} \ \mathbf{I}_S]^T$. In particular, $\mathbf{0}$ defines the No-Transmission (NT) state, while \mathbf{I}_S , i.e., identity matrix, expresses the communication with

one of the S SNs. As a consequence, constraint (4.14) is inherently satisfied. The ant's transition among states happens with a probability described by $\mathbf{e}_{k,z} = \{e_{k,z,m} \forall m : 1 \dots S+1\}$, defined as:

$$\mathbf{e}_{k,z} = \frac{\tau_{k,z} (1 - \sum_{i=1}^{z-1} \frac{\mathbf{e}_{k,i}}{D})^\gamma}{\sum_{m=1}^{S+1} e_{k,z,m}}, \quad \forall k : 1 \dots N, z : 1 \dots D, \quad (4.22)$$

being $\tau_{k,z} = \{\tau_{k,z,m} \forall m\}$ the vector of $S+1$ pheromone trails, $1 - \sum_{i=1}^{z-1} \frac{\mathbf{e}_{k,i}}{D}$ a specific coefficient which discourages adoption of states already selected by other components of the swarm, thus satisfying (4.15), and γ an exponential penalty coefficient. Further, fixed j , a stochastic vector κ is updated with the current cumulative sum of sensing transmitted data:

$$\kappa_j = \kappa_j + \mathbf{x}_{k,z} \mathbf{r}_k^S e_{k,z,j+1}, \quad \forall k : 1 \dots N, z : 1 \dots D. \quad (4.23)$$

Therefore, employing penalty function method, the resultant expression $K_{l,k,z}$, with $l : 1 \dots L$, that characterizes each ant is:

$$K_{l,k,z} = 1 - \underbrace{\frac{\mathbf{x}_{k,z} \mathbf{r}_k^S}{r_{\text{MAX}}^S + \varepsilon}}_{\text{Cost}} + \eta_1 \underbrace{\frac{\mathbf{x}_{k,z} \mathbf{r}_k^S - \hat{r}_{k,z}^D}{\hat{r}_{k,z}^D}}_{\mathcal{C}_1} + \eta_2 \underbrace{\frac{\kappa_j - o_j}{o_j}}_{\mathcal{C}_2}.$$

$Cost$ tends to zero when SN-UAV data rate $\mathbf{x}_{k,z} \mathbf{r}_k^S$ approaches its maximum, i.e., r_{MAX}^S , that cannot be reached. Indeed, $\varepsilon > 0$ is an arbitrary small value that guarantees $K_{l,k,z} \neq 0, \forall l, k, z$. Moreover, \mathcal{C}_1 and \mathcal{C}_2 account for the violation of constraints (4.7) and (4.8), whereas η_1 and η_2 are weight penalty coefficients. Since all terms are normalized between 0 and 1, it is sufficient that η_1 and η_2 are greater, by some orders of magnitude, than $\max(Cost) + \max(\mathcal{C}_1) + \max(\mathcal{C}_2) = 3$ whereas constraints are not satisfied, zero otherwise. Finally, the pheromone trails $\tau_{k,z,m}$ are updated at each iteration as follows:

$$\tau_{k,z,m} = \tau_{k,z,m} + \frac{1}{\sum_{l=1}^L K_{l,k,z}}, \quad \tau_{k,z,m} = (1 - \rho) \tau_{k,z,m}, \quad (4.24)$$

with ρ denoting the evaporation rate. The overall time complexity of the proposed algorithm is $\mathcal{O}(RN DL)$, where R denotes the number of iterations. It is worth specifying that the conceived ACO-based approach leads to a quasi-optimal solution, which theoretically does not guarantee the convergence of the BCD technique. However, it has been numerically verified that convergence to a stationary point is always achieved in considered scenarios.

Parameter	Value	Parameter	Value
η_1, η_2	10^3 [#]	β_0	-60 [dB]
R	500 [#]	α	2 [#]
L	40 [#]	ρ	0.05 [#]
δ_t	1 [s]	H	50 [m]
K_c	10 [dB]	v_{MAX}	50 [m/s]
P_{MAX}^D	1 [W]	a_{MAX}	5 [m/s ²]
ζ	0.01 [#]	\mathbf{q}_b	[500 500] ^T [m]
N_0	-174 [dBm/Hz]	γ	78.2 [#]

TABLE 4.1: Parameter settings.

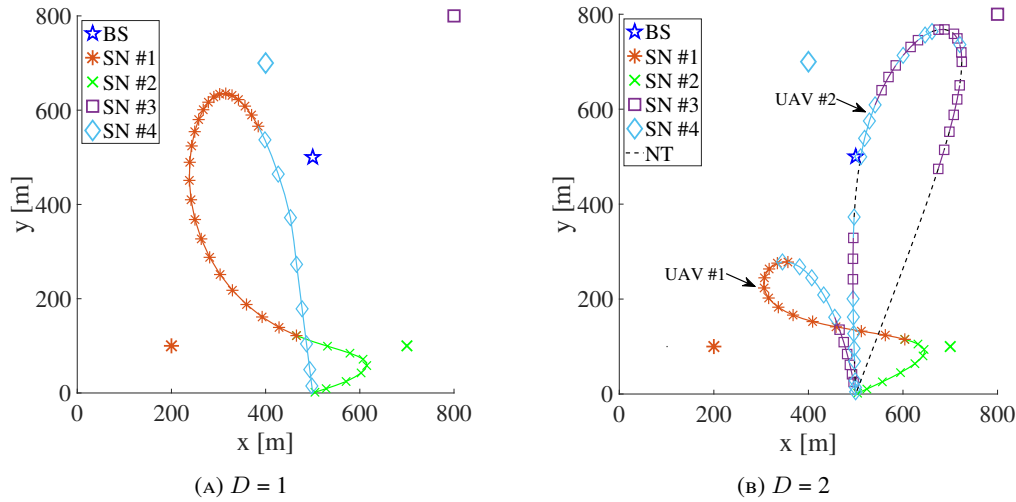


FIGURE 4.1: Trajectories followed by drones and association with SNs in the first scenario.

4.5 Numerical results

In this Section an assessment of the envisioned solution is proposed. To this end, four different scenarios are investigated:

1. The first aims at showing the relation between the trajectories and the scheduling plans obtained with different number of drones, in a simple context.
2. The second analyzes the trajectories and the total relayed data when mission time varies, in a more complex situation.
3. The third considers a larger number of UAVs and SNs with different transmission power levels.
4. The fourth demonstrates the effectiveness of the proposed hybrid approach with respect to an algorithm derived from [52].

The adopted channel model is the Rician fading one, characterized by Rician factor K_c . The CDF $F(\cdot)$ can be modeled as $F(u) = 1 - Q_m(\sqrt{2K_c}, \sqrt{2(K_c + 1)u})$, where $Q_m(\cdot, \cdot)$ is the Marcum Q-function [45]. As for the transmission, according to IEEE P802.16t, bandwidth $B_j = 10$ kHz $\forall j$, whereas $P_j^S = 10$ mW $\forall j$. Further, for each j , the generated sensed data is o_j

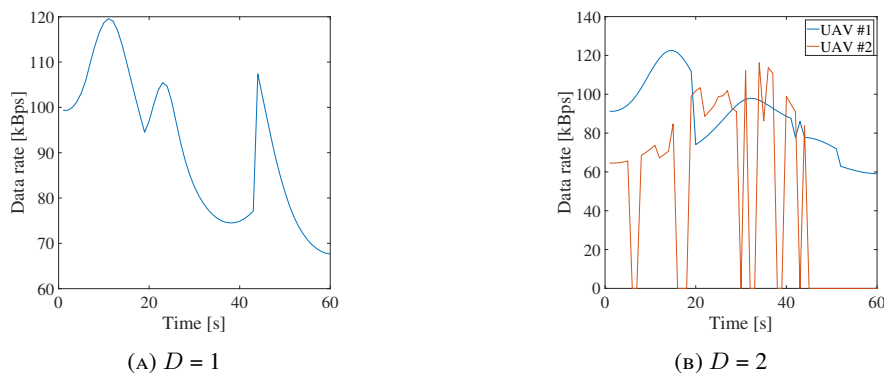


FIGURE 4.2: Data rate in the first scenario.

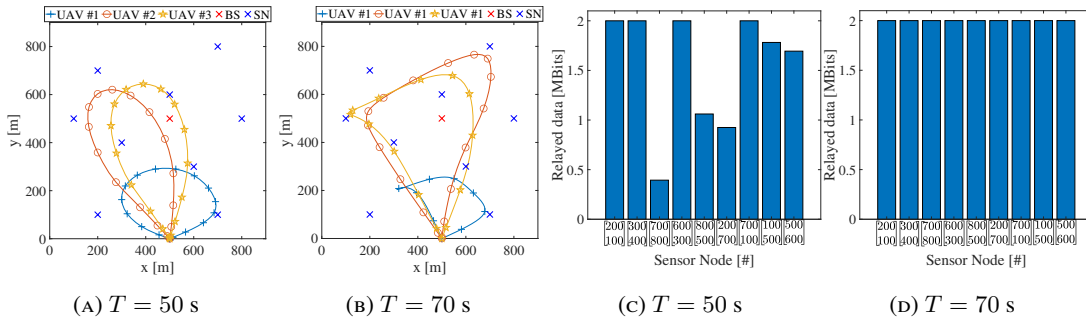


FIGURE 4.3: Trajectories and total relayed data in the second scenario.

= 2 Mbit. Drones take-off/land from/at $[500 \ 0]^T$. Similarly to [56], the remaining simulation parameters are summarized in Table 4.1. All the above are common to the configurations analyzed hereafter, if not otherwise specified.

The first considered scenario compares a single-drone setup with a 2-drones one. In particular, $S = 4$ SNs are randomly deployed as shown in Fig. 4.1 with a fixed mission duration time $T = 60$ s. As can be seen, in both configurations, drones approach SNs to maximize the amount of relayed data. Specifically, in the first setup the UAV starts serving SN located at $[700 \ 100]^T$ and then it proceeds towards the closer one, i.e., SN #1. This process repeats also for the node placed at $[400 \ 700]^T$. Similar considerations can be done for 2-drones configuration. In particular, in Fig. 4.1b it is shown that SNs #1 and #2 are exclusively associated to UAV #1, while SN #3 and SN #4 are cooperatively served by both drones.

A considerable difference between the two setups lies in the absence of NT state in Fig. 4.1a with respect to Fig. 4.1b. In fact, due to the lack of time, in T s, the single-drone setup is only able to partially relay data from the three served nodes, i.e., ~ 5.33 Mbits, while ignoring the farthest one. On the opposite, two drones are more than sufficient to fulfil the mission, thus implying NT timeslots during the mission in order to satisfy (4.8). This is further highlighted in Fig. 4.2b in which valleys are present differently from Fig. 4.2a where a continuous transmission is depicted. To provide further insights, a second more

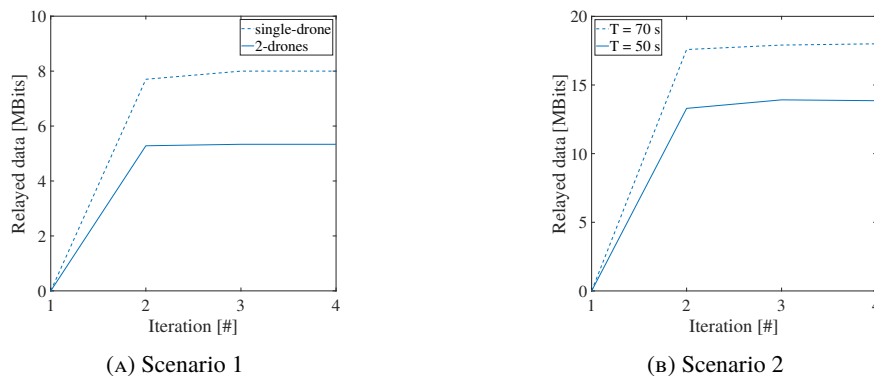


FIGURE 4.4: Convergence in the first two scenarios.

complex scenario is investigated hereby. In particular, $S = 9$ SNs are randomly deployed in the reference area. In this case, given a fixed number of drones $D = 3$, the mission duration is made varying. In the first setup $T = 50$ s, while in the second $T = 70$ s. In Figs. 4.3a and 4.3b, the trajectories adopted by the UAVs are shown. It clearly emerges that, in the first configuration, due to the lack of time, the swarm focuses on relaying signals from the regions closer to the starting point and with higher SNs density. This is further confirmed by Fig. 4.3c, where nodes located at $[700 \ 800]^T$, $[800 \ 500]^T$, and $[200 \ 700]^T$ result to be the most

penalized. Indeed, the total amount of transmitted data is ~ 13.9 Mbits. On the opposite, in the second configuration, the swarm has enough time to successfully complete the mission.

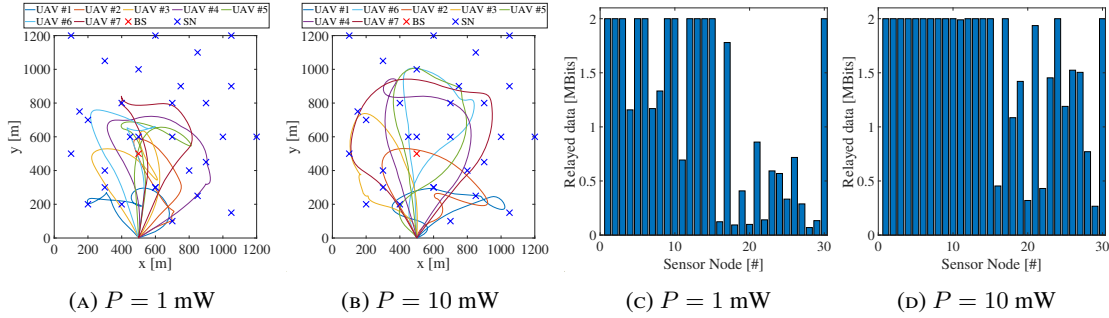


FIGURE 4.5: Trajectories and total relayed data in the third scenario.

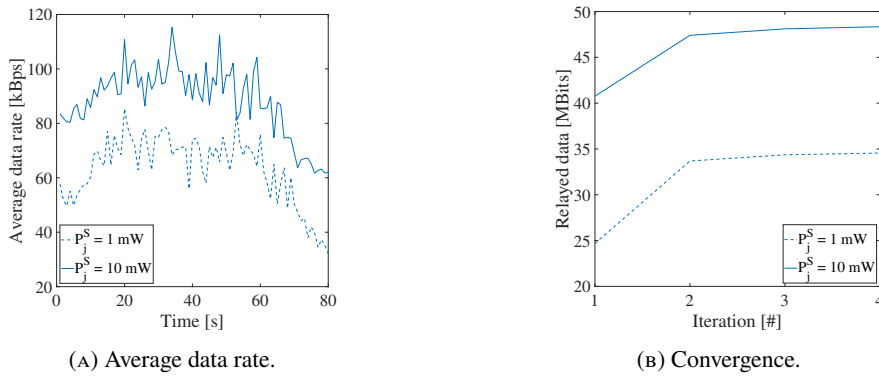


FIGURE 4.6: Average data rate of swarms and convergence in the third scenario.

Remarkably, in both scenarios, the proposed approach converges to a quasi-optimal solution (see Fig. 4.4) in 4 iterations. Besides, all solutions result to be feasible with respect to (4.15) because no overlap has been registered.

To demonstrate the applicability to larger simulations, a swarm of $D = 7$ drones and a set of $S = 30$ SNs are considered in the third scenario. Moreover, to provide parameter variability with respect to previous scenarios, two values of sensors' transmission power is probed, i.e., $P_j^S = \{1, 10\}$ mW, in a mission of $N = 80$ time intervals. As can be seen in Figs. 4.5a and 4.5b, the area covered by the swarm is wider when a higher transmission power is granted and, viceversa, it restricts with a lower P_j^S . This is due to the fact that is more convenient maximizing the data rate of nodes closer to the starting point when the transmission power is limited. Moreover, the amount of relayed data, for each sensor, is less when a lower P_j^S is considered, as highlighted in Figs. 4.5c and 4.5d. This is further confirmed by Fig. 4.6a that shows the average data rate of the swarm in the two configurations. Furthermore, as depicted in Fig. 4.6b the overall algorithm converges, thus granting a total amount of relayed data equal to ~ 34.55 Mbits, for $P_j^S = 1$ mW, and ~ 48.34 Mbits, for $P_j^S = 10$ mW. Note that, in both configurations, mission time and nodes' transmission power are not sufficient to completely relay the sensing data to the BS.

Lastly, in the fourth scenario, the proposed solution is compared with a benchmark approach derived from [52], in a mission with a set of $S = 12$ randomly deployed SNs. In particular, each drone of the swarm relays data of a SN group by hovering over multiple optimum locations, thus maximizing the data rate of each UAV-SN-BS link. The locations can be easily obtained by solving a problem similar to (P2), which is omitted due to space restrictions. Nonetheless, since $B_d \gg B_s$, these spots correspond, on x-y plane, to the SNs'

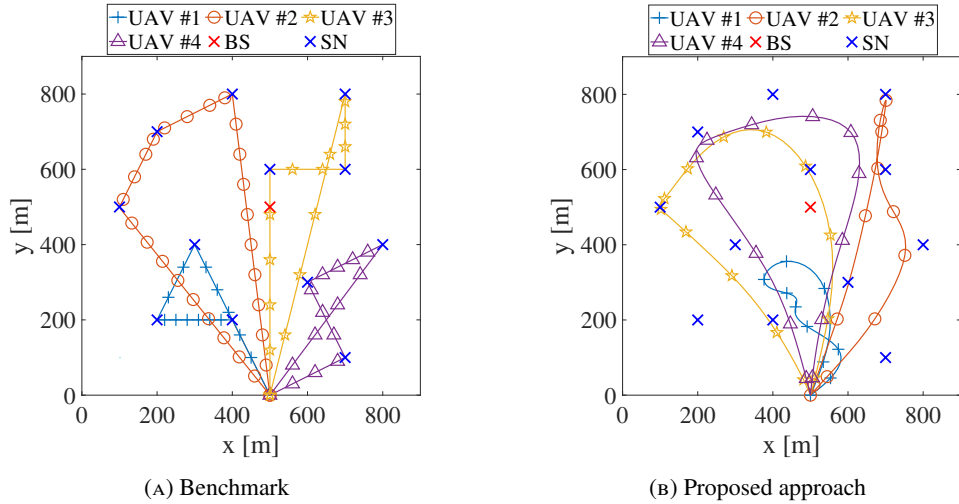


FIGURE 4.7: Trajectory plans comparison in the fourth scenario.

positions which minimize the distance, i.e., $d_{k,j} = H \forall k, j$. Moreover, each UAV flies at constant speed v_{MAX} among assigned locations adopting a rectilinear motion, moving from the current position to the closer one. Therefore, all nodes are equally divided among UAVs, as shown in Fig. 4.7a, where the related swarm path plan is depicted. Besides, trajectories obtained by the proposed hybrid approach are illustrated in Fig. 4.7b. It results that, the trajectory plan derived from the hybrid proposed approach allows drones to accomplish the mission by cooperatively relaying all the data in 70 s. On the contrary, when the benchmark algorithm is employed, the swarm completes the mission in 81 s. This leads to a performance gain of $\sim 14\%$. It is worth specifying that this result is a lower-bound in terms of performance: drones in benchmark approach fly at speed v_{MAX} without taking into consideration the acceleration limitations, as in the hybrid approach.

Chapter 5

Channel Gain Lower Bound for IRS-Assisted UAV-Aided Communications

The MIMO capabilities of an IRS combined with the inherent versatility of an UAV can significantly enhance the communication quality between a GU and a BS. This Chapter investigates a composite channel gain expression for performance analysis in IRS-assisted UAV-aided networks under Rician fading conditions. Differently from the present literature, the proposed formulation takes into account wave interference among surface elements and with the receiver. Moreover, Orthogonal Frequency Multiple Access (OFDMA) is employed to cope with spatial- and frequency-selective fading, enabling direct application in multi-user multi-drone scenarios. Numerical results demonstrate that the proposed model provides an underestimate, as expected from theoretical analysis, from which a lower bound for the data rate can be obtained, to be practically employed for system design and assessment.

5.1 Introduction

Recently, the scientific community considered the possibility to equip UAVs with IRSs, also known as RISs. IRSs are composed by PRUs that can reflect and shift the incident electromagnetic wave by a programmable phase, thus yielding signal beamforming when optimally set. The high mobility of drones allows to change the IRS location to obtain a better LoS link and a lower pathloss. Although this combination grants a huge improvement of the channel quality, it comes with new challenges [6], [7] and in particular with the necessity of a dedicated channel model. This motivated recent studies that started to investigate the achievable performance of IRS-assisted UAV-aided systems, however existing approaches still neglect some important aspects.

In the very recent work [57], a thorough non-convex optimization framework is given to jointly design UAV trajectory, IRS's phase shifts, scheduling, and resource allocation, for an OFDMA communication system. However, only the LoS component is considered and hence a simplified deterministic channel model. A more accurate modeling is performed in [58], where closed-form approximations of outage probability, average symbol error probability, and channel capacity are derived for RIS-aided wireless networks over Rician fading channels. However, in this case the presence of a direct, yet weak, transmitter-receiver link is not considered. To fill this gap, [59] models such a link as a Rayleigh Random Variable (RV), by introducing a closed-form upper bound for the ergodic capacity, and a tight approximation for the outage probability. Moreover, simplified expressions in the asymptotic regime are also provided in [58], [59], motivated by the fact that the resulting distribution cannot be recast as a known one. In this respect, a stochastic model has been recently proposed in [60] in which IRSs are deployed for wireless multi-hop backhauling. The obtained channel gain

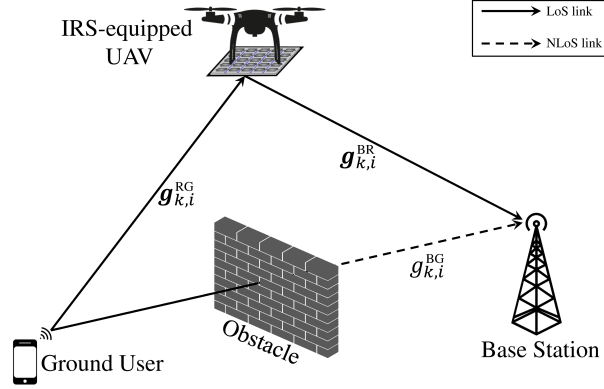


FIGURE 5.1: Reference scenario.

expression is approximated to a Rician distribution and then evaluated in terms of outage and symbol error probability. However, in these works signal components are added coherently, thus neglecting wave interference among surface elements. Moreover, differently from [57], OFDMA is not considered hence multi-user transmission is not supported.

Motivated by the discussion above, this work takes a step further towards the analytical development of a more comprehensive channel model for UAV-aided IRS-assisted OFDMA systems. In particular, it is envisioned a scenario in which a drone equipped with an IRS flies over the reference area to enhance the channel quality perceived by the BS, since the direct link between GU and BS may be too weak due to obstructions or unfavorable propagation, as depicted in Fig. 5.1. A stochastic channel model is derived, which takes into account interference between PRUs and GUs, and considers at the same time the presence of a weak LoS on the GU-BS link. The obtained model provides an underestimate of the actual gain thus yielding a lower bound data rate that can be practically employed for system design and assessment. Moreover, it can be directly applied in multi-user multi-drone scenarios thanks to the adoption of OFDMA.

5.2 System Model

A UAV is involved in a mission of duration T seconds, split in $k = 1, \dots, K$ intervals of δ_t seconds. As a consequence, also the trajectory is discretized into a sequence of K locations denoted by $\mathbf{q}_k = [x_k^u \ y_k^u \ z_k^u]^T \in \mathbb{R}^3$. Moreover, the UAV is equipped with an IRS composed by $M \times N$ PRUs. In each time interval k , each PRU of size $s \text{ m}^2$ is assumed to reflect the incident signal with an amplitude $a_{k,m,n} \in [0, 1]$ and a phase shift $\phi_{k,m,n} \in [0, 2\pi)$, $m = 1, \dots, M, n = 1, \dots, N$, i.e., $\Gamma_{k,m,n} = a_{k,m,n} e^{j\phi_{k,m,n}}$. The centers of row elements are equally separated by $d_r \text{ m}$, while column ones are spaced by $d_c \text{ m}$. During the mission, the drone has to serve a GU, located at $\mathbf{w} = [x^g \ y^g \ z^g]^T \in \mathbb{R}^3$, in order to enhance the communication channel with respect to a BS located at $\mathbf{q}^{\text{BS}} = [x^{\text{B}} \ y^{\text{B}} \ z^{\text{B}}]^T \in \mathbb{R}^3$. Therefore, the distance between the GU and the BS is time invariant and described by $d^{\text{BG}} = \|\mathbf{q}^{\text{BS}} - \mathbf{w}\|$. On the contrary, UAV-GU and BS-UAV distances are time dependent and described by $d_k^{\text{RG}} = \|\mathbf{q}_k - \mathbf{w}\|$ and $d_k^{\text{BR}} = \|\mathbf{q}^{\text{BS}} - \mathbf{q}_k\|$, respectively.

The antennas of GU, BS, and IRS are characterized by power radiation pattern functions (including antenna gains) $F^{\text{GU}}(\theta, \varphi)$, $F^{\text{BS}}(\theta, \varphi)$, and $F^{\text{IRS}}(\theta, \varphi)$, respectively. These functions define the variation of the antennas' transmitted/received power along a certain direction in elevation and azimuth angles θ and φ . OFDMA is adopted by the whole wireless communication system, to avoid frequency interference with other GUs in the reference area. Therefore, the total bandwidth B is divided into subcarriers of $\delta_f \text{ Hz}$ each. Without loss of generality,

the GU can transmit on I subcarriers during the whole mission. Besides, it is assumed that the Channel State Information (CSI) is perfectly known at the BS.

The channel gain between the GU and the BS, in k th timeslot and subcarrier $i = 1, \dots, I$ is

$$g_{k,i}^{\text{BG}} = \sqrt{\beta_0 d^{\text{BG}-\alpha} F^{\text{BG}}(\theta^{\text{BG}}, \varphi^{\text{BG}})} h_{k,i}^{\text{BG}}, \quad (5.1)$$

where β_0 denotes the channel power gain at the reference distance of 1 m, α is the pathloss exponent, $F^{\text{BG}}(\cdot, \cdot) = F^{\text{BS}}(\cdot, \cdot) F^{\text{GU}}(\cdot, \cdot)$, and $h_{k,i}^{\text{BG}} \sim \mathcal{CN}(\mu_{k,i}^{\text{BG}}, 2\sigma_{k,i}^{\text{BG}2})$ describes the channel coefficient. It is worth noting that the channel envelope $|h_{k,i}^{\text{BG}}|$ can be considered Rician [61], with K-factor $\kappa^{\text{BG}} = \frac{|\mu_{k,i}^{\text{BG}}|^2}{2\sigma_{k,i}^{\text{BG}2}}$ and average power $\Omega_{\text{BG}} = |\mu_{k,i}^{\text{BG}}|^2 + 2\sigma_{k,i}^{\text{BG}2} = 1$. Therefore, the channel coefficient can be modeled as

$$h_{k,i}^{\text{BG}} = \sqrt{\frac{\kappa^{\text{BG}}}{\kappa^{\text{BG}} + 1}} \bar{h}_{k,i}^{\text{BG}} + \sqrt{\frac{1}{\kappa^{\text{BG}} + 1}} \tilde{h}_{k,i}^{\text{BG}}. \quad (5.2)$$

In particular, $\bar{h}_{k,i}^{\text{BG}} = e^{-j2\pi i \delta_f \frac{d^{\text{BG}}}{c}}$ is the LoS component, $j = \sqrt{-1}$, characterized by a phase shift depending on the user's subcarrier index, while $\tilde{h}_{k,i}^{\text{BG}} \sim \mathcal{CN}(0, 1)$ is circular symmetric Gaussian distributed, describing small-scale fading.

Similarly, UAV-GU channel gain in timeslot k , subcarrier i , between GU and UAV, is modeled as follows:

$$\mathbf{g}_{k,i}^{\text{RG}} = \sqrt{\beta_0 d_k^{\text{RG}-\alpha} F^{\text{RG}}(\theta_k^{\text{RG}}, \varphi_k^{\text{RG}})} \mathbf{h}_{k,i}^{\text{RG}}, \quad (5.3)$$

$$\mathbf{h}_{k,i}^{\text{RG}} = \sqrt{\frac{\kappa^{\text{RG}}}{\kappa^{\text{RG}} + 1}} \bar{\mathbf{h}}_{k,i}^{\text{RG}} + \sqrt{\frac{1}{\kappa^{\text{RG}} + 1}} \tilde{\mathbf{h}}_{k,i}^{\text{RG}}, \quad (5.4)$$

$$\bar{\mathbf{h}}_{k,i}^{\text{RG}} = e^{-j2\pi i \delta_f \frac{d_k^{\text{RG}}}{c}} \mathbf{h}_{k,\text{LoS}}^{\text{RG}}, \quad (5.5)$$

and $F^{\text{RG}}(\cdot, \cdot) = F^{\text{IRS}}(\cdot, \cdot) F^{\text{GU}}(\cdot, \cdot)$. Moreover, $\mathbf{h}_{k,\text{LoS}}^{\text{RG}}$ denotes the far-field array response [57] defined as

$$\begin{aligned} \mathbf{h}_{k,\text{LoS}}^{\text{RG}} &\in \mathbb{C}^{MN \times 1} = \\ &\begin{bmatrix} 1 e^{-j2\pi f_c \frac{d_r \sin \theta_k^{\text{RG}} \cos \varphi_k^{\text{RG}}}{c}} \dots e^{-j2\pi f_c \frac{(M-1)d_r \sin \theta_k^{\text{RG}} \cos \varphi_k^{\text{RG}}}{c}} \end{bmatrix}^{\text{T}} \\ &\otimes \begin{bmatrix} 1 e^{-j2\pi f_c \frac{d_c \sin \theta_k^{\text{RG}} \sin \varphi_k^{\text{RG}}}{c}} \dots e^{-j2\pi f_c \frac{(N-1)d_c \sin \theta_k^{\text{RG}} \sin \varphi_k^{\text{RG}}}{c}} \end{bmatrix}^{\text{T}}, \end{aligned} \quad (5.6)$$

where $\tilde{\mathbf{h}}_{k,i}^{\text{RG}} \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_{MN})$ and \otimes denotes the Kronecker product. It is worth noting that, with respect to (5.2), one more phase shift affects the channel model which depends on the vertical and horizontal Angles of Arrival (AoA) of the signal, i.e., θ_k^{RG} and φ_k^{RG} . In particular, $\sin \theta_k^{\text{RG}} = \frac{z_k^{\text{U}} - z^{\text{G}}}{d_k^{\text{RG}}}$, $\sin \varphi_k^{\text{RG}} = \frac{x_k^{\text{U}} - x^{\text{G}}}{\sqrt{(x_k^{\text{U}} - x^{\text{G}})^2 + (y_k^{\text{U}} - y^{\text{G}})^2}}$, and $\cos \varphi_k^{\text{RG}} = \frac{y_k^{\text{G}} - y_k^{\text{U}}}{\sqrt{(x_k^{\text{U}} - x^{\text{G}})^2 + (y_k^{\text{U}} - y^{\text{G}})^2}}$. Finally, the channel gain of UAV-BS link, $\forall k$ and $\forall i$, is

$$\mathbf{g}_{k,i}^{\text{BR}} = \sqrt{\beta_0 s d_k^{\text{BR}-\alpha} F^{\text{BR}}(\theta_k^{\text{BR}}, \varphi_k^{\text{BR}})} \mathbf{h}_{k,i}^{\text{BR}}, \quad (5.7)$$

$$\mathbf{h}_{k,i}^{\text{BR}} = \sqrt{\frac{\kappa^{\text{BR}}}{\kappa^{\text{BR}} + 1}} \bar{\mathbf{h}}_{k,i}^{\text{BR}} + \sqrt{\frac{1}{\kappa^{\text{BR}} + 1}} \tilde{\mathbf{h}}_{k,i}^{\text{BR}}, \quad (5.8)$$

$$\bar{\mathbf{h}}_{k,i}^{\text{BR}} = e^{-j2\pi i \delta_f \frac{d_k^{\text{BR}}}{c}} \mathbf{h}_{k,\text{LoS}}^{\text{BR}}, \quad (5.9)$$

$F^{\text{BR}}(\cdot, \cdot) = F^{\text{BS}}(\cdot, \cdot)F^{\text{IRS}}(\cdot, \cdot)$, while θ_k^{RG} and φ_k^{RG} are the vertical and horizontal Angles of Departure (AoD), respectively, so that $\sin \theta_k^{\text{BR}} = \frac{z_k^{\text{U}} - z^{\text{B}}}{d_k^{\text{BR}}}$, $\sin \varphi_k^{\text{BR}} = \frac{x_k^{\text{U}} - x^{\text{B}}}{\sqrt{(x_k^{\text{U}} - x^{\text{B}})^2 + (y_k^{\text{U}} - y^{\text{B}})^2}}$, and $\cos \varphi_k^{\text{BR}} = \frac{y^{\text{B}} - y_k^{\text{U}}}{\sqrt{(x_k^{\text{U}} - x^{\text{B}})^2 + (y_k^{\text{U}} - y^{\text{B}})^2}}$. Finally, similarly to (5.6),

$$\begin{aligned} \mathbf{h}_{k,\text{LoS}}^{\text{BR}} \in \mathbb{C}^{MN \times 1} = & \\ & \begin{bmatrix} 1 e^{-j2\pi f_c \frac{d_r \sin \theta_k^{\text{BR}} \cos \varphi_k^{\text{BR}}}{c}} \dots e^{-j2\pi f_c \frac{(M-1)d_r \sin \theta_k^{\text{BR}} \cos \varphi_k^{\text{BR}}}{c}} \end{bmatrix}^{\text{T}} \\ & \otimes \begin{bmatrix} 1 e^{-j2\pi f_c \frac{d_c \sin \theta_k^{\text{BR}} \sin \varphi_k^{\text{BR}}}{c}} \dots e^{-j2\pi f_c \frac{(N-1)d_c \sin \theta_k^{\text{BR}} \sin \varphi_k^{\text{BR}}}{c}} \end{bmatrix}^{\text{T}}, \end{aligned}$$

and $\tilde{\mathbf{h}}_{k,i}^{\text{BR}} \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_{MN})$. The IRS phase shift matrix $\Phi_k \in \mathbb{C}^{MN \times MN}$, for each timeslot k , is defined as

$$\Phi_k = \text{diag} \left(a_{k,1,1} e^{j\phi_{k,1,1}}, \dots, a_{k,1,N} e^{j\phi_{k,1,N}}, \dots, \right. \\ \left. a_{k,M,1} e^{j\phi_{k,M,1}}, \dots, a_{k,M,N} e^{j\phi_{k,M,N}} \right), \quad (5.10)$$

and, hence, the composite IRS UAV-assisted channel gain between GU and BS in subcarrier i is

$$G_{k,i} = \mathbf{g}_{k,i}^{\text{BR} \text{ T}} \Phi_k \mathbf{g}_{k,i}^{\text{RG}} + g_{k,i}^{\text{BG}}. \quad (5.11)$$

Recalling Shannon's equation, the channel capacity in timeslot k and subcarrier i is thus given by

$$C_{k,i} = \delta_f \log_2 \left(1 + \frac{P_{k,i} |G_{k,i}|^2}{\rho^2} \right), \quad (5.12)$$

where $P_{k,i}$ is the transmit power of GUs in i -th subcarrier and $\rho^2 = N_0 \delta_f$ is the noise power, whereas N_0 denotes the spectral noise power. Given a maximum achievable data rate $R_{k,i}$, to guarantee a reliable communication, it is required that the outage probability $p_{k,i}$ remains below a threshold ε , i.e.

$$\begin{aligned} p_{k,i} = \mathbb{P}(C_{k,i} < R_{k,i}) &= \mathbb{P} \left(|G_{k,i}|^2 < \frac{\rho^2 (2^{\frac{R_{k,i}}{\delta_f}} - 1)}{P_{k,i}} \right) \\ &= F_{|G_{k,i}|^2} \left(\frac{\rho^2 (2^{\frac{R_{k,i}}{\delta_f}} - 1)}{P_{k,i}} \right) \leq \varepsilon, \quad \forall \quad \begin{matrix} k = 1, \dots, K, \\ i = 1, \dots, I, \end{matrix} \end{aligned}$$

with $F_{|G_{k,i}|^2}$ denoting the CDF of $|G_{k,i}|^2$. Therefore, considering the maximum tolerable outage, i.e., $p_{k,i} = \varepsilon$, the data rate is obtained as

$$R_{k,i} = \delta_f \log_2 \left(1 + \underbrace{\frac{P_{k,i} F_{|G_{k,i}|^2}^{-1}(\varepsilon)}{\rho^2}}_{\text{Signal-to-Noise Ratio (SNR)}} \right). \quad (5.13)$$

Unfortunately, a closed-form expression of $F|_{G_{k,i}|^2}$ is difficult to calculate. Indeed, $G_{k,i}$ involves the pairwise product of two complex Gaussian RVs, whose distribution, named complex double Gaussian, is given in terms of an infinite sum of modified Bessel functions [62], hence intractable for the sake of the present analysis. Besides, dealing with the square module of the summation in (5.11) is even more challenging. Therefore, an approximation is useful to handle with such a scenario.

5.3 Channel Modeling Approximation

In this Section an approximation of the composite channel model is provided. Fixed (m, n) , i , and k , let $h_{k,i,m,n}^{\text{BR}} \sim \mathcal{CN}(\mu_{k,i,m,n}^{\text{BR}}, 2\sigma_{k,i,m,n}^{\text{BR}2})$ and $h_{k,i,m,n}^{\text{RG}} \sim \mathcal{CN}(\mu_{k,i,m,n}^{\text{RG}}, 2\sigma_{k,i,m,n}^{\text{RG}2})$ be two generic channel gains related to BS-PRU-GU link $h_{k,i,m,n}^{\text{BRG}}$ where, similarly to (5.2) and related definitions, the following relationship are obtained:

$$\begin{aligned} \mu_{k,i,m,n}^{\text{BR}} &= \sqrt{\frac{\kappa^{\text{BR}}}{\kappa^{\text{BR}} + 1}} e^{-j2\pi i \delta_f \frac{d_k^{\text{BR}}}{c}} \\ &\times e^{-j2\pi f_c \frac{(m-1)d_r \sin \theta_k^{\text{BR}} \cos \varphi_k^{\text{BR}} + (n-1)d_c \sin \theta_k^{\text{BR}} \sin \varphi_k^{\text{BR}}}{c}}, \end{aligned} \quad (5.14)$$

$$\begin{aligned} \mu_{k,i,m,n}^{\text{RG}} &= \sqrt{\frac{\kappa^{\text{RG}}}{\kappa^{\text{RG}} + 1}} e^{-j2\pi i \delta_f \frac{d_k^{\text{RG}}}{c}} \\ &\times e^{-j2\pi f_c \frac{(m-1)d_r \sin \theta_k^{\text{RG}} \cos \varphi_k^{\text{RG}} + (n-1)d_c \sin \theta_k^{\text{RG}} \sin \varphi_k^{\text{RG}}}{c}}, \end{aligned} \quad (5.15)$$

$$2\sigma^{\text{BR}2} = \frac{1}{\kappa^{\text{BR}} + 1}, \quad 2\sigma^{\text{RG}2} = \frac{1}{\kappa^{\text{RG}} + 1}. \quad (5.16)$$

The product $h_{k,i,m,n}^{\text{BR}} h_{k,i,m,n}^{\text{RG}}$ is distributed as a complex double Gaussian that however, as mentioned, is difficult to handle. The following Lemma provides an approximation that is useful for the subsequent development.

Lemma 1. *Given (m, n) , i , and k , the product $h_{k,i,m,n}^{\text{BR}} h_{k,i,m,n}^{\text{RG}}$ can be approximated by a complex Gaussian variable $Z_{k,i,m,n} \sim \mathcal{CN}(\mu_{k,i,m,n}^Z, 2\sigma_{k,i,m,n}^{Z2})$ with $\mu_{k,i,m,n}^Z = \mu_{k,i,m,n}^{\text{BR}} \mu_{k,i,m,n}^{\text{RG}}$ and $2\sigma_{k,i,m,n}^{Z2} = 2\sigma_{k,i,m,n}^{\text{RG}2} |\mu_{k,i,m,n}^{\text{BR}}|^2 + 2\sigma_{k,i,m,n}^{\text{BR}2} |\mu_{k,i,m,n}^{\text{RG}}|^2$.*

Proof. For brevity, let $h_{\text{BR}} \triangleq h_{k,i,m,n}^{\text{BR}}$ and $h_{\text{RG}} \triangleq h_{k,i,m,n}^{\text{RG}}$. The product is defined as $Z \triangleq h_{\text{BR}} h_{\text{RG}} = Z^{\text{R}} + jZ^{\text{I}}$, where the real and imaginary parts are $Z^{\text{R}} = h_{\text{BR}}^{\text{R}} h_{\text{RG}}^{\text{R}} - h_{\text{BR}}^{\text{I}} h_{\text{RG}}^{\text{I}}$ and $Z^{\text{I}} = h_{\text{BR}}^{\text{R}} h_{\text{RG}}^{\text{I}} + h_{\text{BR}}^{\text{I}} h_{\text{RG}}^{\text{R}}$.

In particular, $\{h_{\text{BR}}^{\text{R}}, h_{\text{RG}}^{\text{R}}, h_{\text{BR}}^{\text{I}}, h_{\text{RG}}^{\text{I}}\} \sim \mathcal{N}\left(\left\{m_{\text{BR}}^{\text{R}}, m_{\text{RG}}^{\text{R}}, m_{\text{BR}}^{\text{I}}, m_{\text{RG}}^{\text{I}}\right\}, \left\{\sigma_{\text{BR}}^2, \sigma_{\text{RG}}^2, \sigma_{\text{BR}}^2, \sigma_{\text{RG}}^2\right\}\right)$. The product of two independent Gaussian RVs $X \sim \mathcal{N}(m_X, \sigma_X^2)$ and $Y \sim \mathcal{N}(m_Y, \sigma_Y^2)$ has the following statistics [63]

$$\begin{aligned} \mathbb{E}[XY] &= m_X m_Y, \\ \mathbb{E}[(XY - \mathbb{E}[XY])^2] &= m_Y^2 \sigma_X^2 + (m_X^2 + \sigma_X^2) \sigma_Y^2 \\ &= (1 + \delta_X^2 + \delta_Y^2) \sigma_X^2 \sigma_Y^2, \end{aligned}$$

where $\delta_X = \frac{m_X}{\sigma_X}$ and $\delta_Y = \frac{m_Y}{\sigma_Y}$. The distribution of XY tends to $\mathcal{N}(m_X m_Y, m_X^2 \sigma_Y^2 + m_Y^2 \sigma_X^2)$ for $\{\delta_X, \delta_Y\} \gg 1$, hence the following inequality holds

$$\mathbb{E}[(XY - \mathbb{E}[XY])^2] > m_X^2 \sigma_Y^2 + m_Y^2 \sigma_X^2. \quad (5.17)$$

The statistics of Z^R and Z^I are:

$$\begin{aligned} m_Z^R &= m_{\text{BR}}^R m_{\text{RG}}^R - m_{\text{BR}}^I m_{\text{RG}}^I, & m_Z^I &= m_{\text{BR}}^R m_{\text{RG}}^I + m_{\text{BR}}^I m_{\text{RG}}^R, \\ \sigma_Z^R{}^2 &= \sigma_Z^I{}^2 = \sigma_{\text{RG}}^2 m_{\text{BR}}^R{}^2 + \sigma_{\text{BR}}^2 m_{\text{RG}}^R{}^2 + \sigma_{\text{RG}}^2 m_{\text{BR}}^I{}^2 + \sigma_{\text{BR}}^2 m_{\text{RG}}^I{}^2 \triangleq \sigma_Z^2. \end{aligned}$$

Finally, the first and second central moments of $Z \sim \mathcal{CN}(\mu_Z, 2\sigma_Z^2)$ are

$$\begin{aligned} \mu_Z &= m_Z^R + jm_Z^I = (m_{\text{BR}}^R + jm_{\text{BR}}^I)(m_{\text{RG}}^R + jm_{\text{RG}}^I) = \mu_{\text{BR}}\mu_{\text{RG}} \\ 2\sigma_Z^2 &= \sigma_Z^R{}^2 + \sigma_Z^I{}^2 = 2\sigma_{\text{RG}}^2 |\mu_{\text{BR}}|^2 + 2\sigma_{\text{BR}}^2 |\mu_{\text{RG}}|^2, \end{aligned}$$

hence the thesis is proven. \blacksquare

Remark 1. From (5.17), it follows that the Probability Density Function (PDF) of the provided approximation $Z_{k,i,m,n}$ has lower variance than the actual (complex double Gaussian) PDF of $h_{k,i,m,n}^{\text{BR}} h_{k,i,m,n}^{\text{RG}}$. Moreover, considering also that the latter is positively-skewed [63, Eq. (33)], intuitively, it can be expected that the tail of the approximated PDF decays faster than the actual one. This in turn means that the Inverse CDF of $Z_{k,i,m,n}$ is expected to provide a lower bound, for sufficiently high probability values. Such an intuition will be confirmed by the numerical results in Section 5.4.

From Lemma 1, it results that $h_{k,i,m,n}^{\text{BRG}} = Z_{k,i,m,n} \Gamma_{k,m,n} \sim \mathcal{CN}(\mu_{k,i,m,n}^{\text{BRG}}, 2\sigma_{k,i,m,n}^{\text{BRG}2})$ where, to ease the notation, (5.14)-(5.15) have been rewritten as $\mu_{k,i,m,n}^{\text{BR}} = \sqrt{\frac{\kappa_{k,i,m,n}^{\text{BR}}}{\kappa_{k,i,m,n}^{\text{BR}}+1}} e^{j\psi_{k,i,m,n}^{\text{BR}}}$ and $\mu_{k,i,m,n}^{\text{RG}} = \sqrt{\frac{\kappa_{k,i,m,n}^{\text{RG}}}{\kappa_{k,i,m,n}^{\text{RG}}+1}} e^{j\psi_{k,i,m,n}^{\text{RG}}}$, with obvious definition of the phase terms. Hence, the envelope $|h_{k,i,m,n}^{\text{BRG}}|$ can be considered a Rician RV, leading to the model of the same type as in (5.2), (5.4) or (5.8), i.e.

$$h_{k,i,m,n}^{\text{BRG}} = a_{k,m,n} \left(\sqrt{\bar{\kappa}^{\text{BRG}}} e^{j\Psi_{k,i,m,n}} + \sqrt{\tilde{\kappa}^{\text{BRG}}} \tilde{h}^{\text{BRG}} \right), \quad (5.18)$$

where $\tilde{h}^{\text{BRG}} \sim \mathcal{CN}(0, 1)$ and, recalling that $\Gamma_{k,m,n} = a_{k,m,n} e^{j\phi_{k,m,n}}$ with amplitude $a_{k,m,n}$ and phase shift $\phi_{k,m,n}$,

$$\bar{\kappa}^{\text{BRG}} = \frac{\kappa_{k,i,m,n}^{\text{BR}} \kappa_{k,i,m,n}^{\text{RG}}}{(\kappa_{k,i,m,n}^{\text{BR}} + 1)(\kappa_{k,i,m,n}^{\text{RG}} + 1)}, \quad \tilde{\kappa}^{\text{BRG}} = \frac{\kappa_{k,i,m,n}^{\text{BR}} + \kappa_{k,i,m,n}^{\text{RG}}}{(\kappa_{k,i,m,n}^{\text{BR}} + 1)(\kappa_{k,i,m,n}^{\text{RG}} + 1)},$$

with $\Psi_{k,i,m,n} = \phi_{k,m,n} + \psi_{k,i,m,n}^{\text{BR}} + \psi_{k,i,m,n}^{\text{RG}}$. Notice that phase terms are irrelevant to the scatter component [57].

Once the expression of generic PRU-related cascaded channel has been obtained, it is necessary to derive a tractable expression for the UAV-aided IRS-assisted channel model.

Theorem 2. Given k and i , the channel envelope $|G_{k,i}|$ of the whole system has approximately a Rician distribution characterized by K -factor $\kappa = \frac{\nu_{G_{k,i}}^2}{2\sigma_{G_{k,i}}^2}$ and average power $\Omega = \nu_{G_{k,i}}^2 + 2\sigma_{G_{k,i}}^2$, with $\nu_{G_{k,i}}^2$ defined in (5.20) and $2\sigma_{G_{k,i}}^2 = \eta_k^2 \tilde{\kappa}^{\text{BRG}} \sum_{z=1}^{MN} a_{k,z}^2 + \lambda^2 \tilde{\kappa}^{\text{BG}}$.

Proof. By using the BS-GU link in (5.1) and combining the expressions in Section 5.2 with the result (5.18) derived in Lemma 1, (5.11) can be rewritten as

$$G_{k,i} = \eta_k \sum_{m=1}^M \sum_{n=1}^N h_{k,i,m,n}^{\text{BRG}} + \lambda h_{k,i}^{\text{BG}}, \quad (5.19)$$

where $\eta_k = \beta_0 \sqrt{(d_k^{\text{BR}} d_k^{\text{RG}})^{-\alpha} s F^{\text{BRG}}(\theta_k^{\text{BR}}, \varphi_k^{\text{BR}}, \theta_k^{\text{RG}}, \varphi_k^{\text{RG}})}$, $\lambda = \sqrt{\beta_0 d^{\text{BG}-\alpha} F^{\text{BG}}(\theta^{\text{BG}}, \varphi^{\text{BG}})}$, and $F^{\text{BRG}}(\cdot, \cdot, \cdot, \cdot) = F^{\text{BR}}(\cdot, \cdot) F^{\text{RG}}(\cdot, \cdot)$. Since $G_{k,i} = G_{k,i}^{\text{R}} + j G_{k,i}^{\text{I}}$, where $\{G_{k,i}^{\text{R}}, G_{k,i}^{\text{I}}\} \sim \mathcal{N}(\{\mu_{G_{k,i}}^{\text{R}}, \mu_{G_{k,i}}^{\text{I}}\}, \{\sigma_{G_{k,i}}^2, \sigma_{G_{k,i}}^2\})$, it results

$$\begin{aligned}\mu_{G_{k,i}}^{\text{R}} &= \eta_k \sqrt{\bar{\kappa}^{\text{BRG}}} \sum_{m=1}^M \sum_{n=1}^N a_{k,m,n} \cos \Psi_{k,i,m,n} + \lambda \sqrt{\bar{\kappa}^{\text{BG}}} \cos \psi_i^{\text{BG}}, \\ \mu_{G_{k,i}}^{\text{I}} &= \eta_k \sqrt{\bar{\kappa}^{\text{BRG}}} \sum_{m=1}^M \sum_{n=1}^N a_{k,m,n} \sin \Psi_{k,i,m,n} + \lambda \sqrt{\bar{\kappa}^{\text{BG}}} \sin \psi_i^{\text{BG}},\end{aligned}$$

where $\bar{\kappa}^{\text{BG}} = \frac{\kappa^{\text{BG}}}{\kappa^{\text{BG}}+1}$ and $\psi_i^{\text{BG}} = -2\pi i \delta_f \frac{d^{\text{BG}}}{c}$. For all m and n , define $\{z, z'\} = (m, n)$ as the linearized indexes of IRS elements. The corresponding squared LoS component $\nu_{G_{k,i}}^2 = \mu_{G_{k,i}}^{\text{R}}{}^2 + \mu_{G_{k,i}}^{\text{I}}{}^2$ is

$$\begin{aligned}\nu_{G_{k,i}}^2 &= \eta_k^2 \bar{\kappa}^{\text{BRG}} \left(2 \sum_{z > z'}^{MN} a_{k,z} a_{k,z'} \cos(\Psi_z - \Psi_{z'}) + \sum_{z=1}^{MN} a_{k,z}^2 \right) \\ &+ \lambda \sqrt{\bar{\kappa}^{\text{BG}}} \left(2 \eta_k \sqrt{\bar{\kappa}^{\text{BRG}}} \sum_{z=1}^{MN} a_{k,z} \cos(\psi_i^{\text{BG}} - \Psi_z) + \lambda \sqrt{\bar{\kappa}^{\text{BG}}} \right).\end{aligned}\quad (5.20)$$

Moreover, the Non-Line of Sight (NLoS) component is $2\sigma_{G_{k,i}}^2 = \eta_k^2 \bar{\kappa}^{\text{BRG}} \sum_{z=1}^{MN} a_{k,z}^2 + \lambda^2 \bar{\kappa}^{\text{BG}}$, i.e., the sum of all scatter components involved in the system. ■

Corollary 1. *The effective data rate (5.13) can be directly derived by computing the fading-power $F_{|G_{k,i}|^2}^{-1}$, which can be easily calculated as in [45].*

Corollary 2. *The obtained channel gain can be specialized for the case of Rayleigh fading by considering in (5.20) a completely obstructed BS-GU link, i.e., $\bar{\kappa}^{\text{BG}} \rightarrow 0$.*

Remark 2. *Following the lead of [60, Sec. III.C], it is not difficult to show that $|G_{k,i}|^2$ scales as $(M \times N)^2$. Thus, increasing the size of the IRSs greatly improves the fading-power, and hence SNR, so significantly lowering the outage probability, as shown in the next Section.*

It is worth noting that (5.20), which can be easily extended to scenarios in which multiple IRSs are present, exhibits two wave interference patterns: the former related to constructive/destructive superposition of the received waves among PRUs, the latter related to the coupling between each IRS element and the GU. Note also that (5.20) remains valid also for discrete phase shifts. In that case, however, it cannot be guaranteed that (5.20) is maximized, since cosine patterns may not be equal to 1. This leads to a performance degradation of the system that strictly depends on the phase quantization level. Notice that a similar effect may also arise from other causes, e.g. jitter. Still, the system is robust against the resulting beam-pointing degradation, since the elevation of the UAV yields a favorable ground illumination footprint. The accuracy of the provided approximation in Theorem 2 indirectly depends on two parameters δ_X and δ_Y involved in the approximation of $Z_{k,i,m,n}$, which are the reciprocals of the coefficient of variation for the variables X and Y defined in the proof of Lemma 1. This translates to a gap in $F_{|G_{k,i}|^2}^{-1}$, which reduces when the link is dominated by the LoS component, i.e., $\{\kappa^{\text{BR}}, \kappa^{\text{RG}}\} \gg 1$. This has been proved to be the case for A2G, and thus UAV-aided, communications [57], hence the ultimately provided lower bound on the outage probability is tight in such a regime, typical of UAV-assisted scenarios, as shown below by means of numerical simulations.

5.4 Numerical Results and Discussion

The obtained model is mainly affected by the error introduced by approximating the product of two complex Normal RV as a complex Normal instead of a complex double Gaussian, as discussed in Section 5.3. To validate its accuracy, three parameter settings are considered, i.e., $\delta_X = \delta_Y = \{1, 2, 3\}$. Results for 10^6 realizations are depicted in Fig. 5.2. It clearly

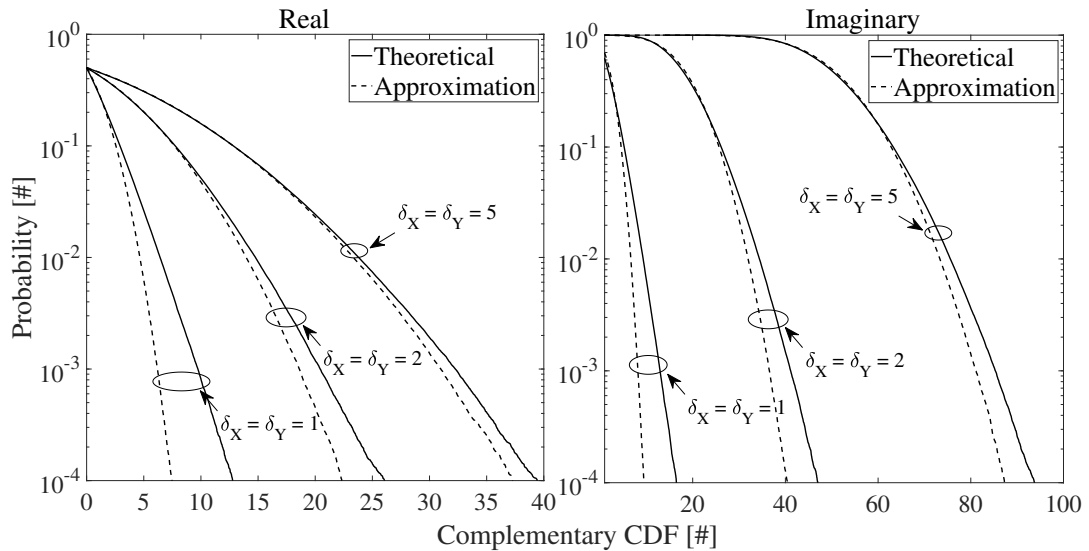


FIGURE 5.2: Comparison between approximation and actual Complementary CDF for different values of δ_X and δ_Y .

emerges that for higher values of $\{\delta_X, \delta_Y\}$ the distance between theoretical and approximation curves reduces. Specifically, the approximation underestimates the actual value in both real and imaginary parts, thus confirming Remark 1. Consequently, depending on $\{\delta_X, \delta_Y\}$, the approximation errors accumulated in (5.19) will translate into a certain underestimation error in $F_{|G_{k,i}|^2}^{-1}$. From a physical meaning perspective, the values of $\{\delta_X, \delta_Y\}$ are proportional to the power of the LoS, which in turn is related to the Rician K-factors. Given that in practical UAV-assisted communications the latter are between 5 dB and 15 dB [45], the provided approximation remains satisfactory also for very small probability.

Numerical results, based on 5×10^5 realizations, are now provided to get insights about the proposed model. Without loss of generality, consider a fixed instant k in which the optimal phase shift matrix Φ_k is applied such that unitary cosine patterns in (5.20) are obtained. According to [45], in all configurations the BS-GU link has a weaker LoS component, i.e., $\kappa^{\text{BG}} = 6$ dB, than BS-UAV and UAV-GU links, for which $\kappa^{\text{BR}} = \{10, 12\}$ dB and $\kappa^{\text{RG}} = \{12, 15\}$ dB. Several numbers of PRUs of the IRS are considered $M \times N = \{32, 64, 128, 256\}$ and the coefficients $\{\eta_k, \lambda\}$ are normalized to unity as in [60], thus not depend on the position-related parameters, which are hence left unspecified.

Fig. 5.3 confirms that the proposed channel gain model leads to a lower bound on the outage probability. This result, following Corollary 1, can be practically used to determine a conservative, but fairly tight data rate level for system provisioning and optimization. In particular, it can be noticed that the distance between exact (solid) and approximate (dashed) fading-power curves is lower when the K-factors increase, for all values of the number of PRUs $M \times N$. From a physical meaning perspective, this reflects the fact that the BS-UAV-GU link is mostly in LoS condition. Thus, as discussed, for typical Rician factors values the approximation of the double Gaussian distribution is quite accurate so yielding, in turn, a good accuracy in the fading-power approximation. To better highlight this fact, Fig. 5.4 shows the relative error between the exact and approximate curves depicted in Fig. 5.3, for

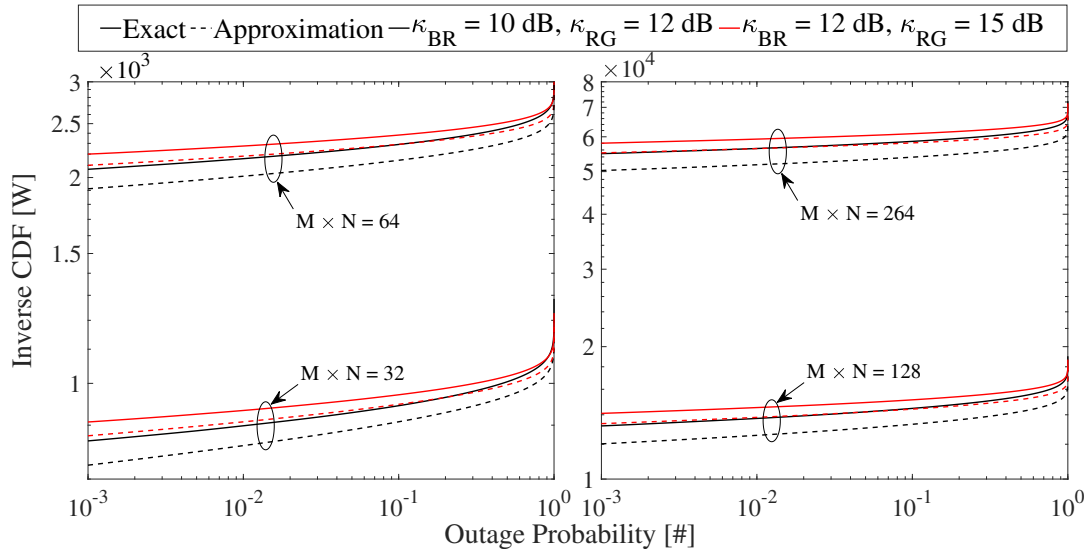


FIGURE 5.3: $F_{|G_{k,i}|^2}^{-1}$ for different K-factors and number of PRUs.

different values of the outage probability, i.e., $\varepsilon = \{10^{-1}, 10^{-2}, 10^{-3}\}$. It can be observed that the relative error decreases with the number of PRUs, which clearly indicates that a larger RIS provides better signal reflection and, hence, is beneficial to the ultimate performance. For instance, $M \times N = 128$ elements already achieve a remarkable 8.5% approximation error, for $\kappa^{\text{BR}} = 10$ dB, $\kappa^{\text{RG}} = 12$ dB, and $\varepsilon = 10^{-2}$.

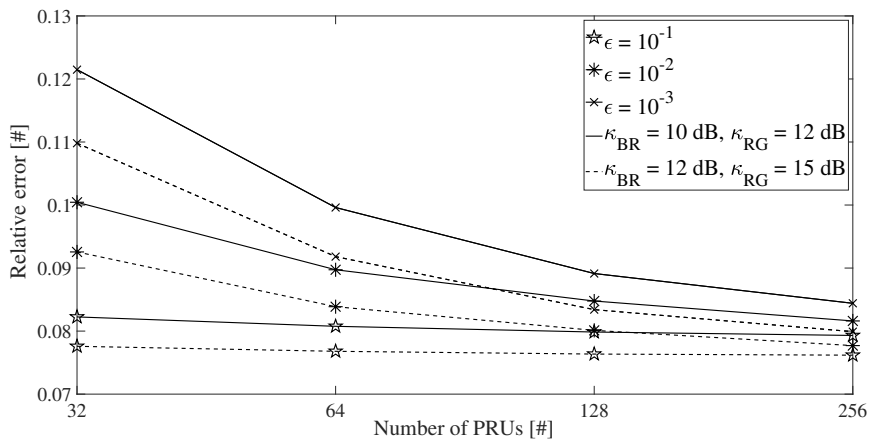


FIGURE 5.4: Relative errors between exact and approximate curves with K-factors, outage probabilities, and number of PRUs.

Chapter 6

Multi-UAV IRS-assisted Communications: Multi-User Channel Modeling and Fair Sum-Rate Optimization via Deep Reinforcement Learning

Unmanned Aerial Vehicles (UAVs) combined with Intelligent Reflective Surfaces (IRSs) represent a cutting-edge technology for improving the channel capacity of wireless communications, by capitalizing on UAVs' 3D mobility coupled with the IRSs' smart radio capabilities. This Chapter investigates a scenario in which a swarm of UAVs equipped with IRSs serves multiple Ground Users (GUs) concurrently transmitting to a single Base Station (BS) via OFDMA. The huge number of passive elements composing the IRSs introduces a significant complexity in the mission design. Therefore, each IRS is divided into patches that can be simultaneously used to serve different nodes. Considering general Rician fading, a comprehensive channel model for IRS-assisted UAV-aided networks is derived. Then, a multi-objective mixed-integer non-linear programming problem is conceived to maximize the sum-rate of the GUs and, at the same time, minimize the difference among the users' data rates, by jointly optimizing the trajectories and the phase shift matrices. This non-convex problem, reformulated in terms of scheduling (i.e., patch-GU assignment), is challenging to solve. Hence, it is rearranged as a Markov Decision Process and a quasi-optimal solution is obtained via Deep Reinforcement Learning. Extensive simulation analysis is performed to validate the results and the accuracy of the proposed model.

6.1 Introduction

As said in the previous Chapter, the possibility to equip UAVs with IRSs is attracting interest. Drones mobility adds more degrees of freedom that can be exploited to further improve the channel quality. Indeed, the high mobility of drones yields a better LoS link and a lower pathloss, due to the possibility to adjust the IRS location. Moreover, IRSs are characterized by a limited Size, Weight and Power consumption (SWaP) with respect to common phased-array antennas [64], which allows to prolong the mission duration, while still providing broad coverage. However, this comes with new challenges [6], [7] and in particular with the necessity of an accurate, general-purpose, and flexible channel model. Indeed, the presence of a BS with multiple IRSs and GUs, leads to interference patterns that need to be taken into account in case of system design and assessment.

This motivated recent studies that started to investigate the achievable performance of IRS-assisted UAV-aided systems. In [57] a non-convex optimization problem is formulated

for the joint design of UAV trajectory, IRS's phase shifts, scheduling, and resource allocation. The system employs OFDMA, which introduces frequency and spatial selectivity in the fading of the resulting channel. However, as the majority of the present literature, it considers only the LoS component and hence a deterministic channel which indeed is not able to capture the full characteristics of an actual wireless channel. In this regard, the literature on IRS has investigated different directions to obtain more realistic channel models. In [58] a formulation is proposed for IRS-aided wireless networks over Rician fading channels, which enables closed-form approximations of outage probability, average symbol error probability, and channel capacity. However, in this contribution the presence of a direct, yet weak, transmitter-receiver link is often not considered. To fill this gap, [59] introduces a Rician model to account for the LoS, which yields a closed-form upper bound for the ergodic capacity, and a tight approximation for the outage probability. However, the resulting distribution cannot be recast as a known one, which limits the practical tractability. To partially circumvent this issue, simplified expressions are provided in [58], [59], but only for the asymptotic regime.

More tractable closed-form expressions considering Rician channels have been recently derived in [60], and evaluated in terms of outage and symbol error probability. However, the signal components are assumed to add coherently, which is not always strictly verified. In fact, there could be scenarios in which the same IRS has to serve multiple GU, thus introducing interference among surface elements. Besides, the presence of a direct, though weak, link is not considered, which would instead require to consider the additional interference between PRUs and GU. In order to account for the channel stochasticity, where interference between PRUs, BS and GUs is present, while considering the presence of a (possibly weak) GU-BS link, [65] proposes a channel gain approximation for IRS-assisted UAV-aided OFDMA communications. Although this contribution acknowledges several aspects of the communications, it does not explicitly model a multi-user multi-drone scenario. Moreover, the model introduces a huge number of degrees of freedom, one for each element of the IRS, which guarantees a great customization but at the same brings a tremendous complexity in terms of beam design. On the other hand, in [66] the number of degrees of freedom of the surface is drastically reduced by imposing a constraint on the phase shifts, which results to be extremely advantageous in terms of complexity but limits the surface to reflect the signal towards a single direction per time. Besides, [66] treats only a piece of the channel model, in a deterministic way, considering the sole path loss and not the stochastic term due to fading.

In light of the above, the major contributions given by this work are listed below.

- A comprehensive channel model for UAV-aided IRS-assisted OFDMA systems is derived. The proposal considers the presence of a swarm of drones equipped with IRSs split into an arbitrary number of patches to simultaneously serve multiple GUs. The proposed formulation captures inherent dependencies on the relative positions of the actors (UAVs, GUs, BS) in the environment, by explicitly modeling constructive/destructive interference patterns, under the constraint of a single controllable direction (two degrees of freedom) per patch. The derived expressions can be specialized in different cases as (i) absence of the direct BS-GU link, (ii) scattered (Rayleigh) BS-GU link, (iii) worst possible UAV-BS alignment, and (iv) best possible UAV-BS alignment. Furthermore, the expression of the SNR and maximum achievable data rate, given a certain outage probability, are derived and a theoretical analysis is provided.
- A Multi-Objective MINLP problem is formulated starting from the proposed channel model, aiming at maximizing the sum-rate of GUs and, at the same time, minimizing the differences among rates. Each node is served by a single BS (direct link) and by a swarm of drones (reflected links), thus requiring the jointly optimization of the phase

Symbol	Description	Symbol	Description
T	Total duration of the mission.	d_g^{BG}	Distance between the BS and g -th GU.
K	Number of discrete timesteps.	$d_{k,g,u,n,m}^{\text{RG}}$	Distance between the a GU and a PRU.
U	Number of UAVs involved in the mission.	$d_{k,u,n,m}^{\text{BR}}$	Distance between the BS and (n, m) -th PRU.
G	Number of GUs deployed on the ground.	$g_{k,i,g}^{\text{BG}}$	The channel gain between a GU and the BS.
I	Number of available subcarriers.	β^{BG}	The BS-GU channel power gain at 1 m.
N	Number of PRUs on the rows.	α^{BG}	The pathloss exponent for BS-GU link.
M	Number of PRUs on the columns.	κ^{BG}	K-factor for BS-GU link.
δ_t	Duration of a timestep.	Ω_g^{BG}	Average power for BS-GU link.
\mathbf{q}^{BS}	Location of the BS.	$\mathbf{g}_{k,i,g,u,p_u}^{\text{RG}}$	The channel gain between a patch and a GU.
\mathbf{q}_g^{G}	Location of the g -th GU.	$\mathbf{g}_{k,i,u,p_u}^{\text{BR}}$	The channel gain between a patch and the BS.
$\mathbf{q}_{k,u}^{\text{U}}$	Location of the u -th drone.	Φ_{k,u,p_u}	The IRS phase shift matrix of a patch.
$\mathbf{v}_{k,u}$	Speed of the u -th drone.	$G_{k,i,g}$	The overall channel gain perceived by g -th GU.
w	Area of a single PRU.	$C_{k,i,g}$	The uplink channel capacity of a GU.
$a_{k,u,m,n}$	The amplitude of the (n, m) -th element.	f_c	The carrier frequency.
$\phi_{k,u,m,n}$	The phase shift of the (n, m) -th element.	δ_f	The bandwidth of a single subcarrier.
P_u	Number of patches of the u -th IRS.	$P_{k,i,g}$	The transmit power of the g -th GUs.
E_{u,p_u}	Number of PRUs in a patch.	ρ^2	The thermal noise power.

TABLE 6.1: Main notation adopted in this work.

shift matrices of the IRSs and the trajectories of the drones. To achieve this goal a suitable objective function is defined, which includes a fairness factor that determines the importance of a uniform distribution of the resources. The problem is then reformulated in terms of patch scheduling, i.e., patch-GU assignment. To overcome the intractability of the non-convex cosine patterns related to wave interference, the problem is finally rearranged as a Markov Decision Process (MDP) and solved via a Deep Reinforcement Learning (DRL) method, namely Proximal Policy Optimization (PPO).

- An extensive simulation campaign is carried out to assess the validity of this work. First, the theoretical findings are corroborated by a thorough analysis which demonstrates, employing Monte Carlo simulations, the accuracy of the proposed channel model. Then, the solutions obtained by the DRL algorithm are analyzed for different number of patches and fairness levels. To further prove the effectiveness of the proposal, the obtained solution is compared with the special case of absence of the swarm and with a baseline approach.

The results of this work clearly indicate that (i) the presence of IRS-equipped drones strongly improves the channel capacity of the GUs, and (ii) the proposed solution outperforms the baseline in terms of total and average transmitted data.

The remainder of the present contribution is as follows: Section 6.2 describes the adopted system model. Section 6.3 presents the proposed channel model. Section 6.4 describes the problem formulation and the proposed solution. Section 6.5 analyzes accuracy of the model and investigates the obtained numerical results.

Notations adopted in this work are hereby described. Boldface lower and capital case letters refer to vectors and matrices, respectively; $j = \sqrt{-1}$ is the imaginary unit; $\text{atan2}(x)$ denotes the four-quadrant arctangent of a real number x ; \mathbf{x}^{T} is the transpose of a generic vector \mathbf{x} ; $\mathbf{x} \otimes \mathbf{y}$ denotes the Kronecker product between two generic vectors; $\text{diag}(\mathbf{x})$ represents a diagonal matrix whose diagonal is given by a vector \mathbf{x} ; $\arg(x)$ returns the phase of a complex number x ; $x \sim \mathcal{CN}(\mu, \sigma^2)$ define a circularly symmetric complex Gaussian distribution x with mean μ and variance σ^2 . For clarity, the adopted notations of this paper are summarized in Table 6.1.

6.2 System Model

The envisioned scenario, depicted in Figure 6.1, considers a swarm of U IRS-equipped UAVs is in charge of optimally reflecting the incident signal coming from a set of G GUs to enhance the channel quality at a single BS, located at a known position $\mathbf{q}^{\text{BS}} = [x^{\text{B}} \ y^{\text{B}} \ z^{\text{B}}]^{\text{T}} \in \mathbb{R}^3$.

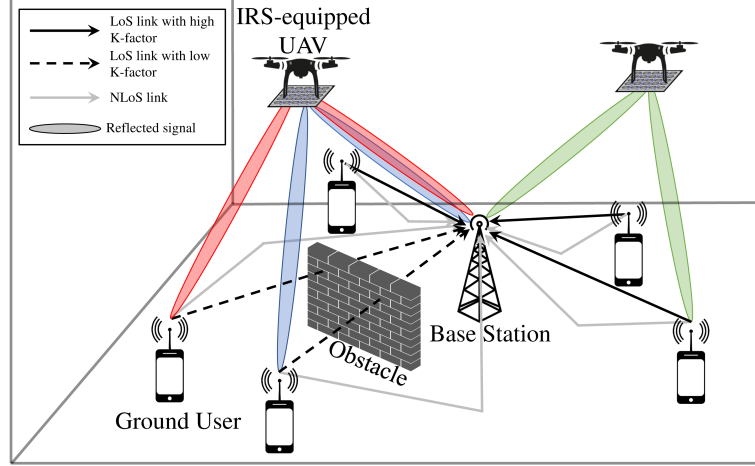


FIGURE 6.1: Reference scenario.

Both GUs and the BS are equipped with a single-antenna communication apparatus. The entire mission lasts T seconds, split into K timeslots of δ_t seconds each. Therefore, the UAVs fly following trajectories discretized into K waypoints $\mathbf{q}_{k,u}^{\text{U}} = [x_{k,u}^{\text{U}} \ y_{k,u}^{\text{U}} \ z_{k,u}^{\text{U}}]^{\text{T}} \in \mathbb{R}^3$ with a speed denoted by $\mathbf{v}_{k,u} = [v_{k,u}^x \ v_{k,u}^y \ v_{k,u}^z]^{\text{T}} \in \mathbb{R}^3$, $u = 1, \dots, U$, $k = 1, \dots, K$. IRSs are composed by $N \times M$ PRUs, all with the same size $w = d^x \times d^y$ m². The (n, m) -th element (i) has a center defined as $[(m - \frac{1}{2})d^x, (n - \frac{1}{2})d^y]^{\text{T}}$ with $m = 1 - \frac{M}{2}, \dots, \frac{M}{2}$, $n = 1 - \frac{N}{2}, \dots, \frac{N}{2}$ and (ii) reflects the incident signal through a complex factor $\Gamma_{k,u,n,m} = a_{k,u,n,m} e^{j\phi_{k,u,n,m}}$, where $a_{k,u,n,m}$ is the amplitude and $\phi_{k,u,n,m} \in [-\pi, \pi)$ is the phase shift. PRUs of the u -th IRS are grouped into P_u patches of $E_{u,p_u} = E_{u,p_u}^{\text{R}} \times E_{u,p_u}^{\text{C}} = (R_{u,p_u} - r_{u,p_u}) \times (C_{u,p_u} - c_{u,p_u})$ elements, $p_u = 1, \dots, P_u$, where $1 - \frac{M}{2} \leq r_{u,p_u} \leq R_{u,p_u} \leq \dots \leq r_{u,p_u} \leq R_{u,p_u} \leq \frac{M}{2}$ and $1 - \frac{N}{2} \leq c_{u,p_u} \leq C_{u,p_u} \leq \dots \leq c_{u,p_u} \leq C_{u,p_u} \leq \frac{N}{2}$. Besides, GUs are located at known positions $\mathbf{q}_g^{\text{G}} = [x_g^{\text{G}} \ y_g^{\text{G}} \ z_g^{\text{G}}]^{\text{T}} \in \mathbb{R}^3$, $g = 1, \dots, G$, leading to the definition of the distance $d_g^{\text{BG}} = \|\mathbf{q}^{\text{BS}} - \mathbf{q}_g^{\text{G}}\|$ from the BS, which does not change over time*. On the opposite, UAV-GU and BS-UAV distances depend on k -th timeslot since they are a function of the time-varying UAV positions. In particular, in far-field, the distance between each surface element of the u -th drone and the g -th GU can be approximated as [66]

$$d_{k,g,u,n,m}^{\text{RG}} \simeq d_{k,g,u}^{\text{RG}} - \left(m - \frac{1}{2}\right) d^x \sin \theta_{k,g,u}^{\text{RG}} \cos \varphi_{k,g,u}^{\text{RG}} + \left(n - \frac{1}{2}\right) d^y \sin \theta_{k,g,u}^{\text{RG}} \sin \varphi_{k,g,u}^{\text{RG}}, \quad (6.1)$$

where $d_{k,g,u}^{\text{RG}}$ is the distance from the center of the IRS, while $\theta_{k,g,u}^{\text{RG}}$ and $\varphi_{k,g,u}^{\text{RG}}$ denote the vertical and horizontal AoA of the signal. In particular, $\theta_{k,g,u}^{\text{RG}} = \text{acos} \left(\frac{z_g^{\text{U}} - z_{k,u}^{\text{G}}}{d_{k,g,u}^{\text{RG}}} \right)$ and $\varphi_{k,g,u}^{\text{RG}} = \text{atan2} \left(\frac{y_g^{\text{G}} - y_{k,u}^{\text{U}}}{x_g^{\text{G}} - x_{k,u}^{\text{U}}} \right)$.

*In this work, since the kinetics-related parameters are known to the BS, the Doppler effect is assumed to be perfectly compensated.

Similarly, the distance between the BS and each PRU of the u -th drone is

$$d_{k,u,n,m}^{\text{BR}} \simeq d_{k,u}^{\text{BR}} - \left(m - \frac{1}{2}\right) d^X \sin \theta_{k,u}^{\text{BR}} \cos \varphi_{k,u}^{\text{BR}} + \left(n - \frac{1}{2}\right) d^Y \sin \theta_{k,u}^{\text{BR}} \sin \varphi_{k,u}^{\text{BR}}, \quad (6.2)$$

where $d_{k,u}^{\text{BR}}$ is defined as the distance from the center of the IRS, while θ_k^{BR} and φ_k^{BR} are the vertical and horizontal AoD, so that $\theta_{k,g,u}^{\text{BR}} = \text{acos} \left(\frac{z_g^{\text{U}} - z_{k,u}^{\text{B}}}{d_{k,g,u}^{\text{BR}}} \right)$ and $\varphi_{k,g,u}^{\text{BR}} = \text{atan2} \left(\frac{y_g^{\text{B}} - y_{k,u}^{\text{U}}}{x_g^{\text{B}} - x_{k,u}^{\text{U}}} \right)$.

6.3 Proposed Channel Modeling

The whole communication system employs OFDMA which allows to mitigate the interference among the network entities. As a consequence, the whole available bandwidth B is split into I subcarriers of δ_f Hz each that can be assigned to GUs to transmit. Moreover, the power radiation pattern functions, including antenna gains, which characterize the antennas of GUs, BS, and IRSs are denoted by $F^{\text{GU}}(\theta, \varphi)$, $\forall g$, $F^{\text{BS}}(\theta, \varphi)$, and $F^{\text{IRS}}(\theta, \varphi)$, respectively. As known, these functions define how the transmitted/received power at each antenna varies along a certain direction in inclination θ and azimuth φ angles.

The channel gain between the g -th GU and the BS, in k -th timeslot and subcarrier $i = 1, \dots, I$ is

$$g_{k,i,g}^{\text{BG}} = \sqrt{\beta^{\text{BG}} d_g^{\text{BG} - \alpha^{\text{BG}}} F^{\text{BG}}(\theta_g^{\text{BG}}, \varphi_g^{\text{BG}}) h_{k,i,g}^{\text{BG}}}, \quad (6.3)$$

where β^{BG} denotes the channel power gain at the reference distance of 1 m, α^{BG} is the pathloss exponent, $F^{\text{BG}}(\cdot, \cdot) = F^{\text{BS}}(\cdot, \cdot) F^{\text{GU}}(\cdot, \cdot)$, and $h_{k,i,g}^{\text{BG}} \sim \mathcal{CN}(\mu_{i,g}^{\text{BG}}, 2\sigma_{k,i,g}^{\text{BG}^2})$ denotes the channel coefficient representing the stochastic fluctuation due to multi-path propagation and fading. It is worth noting that the channel envelope $|h_{k,i,g}^{\text{BG}}|$ is generally Rician [61], with K-factor $\kappa_g^{\text{BG}} = \frac{|\mu_{i,g}^{\text{BG}}|^2}{2\sigma_{k,i,g}^{\text{BG}^2}}$ and average power $\Omega_g^{\text{BG}} = |\mu_{i,g}^{\text{BG}}|^2 + 2\sigma_{k,i,g}^{\text{BG}^2} = 1$, since the presence of a (possibly weak) LoS path is taken into account — according to the propagation characteristics of the actual environment and the presence of obstacles, as depicted in Figure 6.1. Therefore, the channel coefficient can be modeled as

$$h_{k,i,g}^{\text{BG}} = \sqrt{\frac{\kappa_g^{\text{BG}}}{\kappa_g^{\text{BG}} + 1}} \bar{h}_{k,i,g}^{\text{BG}} + \sqrt{\frac{1}{\kappa_g^{\text{BG}} + 1}} \tilde{h}_{k,i,g}^{\text{BG}}. \quad (6.4)$$

In particular, $\bar{h}_{i,g}^{\text{BG}} = e^{-j2\pi f_i \frac{d_g^{\text{BG}}}{c}}$ is the LoS component, characterized by a phase shift depending on the user's subcarrier index, $f_i = f_c + i\delta_f$, f_c is the carrier frequency, and $\tilde{h}_{k,i,g}^{\text{BG}} \sim \mathcal{CN}(0, 1)$ is circular symmetric Gaussian distributed, describing small-scale fading resulting from NLoS propagation.

The channel gain between the p_u -th patch of the u -th UAV and the g -th GU in timeslot k , subcarrier i , is obtained by collecting in a vector the corresponding E_{u,p_u} channel gains of the PRUs, i.e.

$$\mathbf{g}_{k,i,g,u,p_u}^{\text{RG}} = \sqrt{\beta^{\text{RG}} d_{k,g,u}^{\text{RG}} - \alpha^{\text{RG}}} F^{\text{RG}}(\theta_{k,g,u}^{\text{RG}}, \varphi_{k,g,u}^{\text{RG}}) \mathbf{h}_{k,i,g,u,p_u}^{\text{RG}}, \quad (6.5)$$

$$\mathbf{h}_{k,i,g,u,p_u}^{\text{RG}} = \sqrt{\frac{\kappa_{k,g,u}^{\text{RG}}}{\kappa_{k,g,u}^{\text{RG}} + 1}} \bar{\mathbf{h}}_{k,i,g,u,p_u}^{\text{RG}} + \sqrt{\frac{1}{\kappa_{k,g,u}^{\text{RG}} + 1}} \tilde{\mathbf{h}}_{k,i,g,u,p_u}^{\text{RG}}, \quad (6.6)$$

$$\bar{\mathbf{h}}_{k,i,g,u,p_u}^{\text{RG}} = e^{-j2\pi f_i \frac{d_{k,g,u}^{\text{RG}}}{c}} \mathbf{h}_{k,i,g,u,p_u}^{\text{RG, LoS}}, \quad (6.7)$$

$\tilde{\mathbf{h}}_{k,i,g,u,p_u}^{\text{RG}} \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_{E_{u,p_u}})$, and $F^{\text{RG}}(\cdot, \cdot) = F^{\text{IRS}}(\cdot, \cdot) F^{\text{GU}}(\cdot, \cdot)$. Moreover, $\mathbf{h}_{k,i,g,u,p_u}^{\text{RG, LoS}} \in \mathbb{C}^{E_{u,p_u} \times 1}$ in (6.7) denotes the far-field array response defined as

$$\mathbf{h}_{k,i,g,u,p_u}^{\text{RG, LoS}} = \begin{bmatrix} e^{j2\pi f_i \frac{(c_{u,p_u} - \frac{1}{2}) d^{\text{X}} \sin \theta_{k,g,u}^{\text{RG}} \cos \varphi_{k,g,u}^{\text{RG}}}{c}} \dots e^{j2\pi f_i \frac{(C_{u,p_u} - \frac{1}{2}) d^{\text{X}} \sin \theta_{k,g,u}^{\text{RG}} \cos \varphi_{k,g,u}^{\text{RG}}}{c}} \end{bmatrix}^{\text{T}} \\ \otimes \begin{bmatrix} e^{j2\pi f_i \frac{(r_{u,p_u} - \frac{1}{2}) d^{\text{Y}} \sin \theta_{k,g,u}^{\text{RG}} \sin \varphi_{k,g,u}^{\text{RG}}}{c}} \dots e^{j2\pi f_i \frac{(R_{u,p_u} - \frac{1}{2}) d^{\text{Y}} \sin \theta_{k,g,u}^{\text{RG}} \sin \varphi_{k,g,u}^{\text{RG}}}{c}} \end{bmatrix}^{\text{T}}, \quad (6.8)$$

Similarly, the channel gain between a certain patch p_u of a UAV u and the BS is

$$\mathbf{g}_{k,i,u,p_u}^{\text{BR}} = \sqrt{\beta^{\text{BR}} d_{k,u}^{\text{BR}} - \alpha^{\text{BR}}} F^{\text{BR}}(\theta_{k,u}^{\text{BR}}, \varphi_{k,u}^{\text{BR}}) \mathbf{h}_{k,i,u,p_u}^{\text{BR}}, \quad (6.9)$$

$$\mathbf{h}_{k,i,u,p_u}^{\text{BR}} = \sqrt{\frac{\kappa_{k,u}^{\text{BR}}}{\kappa_{k,u}^{\text{BR}} + 1}} \bar{\mathbf{h}}_{k,i,u,p_u}^{\text{BR}} + \sqrt{\frac{1}{\kappa_{k,u}^{\text{BR}} + 1}} \tilde{\mathbf{h}}_{k,i,u,p_u}^{\text{BR}}, \quad (6.10)$$

$$\bar{\mathbf{h}}_{k,i,u,p_u}^{\text{BR}} = e^{-j2\pi f_i \frac{d_{k,u}^{\text{BR}}}{c}} \mathbf{h}_{k,i,u,p_u}^{\text{BR, LoS}}, \quad (6.11)$$

where $\mathbf{h}_{k,i,u,p_u}^{\text{BR, LoS}} \in \mathbb{C}^{E_{u,p_u} \times 1}$ reads

$$\mathbf{h}_{k,i,u,p_u}^{\text{BR, LoS}} = \begin{bmatrix} e^{j2\pi f_i \frac{(c_{u,p_u} - \frac{1}{2}) d^{\text{X}} \sin \theta_{k,u}^{\text{BR}} \cos \varphi_{k,u}^{\text{BR}}}{c}} \dots e^{j2\pi f_i \frac{(C_{u,p_u} - \frac{1}{2}) d^{\text{X}} \sin \theta_{k,u}^{\text{BR}} \cos \varphi_{k,u}^{\text{BR}}}{c}} \end{bmatrix}^{\text{T}} \\ \otimes \begin{bmatrix} e^{j2\pi f_i \frac{(r_{u,p_u} - \frac{1}{2}) d^{\text{Y}} \sin \theta_{k,u}^{\text{BR}} \sin \varphi_{k,u}^{\text{BR}}}{c}} \dots e^{j2\pi f_i \frac{(R_{u,p_u} - \frac{1}{2}) d^{\text{Y}} \sin \theta_{k,u}^{\text{BR}} \sin \varphi_{k,u}^{\text{BR}}}{c}} \end{bmatrix}^{\text{T}}, \quad (6.12)$$

$F^{\text{BR}}(\cdot, \cdot) = F^{\text{BS}}(\cdot, \cdot) F^{\text{IRS}}(\cdot, \cdot)$, and $\tilde{\mathbf{h}}_{k,i}^{\text{BR}} \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_{E_{u,p_u}})$. It is worth noting that $\kappa_{k,g,u}^{\text{BR}}$ depends on the UAV, specifically on the elevation angle $\bar{\theta}_{k,g,u}^{\text{BR}} = \frac{\pi}{2} - \theta_{k,g,u}^{\text{BR}}$ [21]:

$$\kappa_{k,u}^{\text{BR}} = \kappa_{k,u}^{\text{MIN,BR}} e^{\gamma^{\text{BR}} \bar{\theta}_{k,g,u}^{\text{BR}}}, \quad \gamma^{\text{BR}} = \frac{2}{\pi} \ln \frac{\kappa_{k,u}^{\text{MAX,BR}}}{\kappa_{k,u}^{\text{MIN,BR}}}, \quad (6.13)$$

with $\kappa_{k,g,u}^{\text{MIN,BR}}$ and $\kappa_{k,g,u}^{\text{MAX,BR}}$ the minimum and maximum possible K-factors, respectively. The same holds true for $\kappa_{k,g,u}^{\text{BR}}$ and $\kappa_{k,g,u}^{\text{BG}}$. The IRS phase shift matrix $\Phi_{k,u,p_u} \in \mathbb{C}^{E_{u,p_u} \times E_{u,p_u}}$, for each timeslot k , UAV u , patch p_u is defined as

$$\Phi_{k,u,p_u} = \text{diag} \left(\Gamma_{k,u,r_{u,p_u},c_{u,p_u}}, \dots, \Gamma_{k,u,r_{u,p_u},C_{u,p_u}}, \dots, \Gamma_{k,u,R_{u,p_u},c_{u,p_u}}, \dots, \Gamma_{k,u,R_{u,p_u},C_{u,p_u}} \right) \quad (6.14)$$

recalling that in general $\Gamma_{k,u,n,m} = a_{k,u,n,m} e^{j\phi_{k,u,n,m}}$. Since each patch p_u is meant to coherently reflect the incident signal towards a certain location (to serve one of the G GUs), all its elements can share the same amplitude $a_{k,u,n,m} = a_{k,u,p_u}$ for all (n, m) belonging to patch p_u , and their phases can be described in terms of two parameters only, denoted by ϕ_{k,u,p_u}^X and ϕ_{k,u,p_u}^Y . Specifically, it is possible to reduce the amount of degrees of freedom introduced by PRUs by imposing that:

$$\frac{2\pi f_i}{c} \left(d^X \left(m - \frac{1}{2} \right) \phi_{k,u,p_u}^X + d^Y \left(n - \frac{1}{2} \right) \phi_{k,u,p_u}^Y \right) = \phi_{k,u,p_u}.$$

By considering (6.5), (6.9), and (6.14), the composite channel gain G_{k,i,g,u,p_u} is

$$\begin{aligned} G_{k,i,g,u,p_u} &= \mathbf{g}_{k,i,u,p_u}^{\text{BR}} \mathbf{\Phi}_{k,u}^{\text{T}} \mathbf{g}_{k,i,g,u,p_u}^{\text{RG}} \\ &= \eta_{k,g,u} \sum_{n=r_u,p_u}^{R_u,p_u} \sum_{m=c_u,p_u}^{C_u,p_u} \underbrace{h_{k,i,u,p_u,n,m}^{\text{BR}} \mathbf{\Gamma}_{k,u,p_u,n,m}^{\text{T}} h_{k,i,g,u,p_u,n,m}^{\text{RG}}}_{h_{k,i,g,u,p_u,n,m}^{\text{BRG}}} \end{aligned} \quad (6.15)$$

where $h_{k,i,g,u,p_u,n,m}^{\text{BR}}$ and $h_{k,i,g,u,p_u,n,m}^{\text{RG}}$ are the components of the channel vectors $\mathbf{h}_{k,i,g,u,p_u}^{\text{BR}}$ and $\mathbf{h}_{k,i,g,u,p_u}^{\text{RG}}$, respectively. Moreover,

$$\begin{aligned} \eta_{k,g,u} &= \sqrt{\beta^{\text{BRG}} d_{k,u}^{\text{BR} - \alpha^{\text{BR}}} d_{k,g,u}^{\text{RG} - \alpha^{\text{RG}}} F^{\text{BRG}}(\theta_{k,u}^{\text{BR}}, \varphi_{k,u}^{\text{BR}}, \theta_{k,g,u}^{\text{RG}}, \varphi_{k,g,u}^{\text{RG}})}, \\ \lambda_g &= \sqrt{\beta^{\text{BG}} d_g^{\text{BG} - \alpha^{\text{BG}}} F^{\text{BG}}(\theta_g^{\text{BG}}, \varphi_g^{\text{BG}})}, F^{\text{BRG}}(\cdot, \cdot, \cdot, \cdot) = F^{\text{BR}}(\cdot, \cdot) F^{\text{RG}}(\cdot, \cdot), \beta^{\text{BRG}} = \beta^{\text{BR}} \beta^{\text{RG}}, \end{aligned}$$

and $h_{k,i,g,u,p_u,n,m}^{\text{BRG}}$ describes the composite BS-PRU-GU channel coefficient.

The composite BS-PRU-GU channel coefficient involves the pairwise product of two complex Gaussian RVs, whose distribution, named complex double Gaussian, is given in terms of an infinite sum of modified Bessel functions [62]. For better tractability, it is however possible to approximate such a product through a complex Gaussian [65], thus implying that the envelope $|h_{k,i,g,u,p_u,n,m}^{\text{BRG}}|$ is a Rician RV. The following expression for the channel coefficient, for the generic (n, m) -the element of the p_u -th patch, is finally obtained:

$$\begin{aligned} h_{k,i,g,u,p_u,n,m}^{\text{BRG}} &= a_{k,u,p_u} \left(\sqrt{\overline{\kappa}_{k,g,u}^{\text{BRG}}} e^{-j \frac{2\pi f_i}{c} \Psi_{k,i,u,p_u,n,m}} \right. \\ &\quad \left. + \sqrt{\tilde{\kappa}_{k,g,u}^{\text{BRG}}} \tilde{h}^{\text{BRG}} \right), \end{aligned} \quad (6.16)$$

where

$$\overline{\kappa}_{k,g,u}^{\text{BRG}} = \frac{\kappa_{k,u}^{\text{BR}} \kappa_{k,g,u}^{\text{RG}}}{(\kappa_{k,u}^{\text{BR}} + 1)(\kappa_{k,g,u}^{\text{RG}} + 1)}, \quad (6.17)$$

$$\tilde{\kappa}_{k,g,u}^{\text{BRG}} = \frac{\kappa_{k,u}^{\text{BR}} + \kappa_{k,g,u}^{\text{RG}}}{(\kappa_{k,u}^{\text{BR}} + 1)(\kappa_{k,g,u}^{\text{RG}} + 1)}, \quad (6.18)$$

$$\begin{aligned} \Psi_{k,i,g,u,p_u,n,m} &= d_{k,u}^{\text{BR}} + d_{k,g,u}^{\text{RG}} - d^X \left(m - \frac{1}{2} \right) \psi_{k,g,u,p_u}^X \\ &\quad - d^Y \left(n - \frac{1}{2} \right) \psi_{k,g,u,p_u}^Y, \end{aligned}$$

$$\begin{aligned}\psi_{k,g,u,p_u}^X &= \sin \theta_{k,u}^{\text{BR}} \cos \varphi_{k,u}^{\text{BR}} + \sin \theta_{k,g,u}^{\text{RG}} \cos \varphi_{k,g,u}^{\text{RG}} + \phi_{k,u,p_u}^X, \\ \psi_{k,g,u,p_u}^Y &= \sin \theta_{k,u}^{\text{BR}} \sin \varphi_{k,u}^{\text{BR}} + \sin \theta_{k,g,u}^{\text{RG}} \sin \varphi_{k,g,u}^{\text{RG}} + \phi_{k,u,p_u}^Y\end{aligned}$$

and $\tilde{h}^{\text{BRG}} \sim \mathcal{CN}(0, 1)$ since phase terms are irrelevant to the scatter component [57].

The expression derived in (6.15), which includes (6.16) and the subsequent definitions, requires further elaboration in order to obtain a more compact form, with a lower computational complexity. To this aim, the following result will be useful to compute the sum of the PRUs' channel coefficients belonging to the same patch p_u .

Lemma 2. *The finite sum of complex exponentials $e^{j(l-\frac{1}{2})x}$ for $l = 1 - \frac{L}{2}, \dots, \frac{L}{2}$, with L even, can be rearranged as a ratio of sine (or equivalently as L times a ratio of sinc) functions, as*

$$\sum_{l=1-\frac{L}{2}}^{\frac{L}{2}} e^{j(l-\frac{1}{2})x} = \frac{\sin(\frac{L}{2}x)}{\sin(\frac{x}{2})} = L \frac{\text{sinc}(\frac{L}{2}x)}{\text{sinc}(\frac{x}{2})}$$

Proof. Please refer to [66][67]. ■

The following theorem is one of the main contributions of this work. Indeed, it characterizes the overall gain $G_{k,i,g}$, given by the sum of all signals reflected towards user g (by any patch from any UAV) plus the direct link.

Theorem 3. *Let $G_{k,i,g}$ be the channel gain perceived by a GU g , in timeslot k and subcarrier i . The channel envelope $|G_{k,i,g}|$ is a Rician RV having $\kappa_{k,i,g} = \frac{\nu_{k,i,g}^2}{2\sigma_{G_{k,i,g}}^2}$ and average power $\Omega_{k,i,g} = \nu_{k,i,g}^2 + 2\sigma_{G_{k,i,g}}^2$, with $\nu_{k,i,g}^2$ and $2\sigma_{G_{k,i,g}}^2$ defined in (6.23) and (6.24), respectively.*

Proof. Leveraging the result of Lemma 2, since (6.15) is a sum of complex Gaussian distributions, and assuming that all the elements of the same patch p_u have the same amplitude a_{k,u,p_u} , the channel coefficients sum of all the PRUs $G_{k,i,g,u,p_u} \sim \mathcal{CN}(\mu_{G_{k,i,g,u,p_u}}, 2\sigma_{G_{k,i,g,u,p_u}}^2)$ is described by the mean

$$\begin{aligned}\mu_{G_{k,i,g,u,p_u}} &= a_{k,u,p_u} \eta_{k,g,u} \sqrt{\frac{1}{\tilde{K}_{k,g,u}^{\text{BRG}}}} e^{-j\left(\frac{2\pi f_i}{c}(d_{k,u}^{\text{BR}} + d_{k,g,u}^{\text{RG}})\right)} \\ &\times \sum_{m=C_{u,p_u}}^{C_{u,p_u}} e^{j\frac{2\pi f_i}{c}d^X(m-\frac{1}{2})} \psi_{k,g,u,p_u}^X \sum_{n=R_{u,p_u}}^{R_{u,p_u}} e^{j\frac{2\pi f_i}{c}d^Y(n-\frac{1}{2})} \psi_{k,g,u,p_u}^Y \\ &= a_{k,u,p_u} \eta_{k,g,u} \sqrt{\frac{1}{\tilde{K}_{k,g,u}^{\text{BRG}}}} \underbrace{\frac{\sin\left(\frac{\pi f_i}{c}d^X E_{u,p_u}^C \psi_{k,g,u,p_u}^X\right) \sin\left(\frac{\pi f_i}{c}d^Y E_{u,p_u}^R \psi_{k,g,u,p_u}^Y\right)}{\sin\left(\frac{\pi f_i}{c}d^X \psi_{k,g,u,p_u}^X\right) \sin\left(\frac{\pi f_i}{c}d^Y \psi_{k,g,u,p_u}^Y\right)}}_{\chi_{k,g,u,p_u}} \\ &\times \underbrace{e^{-j\left(\frac{2\pi f_i}{c}(d_{k,u}^{\text{BR}} + d_{k,g,u}^{\text{RG}})\right)}}_{\omega_{k,i,g,u}}\end{aligned}\tag{6.19}$$

and by the second moment

$$2\sigma_{G_{k,i,g,u,p_u}}^2 = a_{k,u,p_u}^2 E_{u,p_u} \eta_{k,g,u}^2 \tilde{K}_{k,g,u}^{\text{BRG}}.\tag{6.20}$$

Finally, the expression of the channel gain perceived by a certain GU g is

$$G_{k,i,g} = \sum_{u=1}^U \sum_{p_u=1}^{P_u} G_{k,i,g,u,p_u} + g_{k,i,g}^{\text{BG}}.\tag{6.21}$$

Following the same rationale adopted in [65] it is possible to derive the K-factor $\kappa_{k,i,g}$ and average power $\Omega_{k,i,g}$ of the envelope of $G_{k,i,g}$ as

$$\kappa_{k,i,g} = \frac{\nu_{k,i,g}^2}{2\sigma_{G_{k,i,g}}^2}, \quad \Omega_{k,i,g} = \nu_{k,i,g}^2 + 2\sigma_{G_{k,i,g}}^2, \quad (6.22)$$

being $\nu_{k,i,g}^2 \triangleq |\mu_{G_{k,i,g}}|^2$ the squared LoS component, defined as

$$\begin{aligned} \nu_{k,i,g}^2 &= \sum_{u=1}^U \sum_{p_u=1}^{P_u} |\mu_{G_{k,i,g,u,p_u}}|^2 + |\mu_{i,g}^{\text{BG}}|^2 + 2 \sum_{u \geq u'} \sum_{p_u > p'_u} \Re(\mu_{G_{k,i,g,u,p_u}}^* \mu_{G_{k,i,g,u',p'_u}}) \\ &+ 2 \sum_{u=1}^U \sum_{p_u=1}^{P_u} \Re(\mu_{G_{k,i,g,u,p_u}}^* \mu_{i,g}^{\text{BG}}) \\ &= \underbrace{\sum_{u=1}^U \sum_{p_u=1}^{P_u} \chi_{k,i,g,u,p_u}^2 + 2 \sum_{u \geq u'} \sum_{p_u > p'_u} |\chi_{k,i,g,u,p_u}| |\chi_{k,i,g,u',p'_u}| \cos(\varphi_{k,i,g,u} - \varphi_{k,i,g,u'})}_{\nu_{k,i,g}^{\text{BRG}^2}} \\ &+ \underbrace{\lambda_g^2 \bar{\kappa}_g^{\text{BG}} + 2 \sum_{u=1}^U \sum_{p_u=1}^{P_u} |\chi_{k,i,g,u,p_u}| |\lambda_g \sqrt{\bar{\kappa}_g^{\text{BG}}}| \cos\left(\varphi_{k,i,g,u} + 2\pi f_i \frac{d_g^{\text{BG}}}{c}\right)}_{\nu_{k,i,g}^{\text{BG}^2}} \end{aligned} \quad (6.23)$$

with $\varphi_{k,i,g,u} = \arg(\omega_{k,i,g,u})$, $\forall k, i, g, u$ and $\bar{\kappa}_g^{\text{BG}} = \frac{\kappa_g^{\text{BG}}}{\kappa_g^{\text{BG}} + 1}$. Finally, the NLoS component reads

$$2\sigma_{G_{k,i,g}}^2 = \underbrace{\sum_{u=1}^U \eta_{k,g,u}^2 \tilde{\kappa}_{k,g,u}^{\text{BRG}} \sum_{p_u=1}^{P_u} a_{k,u,p_u}^2 E_{u,p_u}}_{2\sigma_{G_{k,i,g}}^{\text{BRG}^2}} + \underbrace{\lambda_g^2 \tilde{\kappa}_g^{\text{BG}}}_{2\sigma_{G_{k,i,g}}^{\text{BG}^2}}, \quad (6.24)$$

where $\tilde{\kappa}_g^{\text{BG}} = (\kappa_g^{\text{BG}} + 1)^{-1}$. ■

Corollary 3. *The proposed general channel model can be specialized in case of scattered, i.e., Rayleigh fading ($\kappa_g^{\text{BG}} \rightarrow 0$), or even totally absent direct BS-GU link ($\lambda_g \rightarrow 0$). In both cases, the squared LoS component $\nu_{k,i,g}^2 \rightarrow \nu_{k,i,g}^{\text{BRG}^2}$, since $\nu_{k,i,g}^{\text{BG}^2} \rightarrow 0$. However, the NLoS component is $2\sigma_{G_{k,i,g}}^2 \rightarrow 2\sigma_{G_{k,i,g}}^{\text{BRG}^2} + \lambda_g^2$ in the former case, while is $2\sigma_{G_{k,i,g}}^2 \rightarrow 2\sigma_{G_{k,i,g}}^{\text{BRG}^2}$ in the latter case since $2\sigma_{G_{k,i,g}}^{\text{BRG}^2} \rightarrow 0$.*

Based on the above results, it is possible to obtain expressions for the maximum achievable data rates, outage probability, and other inherent performance metrics. In particular, recalling Shannon's capacity formula, the channel capacity in timeslot k and subcarrier i is given by

$$C_{k,i,g} = \delta_F \log_2 \left(1 + \frac{P_{k,i,g} |G_{k,i,g}|^2}{\rho^2} \right), \quad (6.25)$$

where $P_{k,i,g}$ is the transmit power of GUs in i -th subcarrier, $\rho^2 = N_0 \delta_F$ is the noise power, and N_0 is the spectral noise power. Given a maximum achievable data rate $R_{k,i,g}$, to guarantee a reliable communication, it is required that the outage probability $p_{k,i,g}$ remains below a

threshold ε , i.e.

$$\begin{aligned} p_{k,i,g} &= \mathbb{P}(C_{k,i,g} < R_{k,i,g}) = \mathbb{P}\left(|G_{k,i,g}|^2 < \frac{\rho^2(2^{\frac{R_{k,i,g}}{\delta_F}} - 1)}{P_{k,i,g}}\right) \\ &= F_{|G_{k,i,g}|^2}\left(\underbrace{\frac{\rho^2(2^{\frac{R_{k,i,g}}{\delta_F}} - 1)}{P_{k,i,g}}}_u\right) \leq \varepsilon, \quad \forall \quad \begin{array}{l} k = 1, \dots, K, \\ i = 1, \dots, I, \end{array} \end{aligned}$$

with $F_{|G_{k,i,g}|^2}$ denoting the CDF of $|G_{k,i,g}|^2$. Therefore, considering the maximum tolerable outage, i.e., $p_{k,i,g} = \varepsilon$, the previous equation can be rewritten as

$$F_{|G_{k,i,g}|^2}(u) = 1 - Q_m\left(\sqrt{2\kappa_{k,i,g}}, \sqrt{\frac{2(\kappa_{k,i,g} + 1)}{\Omega_{k,i,g}}}u\right) = \varepsilon,$$

with Q_m denoting the Marcum Q-function. It follows that the inverse expression, with respect to the second argument is:

$$\sqrt{\frac{2(\kappa_{k,i,g} + 1)}{\Omega_{k,i,g}}}u = Q_m^{-1}(\sqrt{2\kappa_{k,i,g}}, 1 - \varepsilon) \triangleq \zeta_{k,i,g}(\varepsilon).$$

Since the inverse Marcum Q-function has no closed-form expression, to avoid numerical computation an approximated formula can be used as in [45, Eq. 17]

$$\zeta_{k,i,g}(\varepsilon) = \begin{cases} \sqrt{-2\log(1 - \varepsilon)}e^{\frac{\kappa_{k,i,g}}{2}}, & \text{for } \kappa_{k,i,g} \leq \frac{K_0^2}{2} \\ \sqrt{2\kappa_{k,i,g} + \frac{1}{2Q^{-1}(\varepsilon)}} \\ \times \log\left(\frac{\sqrt{2\kappa_{k,i,g}}}{\sqrt{2\kappa_{k,i,g}} - Q^{-1}(\varepsilon)}\right) - Q^{-1}(\varepsilon), & \text{for } \kappa_{k,i,g} \geq \frac{K_0^2}{2} \end{cases}$$

with K_0 the intersection of the sub-functions at $\sqrt{2\kappa_{k,i,g}} > Q^{-1}(\varepsilon)$ and $Q^{-1}(x)$ the inverse Q-function. Finally, since

$$u = \frac{\rho^2(2^{\frac{R_{k,i,g}}{\delta_F}} - 1)}{P_{k,i,g}} \approx \frac{\zeta_{k,i,g}(\varepsilon)^2 \Omega_{k,i,g}}{2(\kappa_{k,i,g} + 1)},$$

the expression of the achievable data rate can be derived as

$$R_{k,i,g} = \delta_F \log_2 \left(1 + \underbrace{\frac{P_{k,i,g} \zeta_{k,i,g}(\varepsilon)^2 \Omega_{k,i,g}}{2(\kappa_{k,i,g} + 1) \rho^2}}_{\text{SNR}_{k,i,g}} \right), \quad (6.26)$$

with $\text{SNR}_{k,i,g}$ denoting the SNR.

In the following the scaling behavior of the SNR and other properties are analyzed.

Theorem 4. *The SNR of the g -th GU scales by a factor*

$$\sum_{u=1}^U \sum_{p_u=1}^{P_u} E_{u,p_u} \leq \xi \leq \left(\sum_{u=1}^U \sum_{p_u=1}^{P_u} E_{u,p_u} \right)^2. \quad (6.27)$$

Proof. The maximum squared LoS component is obtained when the swarm is perfectly aligned with BS and GUs, which $\forall k, i, g$ and $\forall u \geq u'$ corresponds to

$$\left\{ \begin{array}{l} \text{mod} \left(\frac{2\pi f_i}{c} (\varphi_{k,i,g,u} - \varphi_{k,i,g,u'}), 2\pi \right) = 0, \end{array} \right. \quad (6.28)$$

$$\left\{ \begin{array}{l} \text{mod} \left(\varphi_{k,i,g,u} + 2\pi f_i \frac{d_g^{\text{BG}}}{c}, 2\pi \right) = 0, \end{array} \right. \quad (6.29)$$

thus allowing to recast it in the more compact notation

$$\nu_{k,i,g}^2 = \left(\sum_{u=1}^U \sum_{p_u=1}^{P_u} |\chi_{k,i,g,u,p_u}| + \lambda_g \bar{\kappa}_g^{\text{BG}} \right)^2. \quad (6.30)$$

Therefore, assuming that all the patches of the swarm perfectly reflect the incident signal towards the desired GU yields

$$\begin{aligned} \nu_{k,i,g}^{\text{MAX}^2} &= \lim_{\substack{\{\psi_{k,g,u,p_u}^X\} \rightarrow 2\pi b \\ \{\psi_{k,g,u,p_u}^Y\} \rightarrow 2\pi c}} \nu_{k,i,g}^2, \\ &= \left(\sum_{u=1}^U \eta_{k,g,u} \bar{\kappa}_{k,g,u}^{\text{BRG}} \sum_{p_u=1}^{P_u} a_{k,u,p_u} E_{u,p_u} + \lambda_g \bar{\kappa}_g^{\text{BG}} \right)^2, \end{aligned}$$

for any $b, c \in \mathbb{Z}$, which is verified when

$$\begin{aligned} \phi_{k,u,p_u}^X &= -\sin \theta_{k,u}^{\text{BR}} \cos \varphi_{k,u}^{\text{BR}} - \sin \theta_{k,g,u}^{\text{RG}} \cos \varphi_{k,g,u}^{\text{RG}}, \\ \phi_{k,u,p_u}^Y &= -\sin \theta_{k,u}^{\text{BR}} \sin \varphi_{k,u}^{\text{BR}} - \sin \theta_{k,g,u}^{\text{RG}} \sin \varphi_{k,g,u}^{\text{RG}}. \end{aligned} \quad (6.31)$$

Consequently,

$$\lim_{\{E_{u,p_u}\} \rightarrow +\infty} \nu_{k,i,g}^{\text{MAX}^2} = O(E^2),$$

being $E = \sum_{u=1}^U \sum_{p_u=1}^{P_u} E_{u,p_u}$. Furthermore, recalling the definition of the average power in (6.22) and the NLoS component in (6.24), as $\{E_{u,p_u}\} \rightarrow +\infty$ the following inequalities hold true

$$O(E) \leq \Omega_{k,i,g} \leq O(E^2) \quad (6.32)$$

and since

$$\lim_{\{E_{u,p_u}\} \rightarrow +\infty} \frac{\zeta_{k,i,g}(\varepsilon)^2}{2(\kappa_{k,i,g} + 1)} \approx \lim_{\{E_{u,p_u}\} \rightarrow +\infty} \frac{2\kappa_{k,i,g}}{2(\kappa_{k,i,g} + 1)} = O(1)$$

the SNR derived in (6.26) scales as $\sum_{u=1}^U \sum_{p_u=1}^{P_u} E_{u,p_u} \leq \xi \leq (\sum_{u=1}^U \sum_{p_u=1}^{P_u} E_{u,p_u})^2$. ■

Corollary 4. *Following the same rationale of Theorem 4, it is not difficult to show that in case of a single drone ($U = 1$), since the alignment condition (6.28) is inherently satisfied, the SNR measured at a specific GU illuminated by $\bar{P} \leq P_u$ scales as $\zeta \geq \left(\sum_{p_u=1}^{\bar{P}} E_p \right)^2$.*

Corollary 5. *In the special case of no direct BS-GU link ($\lambda_g \rightarrow 0$), a single drone ($U = 1$) illuminating one GU ($G = 1$) with the entire IRS ($P = 1$), the channel follows a Rician distribution characterized by $\kappa_{k,i} = \frac{\bar{\kappa}_k^{\text{BRG}}}{\bar{\kappa}_k^{\text{BRG}}} E$ and $\Omega_k = a_k^2 \eta_k^2 (\bar{\kappa}_k^{\text{BRG}} E^2 + \bar{\kappa}_k^{\text{BRG}} E)$, where g, u, p are omitted for brevity. Further, the corresponding SNR scales as $\zeta = E^2$.*

6.4 Trajectory and Phase Shift Matrix Optimization

Leveraging the model derived in the previous Section, an optimization problem aiming at maximizing the sum-rate of nodes and, at the same time, at fairly distributing the resources to the GUs is formulated and a resolution approach provided.

6.4.1 Problem formulation

Let assume that (i) the UAVs fly at a constant altitude $z_{k,u}^U = H \forall k, u$, (ii) the GUs have a dedicated subcarrier, and (iii) the drones have the same number of patches, i.e, $P_u = P \forall u$. The objective is to maximize the sum-rate $\sum_{k,i,g} R_{k,i,g}$, with $R_{k,i,g}$ given in (6.26) and, at the same time, to minimize the sum of the absolute differences $\sum_{g>g'} \left| \sum_{k=1}^K \sum_{i=1}^I R_{k,i,g} - R_{k,i,g'} \right|$ among these amounts, so as to provide an adequate level of fairness among users. This multi-objective problem requires the optimization of the trajectories $\mathbf{Q} = \{\mathbf{q}_{k,u}, \forall k, u\}$, the speed profiles $\mathbf{V} = \{\mathbf{v}_{k,u}, \forall k, u\}$, and the phase shift matrices $\Phi_k^X = \{\phi_{k,u,p}^X, \forall u, p\}$ and $\Phi_k^Y = \{\phi_{k,u,p}^Y, \forall u, p\}$. Therefore, it can be formulated as follows

$$\max_{\mathbf{Q}, \mathbf{V}, \{\Phi_k^X\}, \{\Phi_k^Y\}} \frac{\sum_{k=1}^K \sum_{i=1}^I \sum_{g=1}^G R_{k,i,g}}{1 + \varpi \sum_{g>g'} \left| \sum_{k=1}^K \sum_{i=1}^I R_{k,i,g} - R_{k,i,g'} \right|} \quad \text{s.t.} \quad (6.33)$$

$$\mathbf{q}_{1,u} = \mathbf{q}_{0,u}, \quad u = 1, \dots, U, \quad (6.33a)$$

$$\mathbf{q}_{k+1,u} = \mathbf{q}_{k,u} + \delta_t \mathbf{v}_{k,u}, \quad k = 1, \dots, K, u = 1, \dots, U, \quad (6.33b)$$

$$\mathbf{v}_{k,u} \leq v^{\text{MAX}}, \quad k = 1, \dots, K, u = 1, \dots, U, \quad (6.33c)$$

$$\|\mathbf{q}_{k,u} - \mathbf{q}_{k,u'}\| \geq d^{\text{MIN}}, \quad k = 1, \dots, K, u, u' = 1, \dots, U, u \neq u' \quad (6.33d)$$

where ϖ is a penalty factor that determines the fairness of the solution. The constraint (6.33b) encodes the possible 2D movement of the drones and (6.33a) describes the start point of the trajectory. Constraint (6.33c) states that the directional speed is upper-bounded by v^{MAX} , while (6.33d) guarantees that throughout the mission a minimum security distance d^{MIN} is maintained among UAVs. Since the GUs positions are known, it is possible to derive the optimal patch phase shift through (6.31). Therefore, the problem can be rearranged in terms of scheduling, i.e, patch-GU assignment, by means of a matrix $\mathbf{S}_k = \{\mathbf{S}_{k,u,p} \forall k, u\} \in \{1, \dots, G\}^{U \times P}$. Let define an auxiliary function

$$\Theta : \mathbf{S}_k \rightarrow [\Phi_k^X, \Phi_k^Y], \quad (6.34)$$

which takes as input the scheduling matrix and returns the optimal phases vectors. Therefore, the expression of the data rate can be rearranged as $\bar{R}_{k,i,g} \triangleq R_{k,i,g}(\cdot, \Theta)$. The optimization problem is cast as

$$\max_{\mathbf{Q}, \mathbf{V}, \{\mathbf{S}_k\}} \frac{\sum_{k=1}^K \sum_{i=1}^I \sum_{g=1}^G \bar{R}_{k,i,g}}{1 + \varpi \sum_{g>g'} \left| \sum_{k=1}^K \sum_{i=1}^I \bar{R}_{k,i,g} - \bar{R}_{k,i,g'} \right|} \quad \text{s.t.} \quad (6.35)$$

$$(6.33a) - (6.33d)$$

Problem (6.35) is a MINLP problem and hence very challenging to solve. Common techniques as BCD and SCA cannot be applied in this case, since the channel model involves cosine interference patterns, which are not convexifiable. Therefore, in this work a DRL approach is employed to achieve a quasi-optimal solution.

6.4.2 Proposed Solution based on Reinforcement Learning

DRL is a branch of machine learning which involves an agent interacting via actions with an environment. In general, at each discrete time instant k the agent observes the current state s_k and, according to a policy $\pi(s_k|a_k)$, takes a certain action a_k . Consequently, it receives a reward r_k and moves to the next state s_{k+1} . The reward is then employed to update the existing policy in order to obtain better, or to avoid worst, actions. In practice, $\pi(s_k|a_k)$ maps each state to the probability of taking action a_k .

In this work, the problem (6.35) is solved to achieve a quasi-optimal solution, leveraging the well-known PPO approach [68]. To this aim, problem (6.35) is reformulated as a MDP, which is a mathematical framework at the foundation of DRL. First, to increase training speed and prevent divergence, the scheduling matrix is normalized such that $-1 \leq \mathbf{S}_{k,u,p} \leq 1$, as follows

$$\bar{\mathbf{S}}_{k,u,p} = \left\lfloor G \frac{\mathbf{S}_{k,u,p} + 1}{2} \right\rfloor. \quad (6.36)$$

Similarly, also the speed profile vectors $\mathbf{v}_{k,u}$ are normalized as $\bar{\mathbf{v}}_{k,u} = \frac{\mathbf{v}_{k,u}}{v^{\text{MAX}}}$. The reference MDP $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \varsigma \rangle$, is described by:

- *State Space*: The set of all the possible states $s_k \in \mathcal{S}$ that can be observed by the agent while interacting with the environment with

$$s_k = \{ \mathbf{q}_{k,1}^\top, \dots, \mathbf{q}_{1,U}^\top, \dots, \mathbf{S}_{k,1,1}, \dots, \mathbf{S}_{k,1,P}, \\ \dots, \mathbf{S}_{k,U,1}, \dots, \mathbf{S}_{k,U,P} \} \quad (6.37)$$

- *Action Space*: The set of all possible actions $a_k \in \mathcal{A}$ that the agent can perform during a timestep where

$$a_k = \{ \bar{\mathbf{v}}_{k,1}^\top, \dots, \bar{\mathbf{v}}_{k,U}^\top, \dots, \bar{\mathbf{S}}_{k,1,1}, \dots, \bar{\mathbf{S}}_{k,1,P}, \\ \dots, \bar{\mathbf{S}}_{k,U,1}, \dots, \bar{\mathbf{S}}_{k,U,P} \} \quad (6.38)$$

- *Transition Probability* \mathcal{P} : The set of probabilities which denote the transition from a state s_k to s_{k+1} .
- *Reward function* \mathcal{R} : The objective function to be maximized. In this case, it has been already defined in problem (6.35)
- *Discount factor* ς : The discount factor that determines the importance of future rewards, where $0 \leq \varsigma \leq 1$.

At the beginning, it is necessary to initialize the structures needed for the computation. In particular, the initial positions of the drones is set according to (6.33a). After the initialization phase, at each timeslot k , the agent observes from the environment the current positions of the drones \mathbf{Q}_k and the assignment of the patches \mathbf{S}_k . Then, the agent provides the speed profile \mathbf{V}_k and the new scheduling matrix $\bar{\mathbf{S}}_k$ in input to the algorithm that computes the new positions of the drones, according to (6.33b), and the data rate of each GUs. It is worth noting that constraint (6.33c) is inherently satisfied, since $\bar{\mathbf{v}}_{k,u}$ is renormalized to compute the new UAVs' positions. The final output of the computation is the new state (observation), the

generated reward, and a flag that notifies whether the mission has terminated or not. Clearly, whereas the mission ends before K , the cumulative reward would be lower; this mechanism is employed to satisfy constraint (6.33d).

The advantages brought by PPO approach are numerous. The most important ones are (i) ease of implementation, (ii) low complexity, (iii) sample efficiency, and (iv) few hyper-parameters are needed in the learning process. It ensures a smoother training, compared with other approaches, by constraining the new policy to not excessively differ from the previous one. This yields lower variance in the training process and prevents the agent from taking unrecoverable paths.

PPO uses an Actor-Critic approach, which employs two Deep Neural Networks. The former, namely Actor, receives the current state s_k and returns a distribution from which the action a_k is sampled. The latter, namely Critic, evaluates the action taken by the Actor (which led the environment in the next state s_{k+1}) and hence provides a rating. Based on this value, the Actor improves the current policy π , adopting a gradient descent algorithm, to take better actions in the future. Concretely, the aim of this framework is to derive the optimal policy π^* from the transition tuples $\langle s_k, a_k, r_k, s_{k+1} \rangle$ that maximize the discounted cumulative sum of all future rewards. For further details the reader is referred to [68].

6.5 Numerical Results and Discussion

In this Section a simulation campaign is carried out to investigate different aspects of the proposed model.

Symbol	Value	Symbol	Value
K	$\{1, 10\}$ [#]	G	$\{1, 2, 20\}$ [#]
U	$\{1, 3\}$ [#]	I	$\{1, 2, 20\}$ [#]
w	10^{-4} [m ²]	$P_{k,i,g}, \forall k, i, g$	20 [dBm]
$d^X = d^Y$	10^{-2} [m]	N_0	-174 [dBm/Hz]
f_c	10.5 [GHz]	κ^{MIN}	6 [dB]
δ_f	10 [kHz]	κ^{MAX}	10 [dB]
α^{BG}	$\{3, 4\}$ [#]	$a_{k,u,p_u}, \forall k, u, p$	1 [#]
δ_t	1 [s]	ϖ	5e-4, 1.25e-4 [#]
ε	0.01 [#]	d^{MIN}	10 [m]
v^{MAX}	30 [m/s]	ς	0.3 [#]

TABLE 6.2: Parameter set used in the simulations.

In particular, the analysis first focuses on the validation of the channel model from a probabilistic standpoint by comparing the derived approximation with Monte Carlo simulations. Then, the radiation pattern corresponding to different patch configurations is studied. Moreover, results are shown for the optimization problem defined in Section 6.4, to demonstrate the potential of the diversity introduced by the patches. Finally, the solution is compared with the case of absence of IRS-equipped drones and with a baseline approach implementing a random patch scheduling.

In all the simulations, all the involved antennas are isotropic, thus $F^{\text{BG}}(\cdot, \cdot) = F^{\text{BRG}}(\cdot, \cdot, \cdot, \cdot) = 1$, with unitary gains. All the considered IRS are composed by $N \times M = 48 \times 48$ elements. Moreover, according to [66], the cascaded channel BS-UAV-GU is characterized by $\beta^{\text{BRG}} = \frac{c^2 s}{64\pi^3 f_i^2}$ and $\alpha^{\text{BR}} = \alpha^{\text{RG}} = 2$, while the direct weak BS-GU link by $\beta^{\text{BG}} = \frac{c^2 s}{16\pi^2 f_i^2}$ and $\alpha^{\text{BG}} = 4$. Besides, it is assumed $\kappa^{\text{MIN}} = \kappa_g^{\text{MIN,BG}} = \kappa_{k,g,u}^{\text{MIN,RG}} = \kappa_{k,u}^{\text{MIN,BR}}$ and $\kappa^{\text{MAX}} = \kappa_g^{\text{MAX,BG}} = \kappa_{k,g,u}^{\text{MAX,RG}} = \kappa_{k,u}^{\text{MAX,BR}}$, for all k, g, u . The main considered parameters, unless otherwise specified, are summarized in Table 6.2.

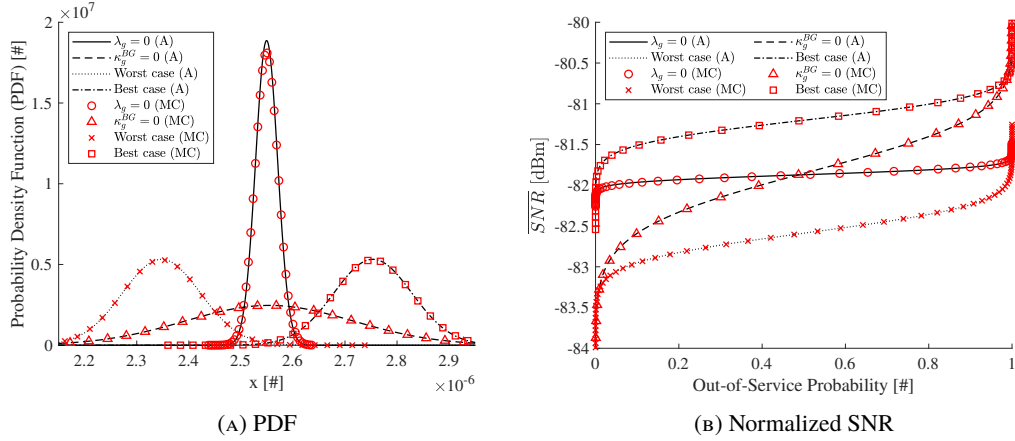
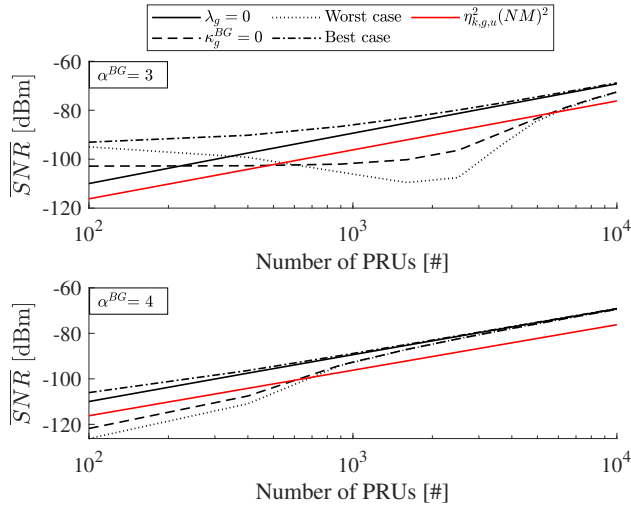


FIGURE 6.2: (A) approximation vs Monte Carlo (MC).

FIGURE 6.3: $\overline{\text{SNR}}$ with respect to number of PRUs.

6.5.1 Channel Model Analysis

Without loss of generality, the first scenario considers $G = 1$ GU, located at $\mathbf{q}_1^G = [50 \ 0 \ 0]^T$, served by a BS at $\mathbf{q}^{\text{BS}} = [-50 \ 0 \ 0]^T$. The communication is assisted by a drone, positioned at $\mathbf{q}_{1,1} = [0 \ 0 \ 50]^T$, equipped with a IRS horizontally split into $P = 2$ patches.

To provide a comprehensive analysis, four cases are considered: (i) absence of the direct BS-GU link ($\lambda_g = 0$), (ii) scattered (Rayleigh) BS-GU link ($\kappa_g^{\text{BG}} = 0$), (iii) worst possible UAV-BS alignment (Worst case), and (iv) best possible UAV-BS alignment (Best case). Define the normalized SNR $\forall k, i, g$, as

$$\overline{\text{SNR}}_{k,i,g} = \frac{\rho^2}{P_{k,i,g}} \text{SNR}_{k,i,g}.$$

Figure 6.2 represents the comparison between the proposed model and a Monte Carlo simulation for $5 \cdot 10^6$ realizations. As a matter of fact, the proposed model provides an accurate approximation in terms of PDF and $\overline{\text{SNR}}$ curves. Moreover, it can be observed that, even if the BS-GU link exists, the UAV-BS alignment is crucial: in the worst case scenario the channel quality results to be deteriorated with respect to the pure reflection case ($\lambda_g = 0$).

Figure 6.3 corroborates the theoretical results obtained in Corollary 4. Indeed, a large

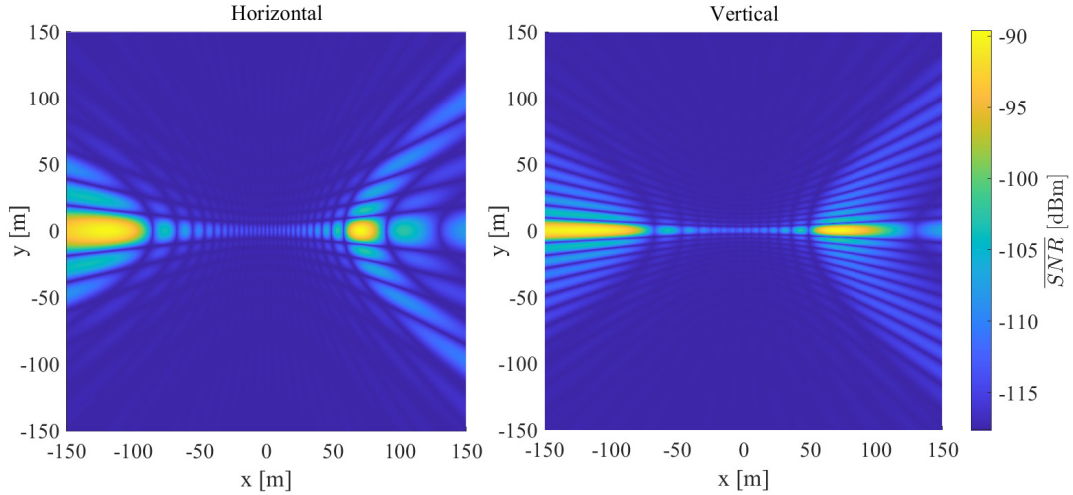


FIGURE 6.4: $\overline{\text{SNR}}$ for $P_u = 2$ patches divided horizontally (left) and vertically (right).

number of PRUs implies a quadratic scale of the $\overline{\text{SNR}}$. Clearly, in the four considered cases, the convergence depends on the involved system setup and environmental surrounding conditions. It is worth noting that a reduced path loss exponent $\alpha^{\text{BG}} = 3$ produces, in the Best case and for a low number of elements, an higher gain with respect to the pure reflection case ($\lambda_g = 0$). When a higher path loss coefficient is considered, i.e. $\alpha^{\text{BG}} = 4$, this gap significantly reduces since the direct link brings a minor contribution to the SNR.

The second scenario involves the same UAV and IRS, but considers $G = 2$ GUs located at $\mathbf{q}_1^G = [-120 \ 0 \ 0]^T$ and $\mathbf{q}_2^G = [70 \ 0 \ 0]^T$, connected to a BS at $\mathbf{q}^{\text{BS}} = [-50 \ 0 \ 0]^T$. Two cases are investigated. In the first one, the IRS surface is divided horizontally, while in the second one, vertically. Then, the phase shift matrices are set so that each user has its own patch assigned. From Figure 6.4 it clearly emerges that the produced radiation patterns yield a different $\overline{\text{SNR}}$ perceived in the environment in the two cases. Indeed, the side of the patch characterized by more elements leads to a narrower beam in the orthogonal direction and viceversa. This phenomenon, which can be observed especially in horizontally-divided case, is also influenced by the UAV-GU distance. Therefore, the shape design is particularly important since remarkably modifies the irradiated area, and has different implications based on the use case considered. For instance, if the objective is to maximize channel quality perceived by multiple GUs, the neighborhood of a illuminated user benefits from configurations that produces a larger fingerprint. At the same time, if the aim is to establish secure communications, patches must be designed on the opposite criteria, thus hinder eavesdropping. An agnostic option is to split the IRS into square-shaped patches that can be directed towards specific GUs, thus implying a customizable radiation pattern which guarantees, however, a computation complexity much lower than solution proposed in [65].

6.5.2 Results on trajectory and phase shift optimization

It is considered a scenario in which $U = 3$ drones, starting from $\mathbf{q}_{1,u} = [0 \ -150 \ 50]^T, \forall u$, are in charge of fairly serving $G = 20$ nodes randomly deployed on the ground. The BS is located at $\mathbf{q}^{\text{BS}} = [0 \ 0 \ 0]^T$. The Actor and Critic neural networks of the PPO algorithm are characterized by four hidden layers of 1024 neurons each, trained for $3 \cdot 10^5$ epochs.

First, it is considered the case in which each drone is equipped with a IRS split into $P_u = 4 \ \forall u$ patches. Moreover, two values for the fairness parameter ϖ are considered, i.e., $5 \cdot 10^{-4}$ and $1.25 \cdot 10^{-4}$. The first value penalizes the objective function more, thus obtaining a

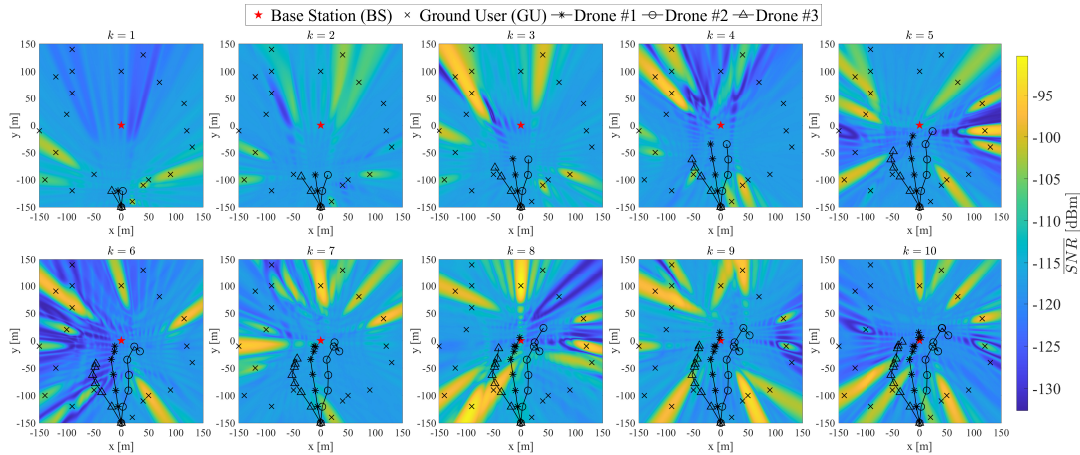


FIGURE 6.5: Drones' trajectories and normalized SNR for $G = 20$, $U = 3$, $P_u = 4$, and $\varpi = 5 \cdot 10^{-4}$.

higher fairness with respect to the second one, which implies a lower balance in the resources distribution. Figure 6.5 shows the trajectories of the drones during the mission, with heatmap representing the values of SNR. It is visible that, during the mission, the drones serve different regions at each instant, with the aim of maximizing the sum-rate while providing a fair allocation to the users. The swarm flies, as result of the optimization, from the starting location towards the center of the area, where the BS is located. Consequently, the patches of the different UAVs are allocated to provide an optimal coverage to all users, thus improving their SNRs.

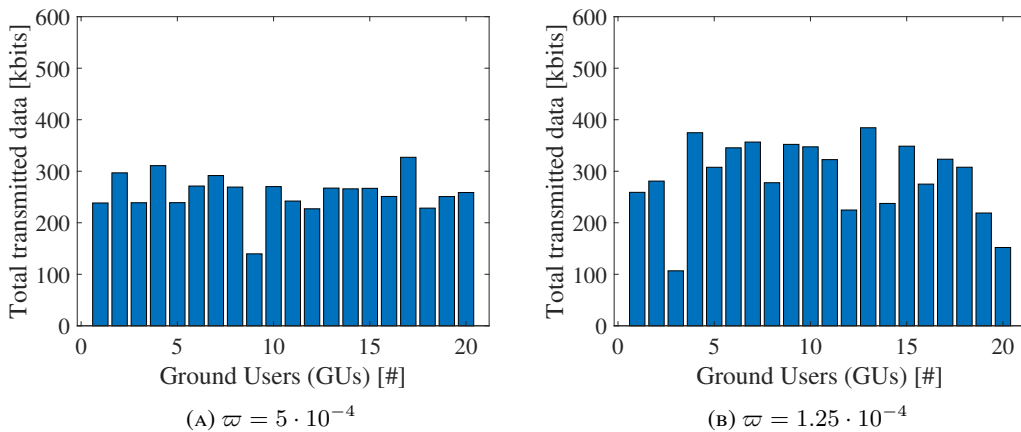


FIGURE 6.6: Total amount of data received by nodes, with $P_u = 4$.

In Figure 6.6, it is depicted the cumulative amount of data transmitted from each GUs to the BS through the direct link plus the reflected one. In case of higher fairness factor, i.e., $\varpi = 5 \cdot 10^{-4}$, the total amount of received data is 5.15 Mbits with a coefficient of variation (ratio between standard deviation and mean) of 0.1495. On the contrary, in the second case, i.e., $\varpi = 1.25 \cdot 10^{-4}$, a greater amount of data is received by the BS, i.e., 5.80 Mbits, but with a coefficient of variation equal to 0.1761 0.2538. This implies that a trade-off between fairness and data transmission exists: it is due to the fact that in case of lower fairness the drones can focus on reflecting the signals from the GU closer to the current position, which in turns implies that farther nodes are less served.

For the same scenario parameters, a simulation with $P_u = 9$ patches (for each drone) is carried out. As expected, Figure 6.7 confirms the same trade-off behavior highlighted in

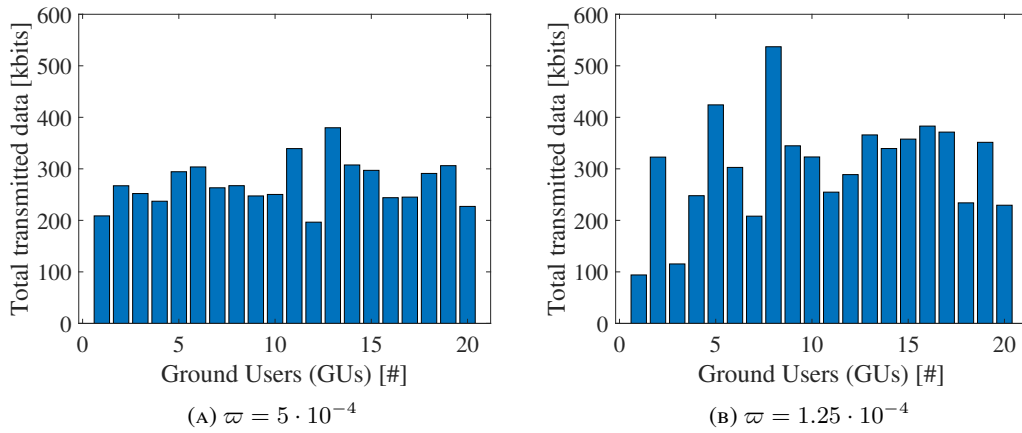


FIGURE 6.7: Total amount of data received by nodes, with $P_u = 9$.

the previous case. As a matter of fact, with $P_u = 9$ patches, in the first case a total of 5.42 Mbits have been transmitted with a coefficient of variation equal to 0.1629, while in second case the values are 6.09 Mbits and 0.3342, respectively. Finally, Figure 6.8 summarizes the

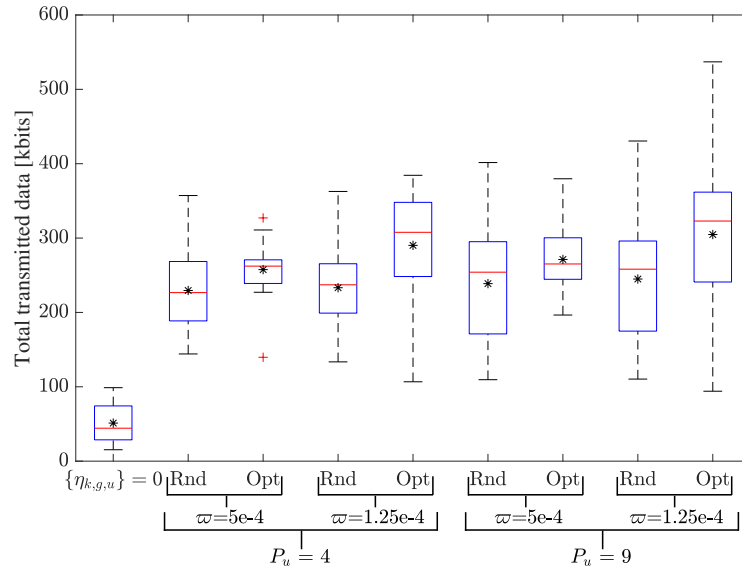


FIGURE 6.8: Comparison among data transmitted amounts for random (Rnd) and optimized (Opt) scheduling. The black stars represent the mean value.

results and the discussion above by comparing the obtained values with the case in which the swarm is absent and only the BS serves the GUs, i.e., $\{\eta_{k,g,u}\} = 0 \forall k, g, u$. Moreover, to provide a benchmark, the proposed solution is also compared with a baseline approach, in which the optimized trajectories are maintained but a random patch scheduling is implemented ($5 \cdot 10^6$ trials have been performed). As expected, in all cases, the drones provide a substantial enhancement of the channel capacity, especially when an optimized solution is employed. Indeed, the scheduling plan obtained from the PPO performs better than that related to the baseline approach. In particular, this is always true in terms of average data rate. Nonetheless, when a higher number of patches is employed a higher fairness factor is required to guarantee a uniform resource distribution.

Chapter 7

Internet of Drones Simulator: Design, Implementation, and Performance Evaluation

Networked drones can unleash disruptive scenarios in many application domains. At the same time, to really capitalize their potential, accurate modeling techniques are required to catch the fine details that characterize the features and limitations of UAVs, wireless communications, and networking protocols. To this end, the present Chapter discusses the IoD-Sim, a comprehensive and versatile Network Simulator 3 (ns-3)-based open-source simulator that addresses the many facets of the IoD. In order to prove the huge potential of this proposal, three different scenarios are presented and analyzed from both a software perspective and a telecommunication standpoint.

7.1 Introduction

The adoption of drones in industry is a huge commercial opportunity, as testified by the several billions forecasts already available for multiple business sectors [2].

Even though several applications are now including drones, and they may look like off-the-shelf utilities, the design of complex IoD systems still requires advanced methodologies to effectively unleash the potential of services based on networked drones. In this regard, channel capacity, available/required data rates, dedicated bandwidth and frequencies, must be characterized, bearing in mind that every link may be realized with a different telecommunication protocol. Moreover, given the variety of available drones on the market, an accurate suitability assessment based on their specifics is required.

Differently from the available IoD simulators [69]–[75], which do not cover all the aforementioned aspects, this work proposes a comprehensive open-source simulation platform, namely IoD-Sim* [76]. Since it was first released, it has been sensibly modified, partially re-written, and thoroughly refactored in order to create complex operating scenarios. The architecture is designed as a 3-layer stack: (i) the *Underlying Platform*, which includes a set of technologies and libraries able to perform high-precision numerical computation; (ii) the *Core*, which embeds a set of unique IoD-related features; (iii) the *Simulation Development Platform* that allows high-level mission design and analysis of simulation results. IoD-Sim is able to create realistic simulations by extending the available features of ns-3 to address the relevant aspects of the IoD, thus including mission design, trajectory planning, hardware and application configuration, mobile wireless communications, mobility and energy consumption models, on-board peripherals, and integration with other network entities.

To prove its potential and to validate its manifold functionalities, an extensive and diversified simulation campaign is carried out. Three different scenarios are conceived through the

*https://github.com/telematics-lab/IoD_Sim

proposed high-level mission design tool, which grants a welcoming user experience via a convenient interface. The different scenarios are characterized in terms of network topologies, communication technologies, drones' equipment, and software applications. In particular, signals experience different propagation conditions introduced by the adoption of channel models that vary from ideal conditions, i.e., free space, up to more realistic ones, i.e., densely populated urban environments. Lastly, simulation results are analyzed to obtain relevant Key Performance Indexs (KPIs), such as Signal-to-Interference-plus-Noise Ratio (SINR), throughput, power consumption, latency, and PLR, from which thoughtful insights are derived.

The present contribution is structured as follows: Section 7.2 presents a general overview of the architecture of the simulator. Section 7.3 describes the underlying platform and the rationale for its choice. Section 7.4 discusses the core of the simulator in detail, with dedicated subsections about the main building blocks of the project. A thorough explanation of the involved mobility models is given together with all the supported communication technologies, and the involved logical entities. Section 7.5 focuses on the simulation design, thus explaining the role and importance of scenario configurations. Section 7.6 is dedicated to the simulation campaign; after an initial focus on scenarios description, the outcomes are discussed to highlight the main findings.

7.2 Architectural Overview

The architecture of IoD-Sim (see Figure 7.1) is organized into three parts: (i) the *Underlying Platform*, that provides the necessary networking components and performs advanced mathematical and parsing operations, (ii) the *Core* of the simulator, which implements the foundation of IoD-related features, and (iii) the so called *Simulation Development Platform*, a high-level component which allows to develop, configure, and analyze advanced scenarios. Each part is modular by design and proposes peculiar functionalities that are depicted as

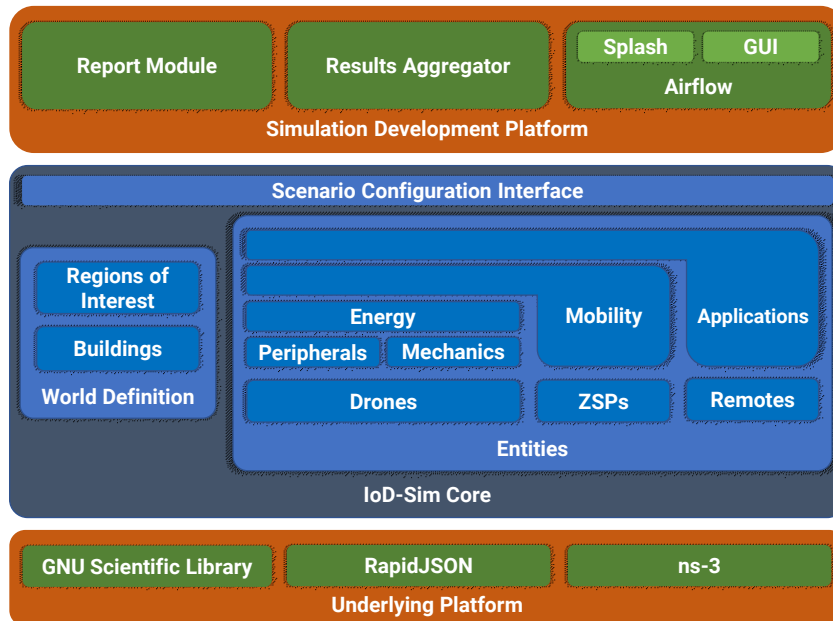


FIGURE 7.1: Overview of IoD-Sim Architecture.

blocks and described in what follows. The joint adoption of these components enables different simulation scenarios, starting from a highly flexible and general purpose one, which is configurable by higher-level entities.

In particular, the IoD-Sim core, as will be discussed in Section 7.4, details all the aspects related to IoD entities, especially drones and ZSPs, spanning from their mechanics and peripherals to remote information services. Moreover, the whole simulated environment is described by a set of elements, such as specific areas of interest, as well as the presence of buildings in the reference scenario.

With reference to the Simulation Development Platform part of the architecture, it is worth specifying that IoD-Sim includes Airflow, a high-level visual configuration environment that drastically eases the interaction between the user and the simulator, i.e., scenario set-up and management.

Finally, a report module, that will be analyzed in Section 7.5, guarantees the readability of simulation results in a clear XML schema, which eases data processing. This eases the integration of the IoD-Sim with external, e.g., third-party, tools specifically designed to create scenario configurations by customising both parameters (e.g., mechanical properties of the drones) and characteristics (e.g., employed communication technologies).

7.3 Underlying Platform

The *Underlying Platform* is a foundation composed by the GNU Scientific Library (GSL), *RapidJSON*, and ns-3.

GSL is a numerical computing framework which implements numerous routines and low-level data structures, such as complex numbers, linear algebra, data analysis, and interpolation. Furthermore, it is offered in Linux-derived distributions with first-class support [77].

RapidJSON is a parser and generator of JavaScript Object Notation (JSON) code. It is one of the most adopted JSON libraries available for C++ projects. It eases the creation, traversal, validity check, and analysis of JSON codes [78]. *RapidJSON* has been chosen for its high performance and its extensive and flexible high-level Application Programming Interfaces (APIs).

Finally, ns-3 emerges as the most relevant component: it is a solid and mature discrete-time event-based network simulator. ns-3 is an open-source project that provides a solid simulation engine and various models for network design and testing. IoD-Sim treats ns-3 as a foundation, extending it with new features that are focused on accurate drone simulations, mobile wireless communications, energy consumption, and their integration with on-board peripherals and ground communication infrastructures.

7.4 Core of IoD-Sim

This Section presents the building blocks of the IoD-Sim Core, which is the main part of the simulator.

7.4.1 World Definition

IoD-Sim offers the possibility to define parameters related to the simulated world, i.e., the environment in which the simulation takes place. The two main features are:

- *Buildings* which are collisionless 3D objects useful to represent urban scenarios, thus making simulations that are particularly suitable for research in *Smart Cities*. Such a feature is important for what concern LTE communication fading, which varies according to the characteristics of each building.

- `ns3::InterestRegion` is a 3D box placed on the simulated world defined, as for buildings, by a vector of two points. Through specific methods, it is possible to acknowledge the presence of a drone in multiple areas, thus granting the possibility to trigger specific events during the simulation. For instance, *Drone* operations can be restricted to a limited space, leading to an optimization of *Drone* power consumption.

The virtual world in IoD-Sim is a theoretically infinite space. The space can be filled with entities, which could be Drones, ZSPs and Remotes, but also with RoIs and Buildings.

7.4.2 Drones

IoD-Sim provides `ns3::Node` class derivatives to consider the characteristics of key actors commonly found in an IoD simulation. The `ns3::Drone` class characterizes a rotary-wing UAV, along with its mechanical properties, shown in Table 7.1. While the first four properties can be defined by the user, the last two are a direct consequence of the given characterization.

TABLE 7.1: `ns3::Drone` properties in IoD-Sim.

Name	Unit of Measurement
Mass	kg
Rotor Disk Area	m ²
Drag Coefficient	(dimensionless)
Peripherals	
Weight Force	N
Air Density	kg/m ³

A UAV is usually equipped with a set of peripherals able to extend its capabilities. Such peripherals include a wide range of devices, implemented in IoD-Sim, through new specific classes. The `ns3::DronePeripheral` object represents a general-purpose on-board peripheral characterized by (i) *Peripheral state*, which can either be set to ON, OFF, or IDLE; (ii) *Power consumption* in watts; (iii) *Reference Region of Interests (RoIs)* which specifies where the peripheral should be operating. `ns3::DronePeripheral` has been specialized in two subclasses:

- `ns3::StoragePeripheral` represents a generic storage device characterized by an attribute describing the initial amount of memory, which can be traced at runtime to record the empty space left.
- `ns3::InputPeripheral` describes a generic input device, characterized by an acquisition `DataRate`, constant over a `DataAcquisitionTimeInterval`.

These two peripheral types are strongly connected, since a `ns3::InputPeripheral` can offload acquired data to a `ns3::StoragePeripheral` through a boolean attribute. Nonetheless, the association between input and storage is not mandatory. In fact, in a real-world

TABLE 7.2: Drone Peripherals Properties.

Class	Attribute	Description
<code>DronePeripheral</code>	<code>PowerConsumption</code>	Power consumption of the peripheral in J/s
<code>StoragePeripheral</code>	<code>Capacity</code>	The capacity of the disk in bit
	<code>DataRate</code>	The acquisition data rate of the peripheral in bit
	<code>InitialRemainingCapacity</code>	The starting remaining capacity in bit
<code>InputPeripheral</code>	<code>DataAcquisitionTimeInterval</code>	The time interval occurring between any data acquisition
	<code>HasStorage</code>	Acquired data are offloaded to the <code>StoragePeripheral</code>

scenario, an `ns3::InputPeripheral` can deliver data directly to a processing unit or to a remote host, thus neglecting the need to permanently store the information.

A complete list of the attributes of these classes is given with Table 7.2. It is worth specifying that all peripherals hold a reference to the drone they are equipped to.

For what concerns the energy-related aspects, ns-3 already models and manages consumption, harvesting, and monitoring, through the abstract class `ns3::EnergySource`. Although there is no specific energy source model available that is suitable for drones, the `ns3::LiIonEnergySource` is sufficiently general to be employed for simulation purposes [79], [80].

In this work a specific class, i.e. `ns3::DroneEnergyModel`, is developed to compute the drone mechanical power consumption. Given a simulation duration T split into $n = 1, \dots, N$ equal discrete intervals, the power consumption of the drone flying at speed $\mathbf{v}[n] = (v_x[n], v_y[n], v_z[n])$, is [15]:

$$P_{UAV}[n] = P_{level}[n] + P_{vertical}[n] + P_{drag}[n], \quad (7.1)$$

where

$$P_{level}[n] = \frac{W^2}{\sqrt{2}\rho A} \frac{1}{\sqrt{\Omega + \sqrt{\Omega^2 + 4V_h^4}}}, \quad (7.2)$$

being

$$\Omega = \|(v_x[n], v_y[n])\|^2 \quad (7.3)$$

$$P_{vertical}[n] = Wv_z[n], \quad (7.4)$$

$$P_{drag}[n] = \frac{1}{8}C_{D0}\rho A\|(v_x[n], v_y[n])\|^3, \quad (7.5)$$

$W = mg$, with m defining the mass of the drone and g as the gravitational acceleration. Moreover, ρ is the air density, A is the total rotor disk area, C_{D0} is the profile drag coefficient depending on the geometry of the rotor blades, and $V_h = \sqrt{\frac{W}{2\rho A}}$ uses parameters to calculate the power required for hovering operations.

The energy model can be aggregated to a drone by means of the `ns3::DroneEnergyModelHelper`, which provides an `Install()` method that aggregates it to `ns3::Drone`. In this way, it is possible to simulate the energy characteristics of a drone, both for its mechanics and its peripherals, in addition to its networking operations.

7.4.3 Other Simulation Entities: ZSPs and Remotes

Entities beyond `ns3::Drone` are *ZSPs* and *Remotes*. *ZSPs* are smart entities, modeled as `ns3::Node` objects, equipped with multiple `ns3::NetDevice` which provide multi-protocol radio access, thus enabling communications between drones and the rest of the Internet. Typically, they are configured as ground entities that maintain a constant position in time [1], by means of `ns3::ConstantPositionMobilityModel`. Nonetheless, in IoD-Sim their mobility model can be customized to fit simulation purposes, envisioning the adoption of dynamic wireless infrastructure proposed in 5G & Beyond architectures. *Remotes*, instead, are `ns3::Node` objects with no mobility model and only rely on installed applications which provide remote services to consumers. *Remotes* and *ZSPs* are interconnected through a backbone, simplified as a Carrier Sense Multiple Access (CSMA)-based bus network, that represents the *Internet*. This architecture allows service provisioning on different classes of nodes,

employing *Remotes* in case of applications with high computational costs, e.g., multimedia data processing, and *ZSPs* in case of low latency requirements, e.g., traffic management.

7.4.4 Mobility

ns-3 provides a basic foundation to represent the movement of drones. However, an important gap arises when such models are analysed in details: none of the available ones are able to construct a curve trajectory that take into account how much a spot is relevant.

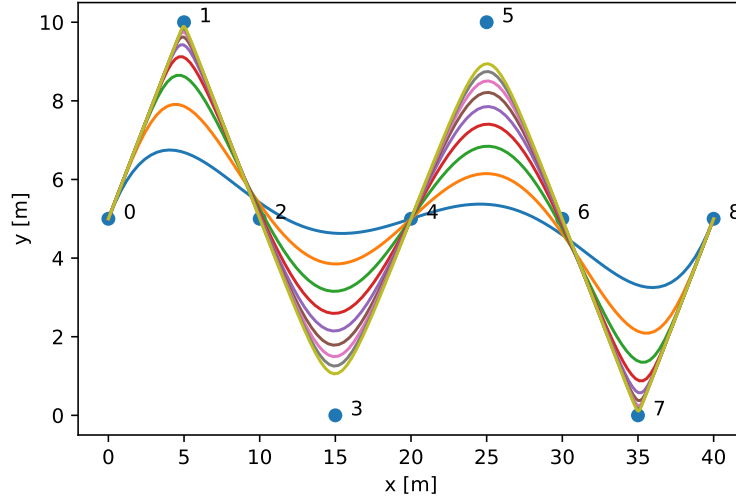


FIGURE 7.2: A set of trajectories, generated with (7.6), with different Interest Levels (from 1 to 10, incrementally) for PpoI 1, 3, 5, and 7. The other points have constant Interest Level set to 1.

To overcome these limitations, dedicated mobility models have been developed. In particular, the trajectory has been modeled using Bézier curves by specifying a set of PpoI.

Let $\mathbf{P} = \{\mathbf{P}_0, \mathbf{P}_1, \dots, \mathbf{P}_{N-1}\}$ with $\mathbf{P}_i \in \mathbb{R}^3$, $\forall i = 0, \dots, N-1$ be an ordered sequence of N interest points, $\mathbf{l} = \{l_0, l_1, \dots, l_{N-1}\}$, $l_i \in \mathbb{N}^+$, the interest level associated to each point, $\Lambda = \left(\sum_{i=0}^{N-1} l_i\right) - 1$ and $L_i = \sum_{h=0}^{i-1} l_h$. The *Trajectory Generator* can be expressed as

$$\mathbf{G}(t) = \sum_{i=0}^{N-1} \mathbf{P}_i \sum_{j=0}^{l_i-1} \binom{\Lambda}{L_i + j} (1-t)^{\Lambda-L_i-j} t^{L_i+j}, \quad t \in [0, 1] \quad (7.6)$$

An increment in the interest level l produces a trajectory that passes closer to that point, as illustrated in Figure 7.2, without reaching it, except for the first and last one.

Finally, the obtained trajectory is used by the new implemented models:

- `ns3::ConstantAccelerationDroneMobilityModel` employs (7.6) and the uniform acceleration motion law to retrieve the points of the desired trajectory. After the maximum speed is reached, the uniform linear motion law is adopted.
- `ns3::ParametricSpeedDroneMobilityModel`: takes a $v(t)$ speed profile in a polynomial form and, thanks to the modified Bézier equation (7.6), it retrieves the discretized trajectory.

A summary of the attributes of these mobility model is reported in Tables 7.3 and 7.4.

TABLE 7.3: ns3::ConstantAccelerationDrone MobilityModel TypeId attributes.

Attribute	Description
Acceleration	Drone's constant acceleration, expressed in m/s^2 .
MaxSpeed	Drone's maximum speed, expressed in m/s.
FlightPlan	Interest points for the trajectory.
SimulationDuration	Simulation duration, expressed in seconds.
CurveStep	Discretization step of the curve.

TABLE 7.4: ns3::ParametricSpeedDrone MobilityModel TypeId attributes.

Attribute	Description
SpeedCoefficients	The set of coefficients for the polynomial $v(t)$.
FlightPlan	Interest points of the trajectory.
SimulationDuration	Simulation duration, expressed in seconds.
CurveStep	Discretization step of the curve.

7.4.5 Applications

IoD-Sim offers simple applications that can be used to communicate telemetry from a drone to a ZSP or to a *Remote* by adopting client-server paradigm, via User Datagram Protocol (UDP). These applications are modeled as classes named ns3::DroneClientApplication and ns3::DroneServerApplication. When the application is started, a UDP-based communication takes place. The process starts with the client application in NEW state. Therefore, a HELLO packet is sent to the destination address (or in broadcast), thus implying a state transition in HELLO_SENT. If the application server receives such packet, it replies with an HELLO_ACK packet to confirm the reception. When the client receives the acknowledgement, its state changes again, into CONNECTED, which allows it to periodically send JSON-encoded telemetry data. These packets are named UPDATE and UPDATE_ACK. The entire procedure is illustrated in Figure 7.3 and the configuration parameters are summarized in Table 7.5.

Relying on the same architecture, two Transmission Control Protocol (TCP)-based applications are also available to enable reliable data transfer between hosts, which are implemented as TcpPeriodicClientApplication and TcpEchoServerApplication classes, respectively. The aim is to transfer a certain amount of information between the two hosts according to the specified PayloadSize, expressed in bytes, and TransmissionFrequency, measured in Hz, set on the client. The server is characterized by a socket, composed by a

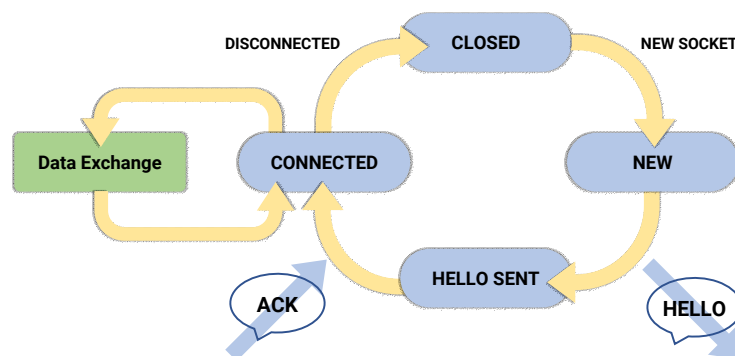


FIGURE 7.3: FSM of the Drone Client and Server Application.

TABLE 7.5: Configuration parameters for *Telemetry Applications*.

Application Type	Name	Type	Description
Client	<code>DestinationIpv4Address</code>	String	IPv4 address of the remote application server.
Client and Server	<code>Port</code>	UInt32	Port of the remote listening port.
Client	<code>TransmissionInterval</code>	Double	Transmission interval of the telemetry updates.
Client and Server	<code>StartTime</code>	Double	Time at which to start the application.
Client and Server	<code>StopTime</code>	Double	Time at which to stop the application, in seconds.
Client	<code>FreeData</code>	Boolean	Free data from the equipped storage peripheral.
Server	<code>StoreData</code>	Boolean	Store data to the equipped storage peripheral.

TABLE 7.6: Configuration parameters for *Generic Traffic Applications*.

Application Type	Name	Type	Default Value	Description
All Server and Clients	<code>Address</code>	String	127.0.0.1	Listening or remote address of the server.
All Server and Clients	<code>Port</code>	UInt16	4242	Listening or remote port of the server.
All Clients	<code>PayloadSize</code>	UInt16	65470	Size of the payload for each packet, in bytes. In case of Storage Client, it is the maximum size to be used when freeing storage memory.
Periodic Client only	<code>Frequency</code>	Double	1.0	Number of times in a second when a new packet is sent to the server.

listening Address and Port. These configuration parameters are summarized in Table 7.6.

Clearly, all these applications are developed so that multiple instances can run concurrently, on the same entity, if different ports are specified. Moreover, they are independent from the particular communication technology adopted.

Finally, a Network Address Translation (NAT)-like application is provided to design relaying network architectures, implemented through the class `ns3::NatApplication`. The NAT forwarding behavior leverages a hash map, i.e., *NAT Table*, where an external port number is coupled with the source IP address and port. Inbound frames are forwarded to the external network by replacing this information with the one of the relaying drone. The same rationale is applied for frames received from the external network.

7.4.6 Scenario Configuration Interface

The *Scenario Configuration Interface* is an abstraction layer that allows the configuration of the entire simulation by means of a dedicated JSON configuration file, which makes the entire scenario parametric. An example of JSON configuration file that realizes a simple scenario is shown in Figure 7.4.

The core of this abstraction layer is the `ns3::ScenarioConfigurationHelper`, a low-level object that directly deals with the JSON file. This helper returns a set of specific data classes that contain exclusively the parameters required to configure IoD-Sim models. Each of them is also loosely coupled with a JSON validator and parser, also known as configuration helper. The output data classes are then used by the *General Purpose Scenario*, which has been developed in order to setup scenario's entities and, at the same time, to provide abstractions which minimize the effort from a programming perspective. It is fully dependent on a semantic analyzer and allows the entire simulation platform to be compiled beforehand, providing ways to dynamically reconfigure the scenario at run-time. Its development started from the analysis and the detection of a common structure typically followed by the Open Systems Interconnection (OSI) protocol stack. Specifically, once the file is decoded, the number of entities are retrieved to create the initial structures. The entire configuration workflow followed by the *General Purpose Scenario*, depicted in Figure 7.5, is described hereby. After entity creation, the ns-3 static configuration parameters are applied to the simulation and World definition is made. Then, global communication stacks are linked to the configured

```

{
  "name": "iod_sim_ftw",
  "resultsPath": "../results/",
  "logOnFile": true,
  "duration": 50,
  "staticNs3Config": [...],
  "world": {
    "buildings": [
      {
        "type": "residential",
        "walls": "concreteWithoutWindows",
        "boundaries": [0.0, 70.0, 0.0, 70.0, 0.0, 20.0],
        "floors": 6,
        "rooms": [2, 1]
      }
    ],
    "RegionsOfInterest": [
      [170.0, 340.0, 180.0, 250.0, 0.0, 15.0]
    ]
  },
  "phyLayer": [
    {
      "type": "lte",
      "channel": {
        "propagationLossModel": {...},
        "spectrumModel": {...}
      }
    }
  ],
  "macLayer": [...],
  "networkLayer": [
    {
      "type": "ipv4",
      "address": "10.1.0.0",
      "mask": "255.255.255.0",
      "gateway": "10.1.0.1"
    }
  ],
  "drones": [
    {
      "netDevices": [
        {
          "type": "lte",
          "networkLayer": 0,
          "role": "UE",
          "bearers": [...]
        }
      ],
      "mobilityModel": {...},
      "applications": [
        {
          "name": "ns3::DroneClientApplication",
          "attributes": [...]
        }
      ],
      "mechanics": {
        "name": "ns3::Drone",
        "attributes": [...]
      },
      "battery": {
        "name": "ns3::LiIonEnergySource",
        "attributes": [...]
      },
      "peripherals": [
        {
          "name": "ns3::DronePeripheral",
          "attributes": [...]
        }
      ]
    }
  ],
  "ZSPs": [
    {
      "netDevices": [...],
      "mobilityModel": {...},
      "applications": [...]
    }
  ],
  "remotes": [
    {
      "networkLayer": 0,
      "applications": [
        {
          "name": "ns3::DroneServerApplication",
          "attributes": [...]
        }
      ]
    }
  ],
  "logComponents": [...]
}

```

FIGURE 7.4: An excerpt of scenario configuration with an overlay of the models associated to the analyzed parts.

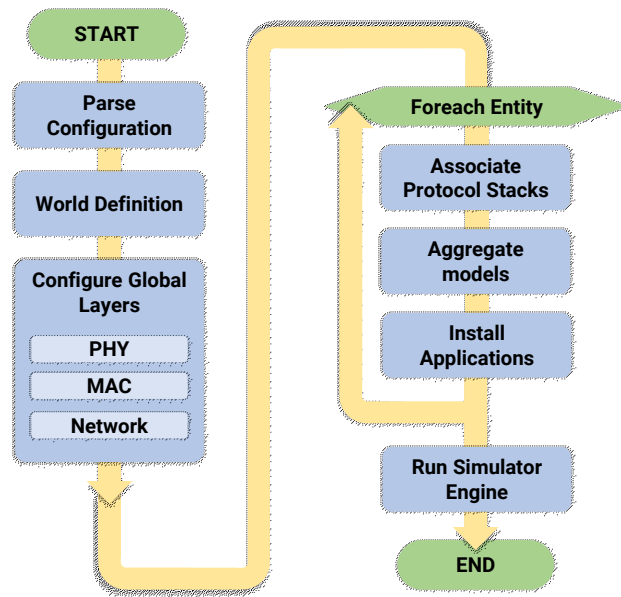


FIGURE 7.5: Logical flow to initialise and configure a scenario in IoD Sim.

entities. When the entity network configuration is done, the mobility models are configured and the applications are installed. Furthermore, if the entity is a *Drone*, its peripherals are installed, together with the associated energy model. Once all entities are ready, the virtual internet backbone is configured.

7.5 Simulation Development Platform

Simulation Development Platform, whose schema is illustrated in Figure 7.6, provides a set of extensions, output files for data analysis, and standalone applications for scenarios design. These tools ease scenario design and analysis, thus ensuring that IoD-Sim can be easily introduced to newcomers, especially university students and researchers. It is mainly composed by the *Report Module*, the *Results Aggregator* and *Airflow*.

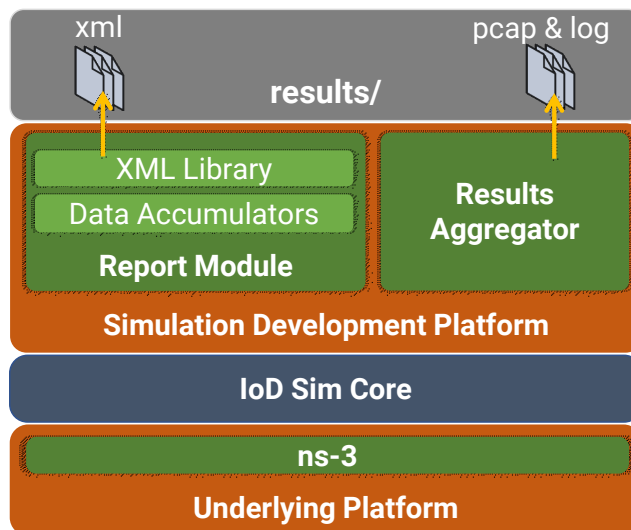


FIGURE 7.6: *Simulation Development Platform* high-level schema.

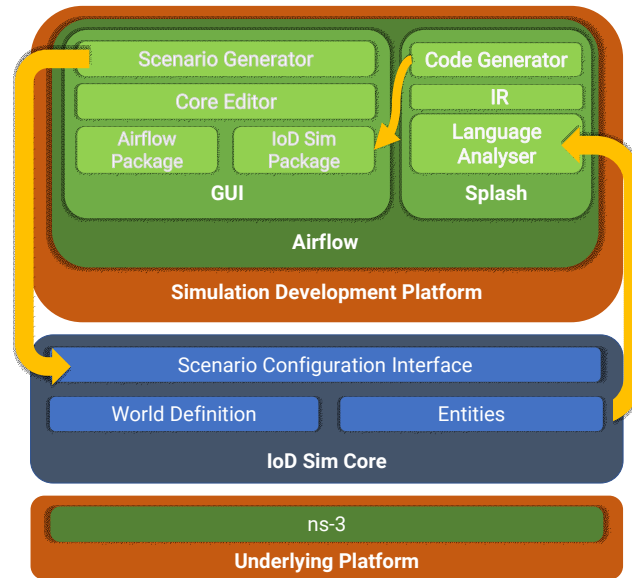


FIGURE 7.7: Airflow Architectural Design.

The *Report Module* is an extension of IoD-Sim which stores data at run-time and elaborates, at the end of simulation, a comprehensive summary. The aim of the extension is to introspect simulator's data structures to gather relevant data to be reported (e.g., data traffic, trajectory, and telemetry). To provide a final report that is both human and machine readable, the Extensible Markup Language (XML) format has been chosen. Therefore, a schema is defined to describe the expected structure of the produced file. This output XML file is put together with other files relevant to the simulation in the `results` directory.

The *Results Aggregator* gathers all the relevant information and debug messages of simulation internal components. Primarily, the *General Purpose Scenario* emits `progress.log` and `IoD Sim.log` files. The former is the output of the progress information messages that are also delivered on the standard output during scenario execution. The latter contains all debug messages coming from different internal components of IoD-Sim. The log components can be enabled by specifying them in the `logComponents` field of the scenario configuration JSON file.

Airflow is a high-level abstraction tool that gives visual clues during simulation design, thus enriching the user experience especially for newcomers. It has been developed on top of Splash, a specialized transpiler for IoD-Sim. It scans the source code of the simulator and outputs visual blocks that can be referenced in the *Core Editor* to configure a scenario. Thanks to the Graphical User Interface (GUI) editor, a scenario can be exported into a JSON file that can be interpreted by IoD-Sim *Scenario Configuration Interface*. From a software design standpoint, as illustrated in Figure 7.7, the Airflow project is entirely decoupled from IoD-Sim. Its integration with the simulator relies on interfaces that enable bidirectional communications. The Airflow GUI, shown in Figure 7.8, is based on the open source Ryven[†] engine, which is a dynamic runtime, *flow-based* visual programming environment for Python scripts. It offers: (i) a central rendering view to place blocks and link them together, (ii) a settings area to customize options, (iii) a variable management section to include and store data that can be integrated with the flow, and (iv) a console to report errors. Ryven includes additional features to optionally debug internal routines with the help of console messages. Moreover, thanks to its modular design, it allows blocks generated by Splash to be aggregated

[†]<https://ryven.org/>

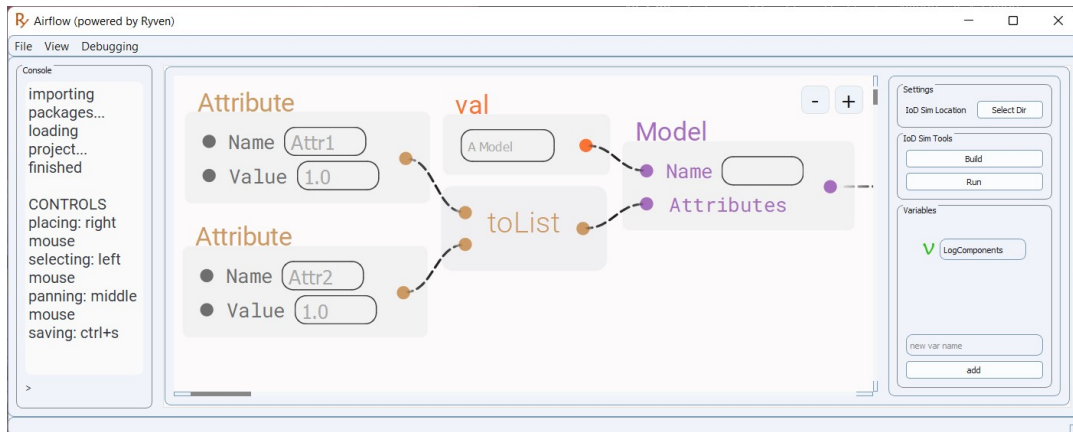


FIGURE 7.8: An overview of the configuration of a generic model in Airflow.

into *packages*. Ryven has been deeply extended to inter-operate with IoD-Sim, especially for its compatibility with the *Scenario Configuration Interface*.

The user interface is organized into the following components:

1. A menu bar at the top of the GUI window.
2. A *Console* on the left in order to monitor errors and messages coming from Airflow or IoD-Sim. Informative messages are reported in blue, while errors are displayed in red.
3. A central workspace to design the scenario by placing blocks and connecting them together.
4. A settings panel on the right.

The central workspace is the canvas where blocks and links are placed by the user to design a scenario. A block, as depicted in Figure 7.8, consists of a set of inputs and outputs. Each input and output can be connected to other outputs and inputs of other blocks, in order to create a tree. Moreover, in the settings panel, it is possible to run the scenario by clicking on the **Build** and **Run** buttons. Finally, a variable manager can be used to create, store, and later reference values by their respective labels on the workspace. This allows to reduce redundancy and to make the block tree more compact.

7.6 Simulation Campaign

This Section demonstrates the huge potential of IoD-Sim by means of an extensive simulation campaign which investigates the many facets of IoD scenarios. Firstly, the discussion explains how the simulation can be designed. Secondly, three different scenarios with increasing complexity are presented.

In particular, the first scenario discusses the use-case of telemetry with few drones flying in a RoI, which follow customized trajectories while gathering data. The purpose of this scenario is to demonstrate that it is possible to monitor one or more variables with on-board sensors, while estimating the energy consumption associated with flight dynamics.

The second scenario has a wider perspective, since it focuses on surveying and monitoring activities, further completed with the acquisition of multimedia signals by each drone. The possible applications include several real-world use cases in the fields of civil engineering, smart agriculture, or in environmental monitoring, e.g., coastal erosion and other slow phenomena. In fact, in this scenario, drones are on a mission in neighboring areas, since it

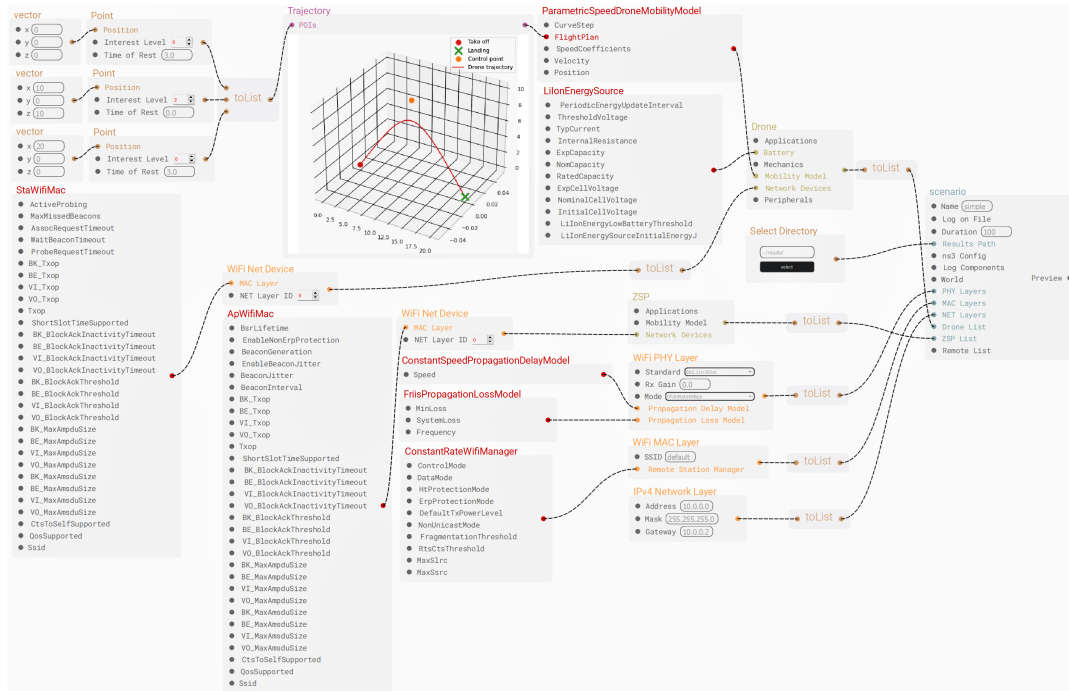


FIGURE 7.9: A simple scenario with one drone and ZSP designed from scratch in Airflow.

is assumed that the information of interest need to be contextualized, i.e., must be gathered at the same time. Furthermore, this case investigates the possibilities enabled by different data storage capabilities of by drones. Also, the offloading functionality, of the acquired data, avoids the overload/saturation of onboard available resources. Once data is gathered, they can be involved in offline post-processing, evaluation, and analysis.

The third scenario has been specifically designed to be the reference benchmark for IoD applications. It is settled in the context of smart cities, and it involves clusters of low-power Internet of Things (IoT) sensors. This scenario models real-world applications and, hence, shadowing and pathloss phenomena are included, thanks to the adoption of propagation models that are influenced by the presence of buildings. In order to guarantee a reliable communication, drones are in charge of relaying traffic to ensure coverage to all sensors in the city.

7.6.1 Scenario Design

Airflow represents the foremost application for visual scenario development. To better understand how to design simulations, a simple configuration setup is provided hereby. The envisioned scenario considers a drone that follows an arc-like trajectory and communicates telemetry to a ZSP by means of Wi-Fi. Specifically, the drone acts as a *station* and the ZSP as an *access point*. The entire configuration is depicted in Figure 7.9, where all the visual components, encompassed in the Airflow workspace, are properly set up and linked together. Starting from the right, the block Scenario glues some configuration input values, e.g. Name and Duration, with more complex components, such as (i) PHY/MAC/NET Layers, (ii) Drone List, and (iii) ZSP List.

In particular, the communication layers are configured to implement the Wi-Fi stack. The WiFi PHY Layer object defines the PHY layer to be used with particular propagation and loss models. The WiFi MAC Layer, instead, specifies the Service Set Identifier (SSID) of

the network and the Wi-Fi Manager object that handles MAC control plane. Further, the IPv4 Network Layer determines the address and mask of the overlying network.

Both Drone List and ZSP List properties are connected to the simulated entities, namely Drone and ZSP. These two components share different properties such as Applications, Mobility Model and Network Devices. However, the Drone block is characterized also by its unique features, i.e., Peripherals, Mechanics, and Battery. In this configuration, the ConstantPositionMobilityModel allows to place the ZSP to a fixed location, while the ParametricSpeedMobilityModel is employed to define the drone trajectory. In this regard, the Trajectory component, linked to the FlightPlan property of the mobility model, facilitates the design of the desired path.

Both drone and ZSP's Network Devices property is linked to a WiFi Net Device block. While StaWifiMac characterizes the device of the former, ApWifiMac is associated to the latter. Finally, a LiIonEnergySource defines the power supply of the drone.

The development strategy discussed above represents the common ground for the design of the following three scenarios.

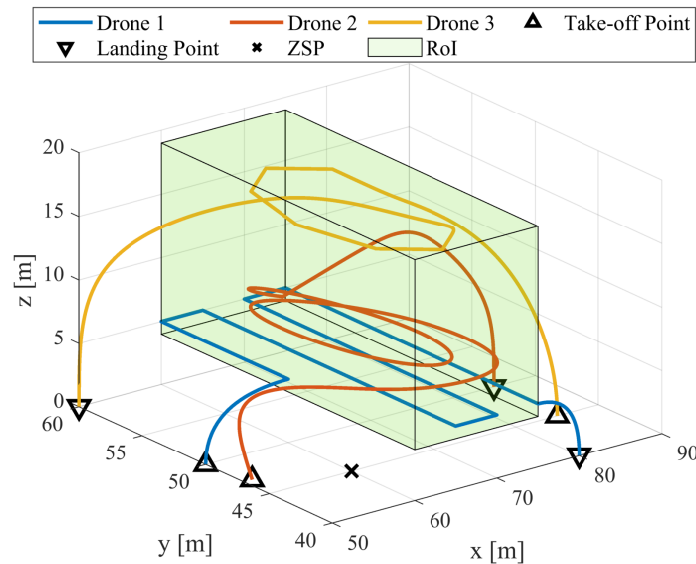


FIGURE 7.10: Scenario #1.

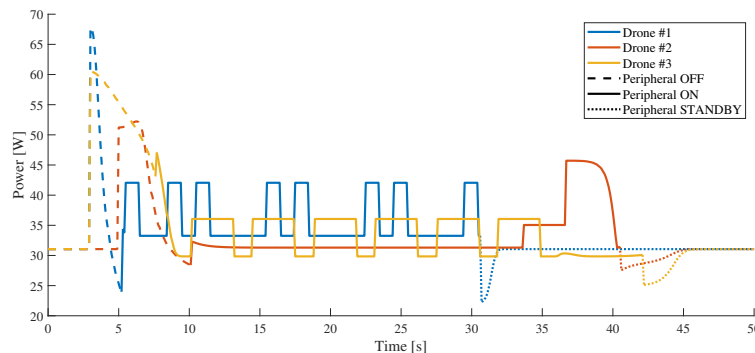


FIGURE 7.11: Power consumption and peripheral state for each drone, in the first scenario.

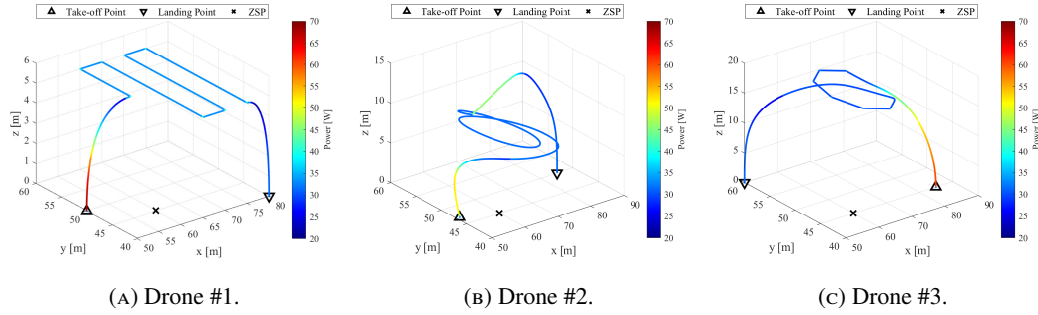


FIGURE 7.12: Drones' trajectories with their power consumption, in the first scenario.

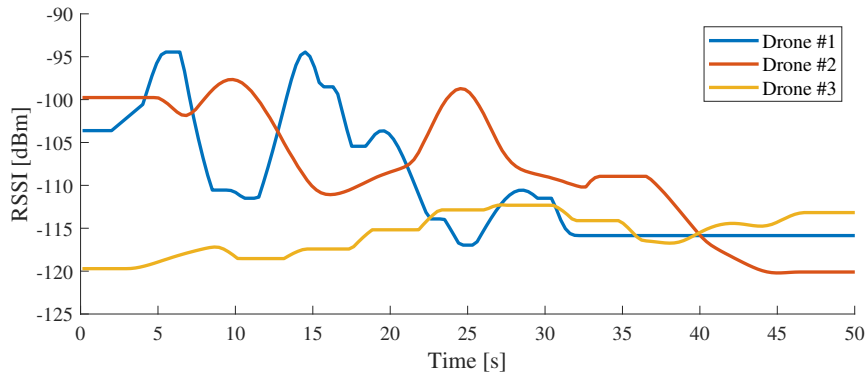


FIGURE 7.13: RSSI of each drone in the first scenario.

7.6.2 Scenario #1 - Telemetry

The first scenario, as depicted in Figure 7.10, envisions three drones with the same mechanical characteristics, all equipped with an Inertial Measurement Unit (IMU). In this scenario, drones are flying in the same RoI, at a constant speed, following different trajectories. Moreover, a ZSP is deployed on the ground. The latter is released in $[60 \ 45]^T$, which continuously monitors drones' operations by acquiring telemetry through Wi-Fi.

UAVs' trajectories are based on the ParametricSpeedDroneMobilityModel, which is configured to guarantee a constant speed of 5 m/s, 3 m/s, and 4 m/s, respectively. They are also equipped with IMUs which are generic drone peripherals that provide basic telemetry data to the ZSP thanks to a dedicated application. It is worth specifying that drones' IMUs have different power consumption, i.e., 12 W, 5 W, and 6 W.

The outcome of the simulation is hereby discussed. Figures 7.11 and 7.12 depict the power consumption trend with respect to time and trajectories. In the former, the three curves share an initial peak which corresponds to the energy required to take-off. Indeed, acquiring altitude requires more power than flying along the xy plane, as highlighted. This phenomenon is further remarked in Drone #2 landing maneuver. It includes a little parabola that yields a peak in the last part of the associated curve of Figure 7.11, which is also present in Figure 7.12b. After ~ 10 s, the drones reach and almost maintain a target altitude. The corresponding power consumption, for Drones #1 and #3, is characterized by peaks due to hovering over the interest points for 1 s and 3 s, respectively. These points are identified by the vertices of the snake-like and octagon-shaped trajectories. Instead, this phenomenon is not present on Drone #2, since its trajectory describes a continuous curve. When the drones enter the RoI, the peripherals become active and hence the IMUs' power contribution is non-zero. It can be noticed as spikes in the curves of Figure 7.11, especially in Drones #1 and #2, since they are equipped with two more energy-demanding peripherals. As soon as drones exit such region,

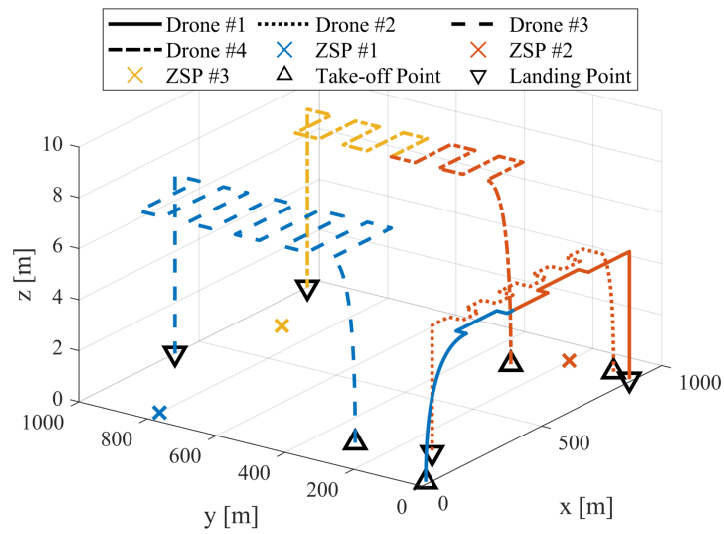


FIGURE 7.14: Trajectory design and eNB attachment for each drone, in the second scenario.

the peripherals go into standby mode, which preserves energy.

Figure 7.13 illustrates the measured RSSI of each drone at the ZSP. It clearly emerges that, on average, Drones #1 and #2 maintain a better signal quality with respect to the UAV #3. Obviously, the higher altitude, and hence the greater distance from the ZSP, worsens the communication quality, due to the Friis propagation loss employed to model the fading effects in this scenario.

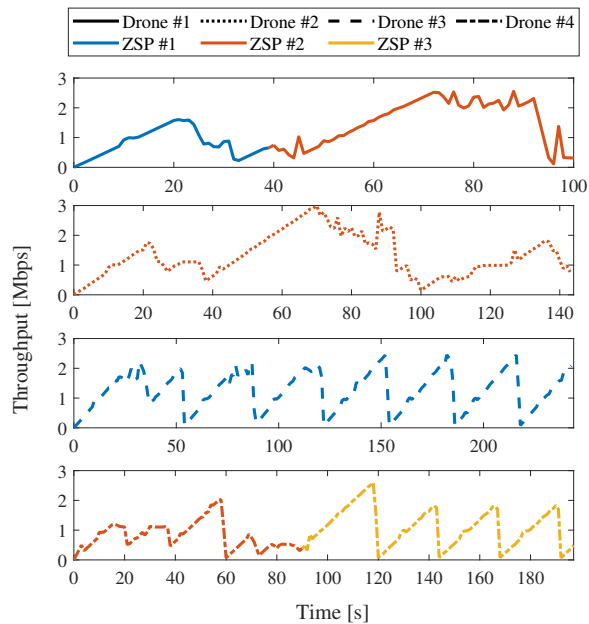


FIGURE 7.15: Drones' throughput, in the second scenario.

7.6.3 Scenario #2 - Multimedia Signals Acquisition

The second scenario, as depicted in Figure 7.14, involves a swarm composed by four drones in charge to acquire multimedia signals at different data rates which are then stored on-board and off-loaded to a remote server. To allow data upload, three ZSPs, also referred to as eNB, are deployed on the ground at $[50 \ 800]^T$, $[900 \ 200]^T$, and $[700 \ 900]^T$, respectively. All the

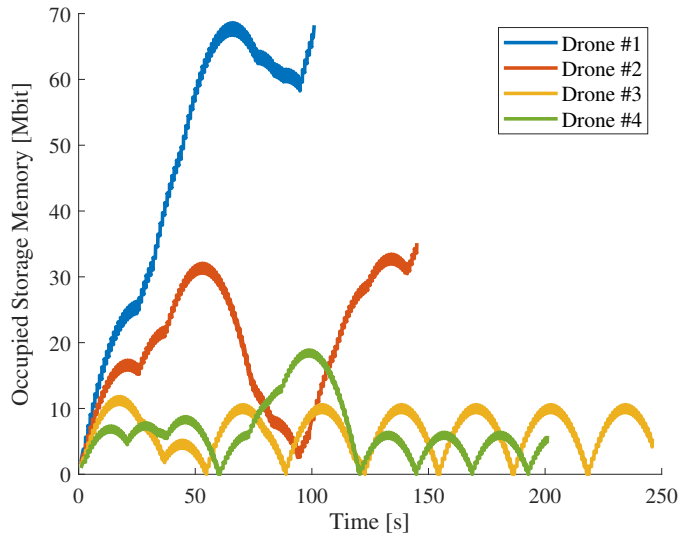


FIGURE 7.16: Memory occupancy for each drone, in the second scenario.

entities involved in the mission, which lasts 250 s, are equipped with LTE interfaces, where the Okumura-Hata propagation loss model has been employed. Drones follow snake-like trajectories, each different from the other in terms of amplitude and frequency. Nevertheless, they adopt the same mobility model with a constant acceleration of 4 m/s^2 and a maximum velocity between 15 and 20 m/s. Moreover, they are equipped with cameras that acquire at 2 Mbps, 1.6 Mbps, 1.3 Mbps, and 1 Mbps, respectively. The communication between each UAV and the remote server is handled by *Generic Traffic Applications*, with a payload size of 1024 bytes and a TCP Max Segment Size of 1380 bytes.

In the same figure, it can be further observed the attachment of the drones to the ZSPs. Throughout the mission, Drones #2 and #3 remain linked to the same eNB, i.e., ZSP #2 and #1. On the other hand, UAV #1 and #4 perform a handover procedure which changes the reference ZSP from #1 to #2 and from #2 to #3, respectively. It is worth noting that, despite Drone #1 takes-off in the same area where Drone #2 lands, they are not attached to the same ZSP. Indeed, even if the two trajectories share the same direction, they have opposite verse: while one approaches a eNB, as the mission goes by, the other flies away from the ZSP without really getting closer to another one.

Figure 7.15 shows the throughput for each drone on the associated ZSP, over time. UAV #1 experiences an average data rate of $\sim 1 \text{ Mbps}$, until the handover procedure takes place, which increases this value by $\sim 50\%$. Similarly, the average throughput of Drone #4 is ameliorated, since it increases from $\sim 800 \text{ kbps}$ to $\sim 1.1 \text{ Mbps}$. It is worth noting that there exists a pattern correspondence between the throughput and occupied storage curves (see Figure 7.16). This is particularly evident for Drones #3 and #4. When the occupied memory goes to zero, the data rate goes to zero as well. Indeed, for the information causality principle, it cannot be transmitted more information than the stored amount. Notice that this happens as long as the acquiring rate remains lower or equal to the channel capacity which, for instance, is not the case of Drone #1.

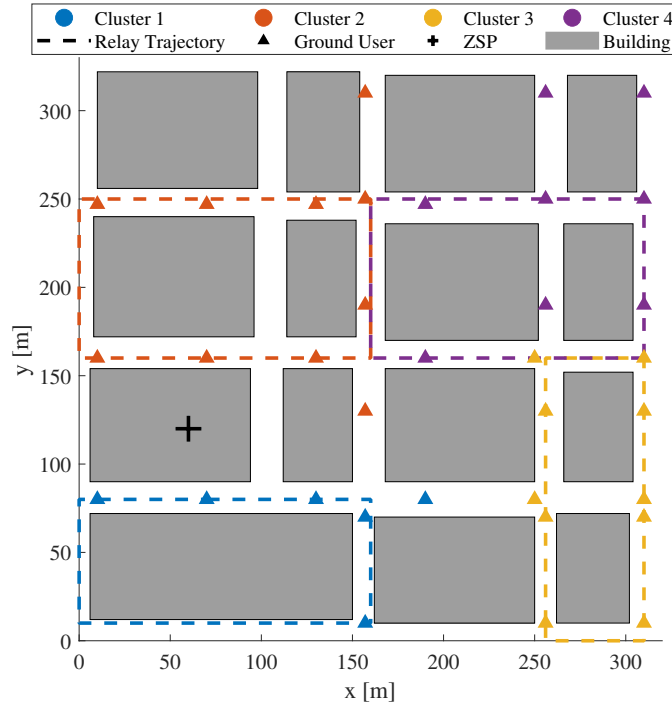


FIGURE 7.17: Scenario #3 simulation environment.

7.6.4 Scenario #3 - Smart Cities

The third scenario reproduces a smart city context, in which drones are in charge of relaying traffic coming from clusters of GUs, using the Wi-Fi technology, to a remote server over the Internet, through LTE. In this regard, the presence of buildings plays an important role both in trajectory design and in fading phenomena. The envisioned scenario is designed starting from the map of an urban area in the neighborhood of the Central Station of Bari, Puglia, Italy. The xy coordinates (i) are extracted from OpenStreetMap with the aid of OpenCV [81], (ii) rescaled according to their real profile, and (iii) transposed into the spatial reference system of the simulator. Finally, the buildings' heights are generated using a random variable uniformly distributed in [24, 30], which corresponds to the characteristic height (in meters) of the buildings in that area. As shown in Figure 7.17, four GUs clusters of different size are present on the ground. Each of them is served by a drone, which relays the traffic by means of the NAT application. The entire simulation lasts 180s and employs the `ns3::HybridBuildingsPropagationLossModel` to take into account the fading caused by the presence of buildings. It includes a combination of Okumura-Hata model and COST231 for long-range communications, ITU-R P.1411 for short-range communications, and ITU-R P.1238 for indoor ones. This allows to support a wide range of frequencies spanning from 200 MHz up to 2600 MHz. Moreover, each building is characterized by a window per room and is assumed to be built with concrete walls. The Wi-Fi stack has been configured based on the 802.11ax standard operating at 2.4 GHz and is controlled by the `ns3::IdealWifiManager`, which allows to keep track of the SINR. Thanks to this mechanism, it is possible to always choose the best transmission mode to be used, i.e., a combination of modulation, coding scheme, and data rate.

As for the network level, each cluster is connected to its relay according to the 10.[1 – 4].0.0/24 network address range, while LTE uses 7.0.0.0/8. Drones' trajectories are designed to the layout of the streets in order to minimize the shadowing effects and maximize the LoS with the GUs. Furthermore, the path also maximizes energy efficiency as the translation in the

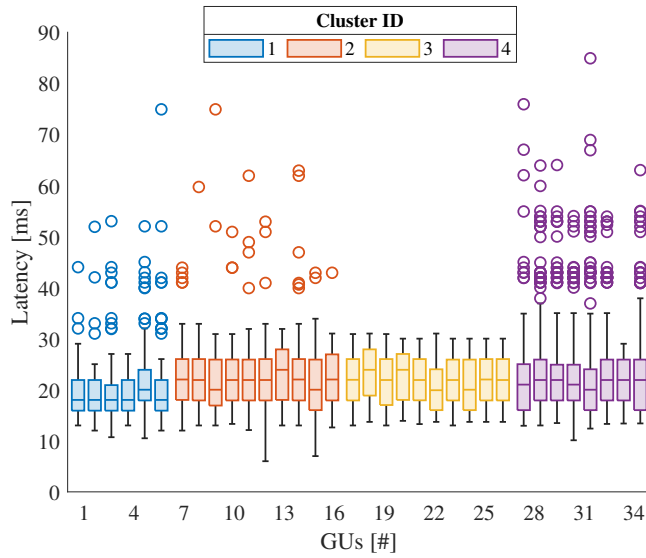


FIGURE 7.18: GUs application latency of link combined by Wi-Fi, relay drone, and LTE.

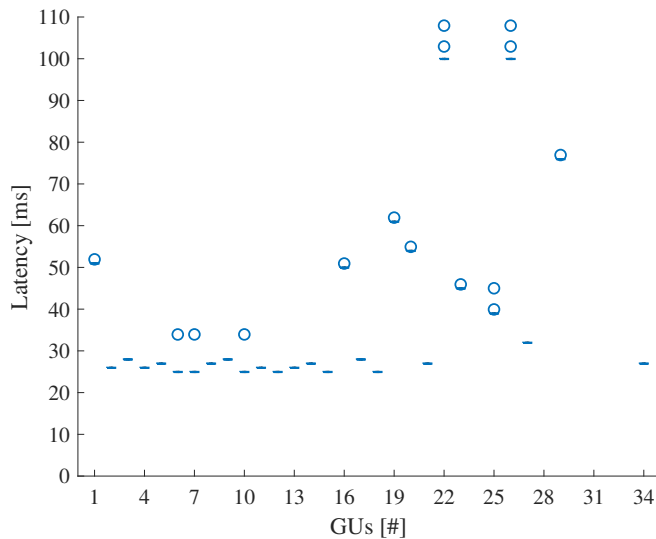


FIGURE 7.19: GUs application latency over LTE-only link.

xy plane is less costly when compared to changes of altitude. At each angle of the trajectory, the drones pause for $1s$ in order to simulate an accurate 90 degrees yaw.

Accordingly, each relay drone flies at a constant altitude of $50m$ at $5m/s$. Drones are equipped with the `ns3::NatApplication`, which implements a simple Port-based NAT strategy for UDP communications. Each GUs has a constant position and is equipped with a simple `ns3::UdpEchoClientApplication`, which periodically sends a packet of 1024 bytes to the remote address `200.0.0.1:1337` with a frequency of $10Hz$. Each packet is equipped with an application header that reports an incremental sequence number and the time of creation. Finally, the remote has a `ns3::DroneServerApplication`, which records via log messages the received packets.

The only ZSP, located at $[60, 120, 40]^T$, provides LTE access to the drones, thus allowing the communication with the remote host. Figures 7.18 and 7.19 clearly show the advantage

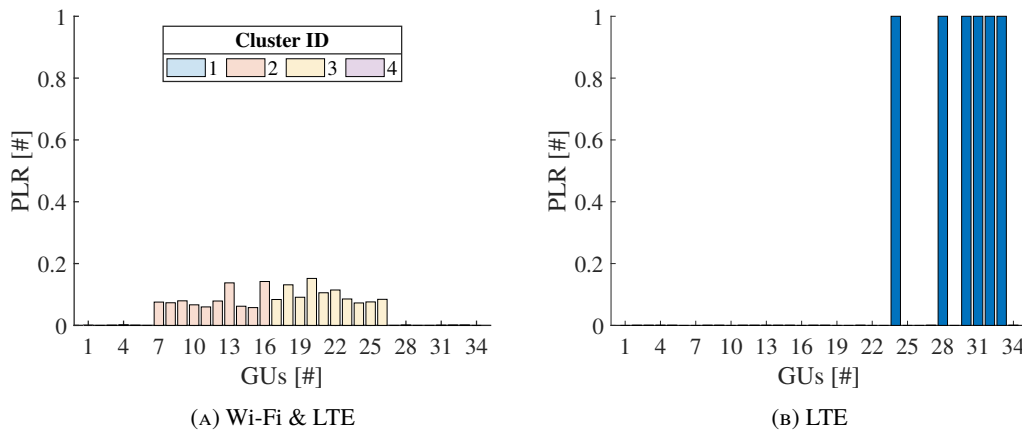


FIGURE 7.20: GUs application PLR for Scenario #3.

Scenario #	Events [#]	Real Time [s]	Sim. Time [s]
1	57,437	9	50
2	18,226,323	761	250
3 LTE	37,178,812	4,620	180
3 Wi-Fi & LTE	28,903,306	2,858	180

TABLE 7.7: Comparison of the total number of events, the real time taken to execute, and the simulated time of each scenario.

brought by the relay activity by the drones. In the relay case (Figure 7.18), all the GUs experience an average latency of $\sim 25ms$, a result that is achieved also thanks to the proposed trajectory design.

On the contrary, in absence of relay drones (see Figure 7.19), while the GUs that are closer to the ZSP are affected by a latency similar to the previous case, the farther ones register a significant delay, which inevitably compromises the reliability of the link and, hence, the Quality of Service (QoS). Nevertheless, this comes with a trade-off as highlighted in Figure 7.20, which shows the PLR in both cases. In the former, all nodes are able to transmit data to the remote, but with a loss ratio of $\sim 10\%$ for the cluster #2 and #3. It is worth noting that this result can be further improved by properly optimizing the trajectory design to target the desired trade-off. In the latter, instead, six nodes have 100% PLR, which means that there is no exchange of data.

7.6.5 Performance Evaluations

To evaluate the performance of the simulator, and hence its scalability, the performance metrics of the simulated scenarios are analyzed and compared hereby. The runtime environment is characterized by the following hardware and software specifications: (i) Intel (R) Xeon (R) Bronze 3106 at 1.70 GHz with 16 cores and no hyper-threading, (ii) RAM 92 GB DDR4 at 2666 MHz, (iii) 7200 RPM hard drives and (iv) OS Fedora 35 on LXD container [82]. To fairly compare the simulations, two metrics are selected. The former takes into account the number of events processed per second for each simulation, thus providing an insight related to the scenario complexity. The latter considers the ratio between the simulated time and the real time, thus further addressing the complexity of the designed missions. Moreover, Table 7.7 summarizes the total number of events, the time taken to simulate (Real Time), and the

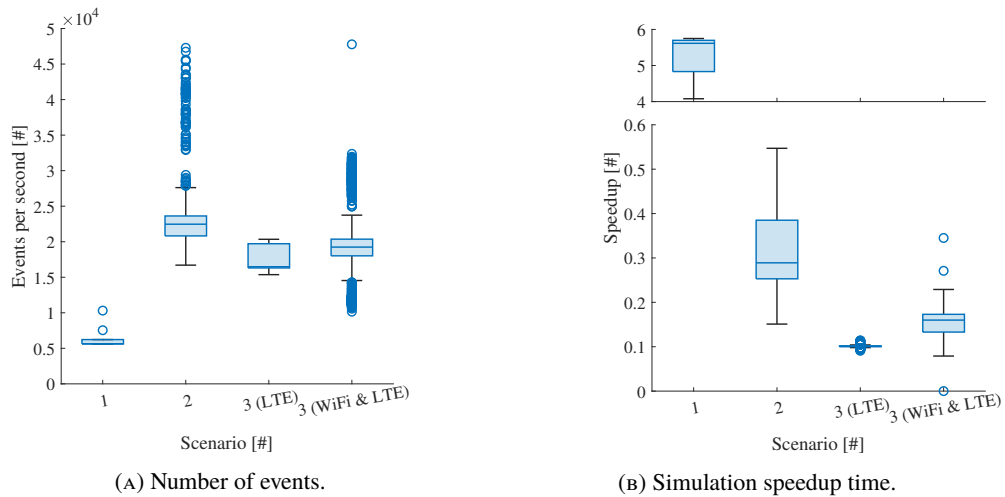


FIGURE 7.21: Performance evaluation of the different simulated scenarios.

simulated time of each scenario. It is worth noting that all scenarios are constructed differently and hence are difficult to compare. However, some clear indications can be derived from the following analysis. Indeed, Figure 7.21 shows that in the Scenario #1 the employment of the Wi-Fi technology slows the number of events processed per second, which means that the complexity is higher. On the contrary, the adoption of LTE (either mixed with Wi-Fi) reduces the overall computational complexity. However, in the first case (Scenario #1) the speedup is greater with respect to the second case (remaining scenarios): this is due to the fact that the number of generated events is way lower. This is particularly evident in the Scenario #3, where the simulation time and the number of GUs are the same, as shown in Table 7.7. Overall, even if the number of actors increases when drones relay are employed (LTE & Wi-Fi), the lower number of events generated guarantees better performances.

Conclusions and Future Works

UAVs are key enablers in many emerging verticals, thanks to their versatility in fulfilling sensing, actuation, and communications tasks. At the same time, their resources are quite constrained so that sophisticated optimization approaches are required to prolong mission lifetime and to allocate radio resources. This thesis investigates various scenarios in which drones are in charge to acquire multimedia signals which are then transmitted to a remote network entity, relying on cellular communication technologies. These missions, as discussed in Chapters 2, 3 and 4, are subject to different constraints related to kinematics, energy, memory, and radio resources allocation. Starting from the reference system model, an optimization problem is formulated to derive a mission plan able to successfully achieve the objective, which however is often non-convex and hence challenging to solve. Therefore, convex approximations and dedicated methodologies, such as BCD and SCA, are employed to cope with this issue.

Modularity, among others, is a fundamental feature characterizing drones, which can be seen from software and hardware standpoints. The former has been studied in Chapter 1, where a middleware solution has been proposed to boost the deployment of the IoD, with a flexible software interface able to abstract mission plan from drone specifics and providing a safe and unified control structure. The latter, instead, concerns the possibility to equip drones with dedicated hardware, such as multi-spectral cameras or IRSs. These are metasurfaces able to reflect incident electro-magnetic waves towards specific locations. A dedicated channel model is required to characterize the capacity of the communication link, which has been derived and analyzed in Chapters 5 and 6.

Scalability is the aspect that mostly hinders a wide spread of the IoD in civil applications. As a matter of fact, the large scale adoption should be evaluated after a prototyping phase that can be time consuming and may require unfeasible costs. To tackle this problem, simulators are an essential tool to facilitate the testing phase and state the readiness for real world exploitation. At the same time, simulators can be a learning tool for young professionals, engineering students and researchers to improve their knowledge and explore scenarios never considered before. On these bases, IoD-Sim, presented in Chapter 7, is a powerful tool which can be used to evaluate the many facets of IoD scenarios, including trajectory design, networking functionalities, mechanical characteristics, and data analytics.

Although a remarkable number of challenges brought by IoD remain disregarded, the achieved results prompt the scientific community to conduct more studies in this field. In particular, future research will

- Evaluate a real implementation of the DCL in different application scenarios, thus demonstrating the huge advantages of such middleware solution.
- Investigate more sophisticated channel models along with the achievable performance, considering all the environmental and system-related phenomena, such as Doppler effect which is often neglected.
- Investigate novel techniques and methodologies to perform the estimation of the CSI, with particular attention to the efficiency and the computational complexity.

- Assess future integrated Terrestrial/Non-Terrestrial (T/NT) networks where UAVs, aircrafts, and Geostationary and non-Geostationary satellite integrate with terrestrial nodes to provide ubiquitous and resilient wireless connectivity.
- Design and implement a dedicated module for IoD-Sim to assess novel scheduling strategies when IRS-assisted UAV-aided communication systems are involved.

Clearly, the theoretical findings that will be derived following these research directions require, after a simulation-driven validation, a real-world assessment. This implies the necessity to assemble customized drones, which in turns means that mechatronic-related knowledge and skills must be developed.

To conclude, it can be definitely asserted that drones are going to play a central role in the next wireless communication technologies, and hence the way for the IoD integration in 6G is already paved.

Bibliography

- [1] M. Gharibi, R. Boutaba, and S. L. Waslander, "Internet of Drones," *IEEE Access*, vol. 4, pp. 1148–1162, 2016.
- [2] P. Boccadoro, D. Striccoli, and L. A. Grieco, "An Extensive Survey on the Internet of Drones," *Ad Hoc Netw.*, vol. 122, p. 102 600, 2021, ISSN: 1570-8705.
- [3] M. Mozaffari, W. Saad, M. Bennis, Y. Nam, and M. Debbah, "A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems," *IEEE Commun. Surveys Tuts*, vol. 21, no. 3, pp. 2334–2360, 2019.
- [4] N. Yumi, S. Yuki, and O. Tetsuya, "An Effective Visual Programming Tool for Learning and Using Robotics Middleware," in *2016 IEEE/SICE International Symposium on System Integration (SII)*, 2016, pp. 156–161.
- [5] R. Liu, Q. Wu, M. Di Renzo, and Y. Yuan, "A Path to Smart Radio Environments: An Industrial Viewpoint on Reconfigurable Intelligent Surfaces," *IEEE Wireless Communications*, pp. 1–7, 2022.
- [6] A. S. Abdalla, T. F. Rahman, and V. Marojevic, *UAVs with Reconfigurable Intelligent Surfaces: Applications, Challenges, and Opportunities*, 2020. [Online]. Available: <https://arxiv.org/abs/2012.04775>.
- [7] B. Shang, R. Shafin, and L. Liu, *UAV Swarm-Enabled Aerial Reconfigurable Intelligent Surface*, 2021. [Online]. Available: <https://arxiv.org/abs/2103.06361>.
- [8] Y. Choi, Y. Choi, S. Briceno, and D. N. Mavris, "Energy-Constrained Multi-UAV Coverage Path Planning for an Aerial Imagery Mission Using Column Generation," *J. Intell. Robot. Syst.*, vol. 97, no. 1, pp. 125–139, 2020.
- [9] G. Iacovelli, P. Boccadoro, and L. A. Grieco, "An Iterative Stochastic Approach to Constrained Drones' Communications," in *2020 IEEE/ACM 24th International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*, 2020, pp. 1–8.
- [10] S. Zhang, H. Zhang, B. Di, and L. Song, "Joint Trajectory and Power Optimization for UAV Sensing Over Cellular Networks," *IEEE Commun. Lett.*, vol. 22, no. 11, pp. 2382–2385, 2018.
- [11] A. S. Prasetia, R. Wai, Y. Wen, and Y. Wang, "Mission-based energy consumption prediction of multicopter uav," *IEEE Access*, vol. 7, pp. 33 055–33 063, 2019.
- [12] Q. Wu, L. Liu, and R. Zhang, "Fundamental trade-offs in communication and trajectory design for uav-enabled wireless network," *IEEE Wirel. Commun.*, vol. 26, no. 1, pp. 36–44, Feb. 2019.
- [13] H. Wang, H. Zhao, W. Wu, J. Xiong, D. Ma, and J. Wei, "Deployment Algorithms of Flying Base Stations: 5G and Beyond With UAVs," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10 009–10 027, 2019.
- [14] L. Amorosi, L. Chiaraviglio, F. D'Andreagiovanni, and N. Blefari-Melazzi, "Energy-Efficient Mission Planning of UAVs for 5G Coverage in Rural Zones," in *IEEE International Conference on Environmental Engineering (EE)*, Mar. 2018, pp. 1–9.

- [15] Y. Sun, D. Xu, D. W. K. Ng, L. Dai, and R. Schober, "Optimal 3D-Trajectory Design and Resource Allocation for Solar-Powered UAV Communication Systems," *IEEE Trans. Wireless Commun.*, vol. 67, no. 6, pp. 4281–4298, 2019.
- [16] C. Zhan, H. Hu, X. Sui, Z. Liu, and D. Niyato, "Completion Time and Energy Optimization in UAV-Enabled Mobile Edge Computing System," *IEEE Internet Things J.*, pp. 1–1, 2020.
- [17] C. Zhan, H. Hu, Z. Wang, R. Fan, and D. Niyato, "Unmanned Aircraft System Aided Adaptive Video Streaming: A Joint Optimization Approach," *IEEE Trans. Multimedia*, vol. 22, no. 3, pp. 795–807, 2020.
- [18] Y. Ji, Z. Yang, H. Shen, W. Xu, K. Wang, and X. Dong, "Multicell Edge Coverage Enhancement Using Mobile UAV-Relay, year=2020," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 7482–7494,
- [19] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Mobile Unmanned Aerial Vehicles (UAVs) for Energy-Efficient Internet of Things Communications," *IEEE Trans. Wireless Commun.*, vol. 16, no. 11, pp. 7574–7589, 2017.
- [20] C. Zhan and R. Huang, "Energy Efficient Adaptive Video Streaming With Rotary-Wing UAV," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 7, pp. 8040–8044, 2020.
- [21] C. You and R. Zhang, "3D Trajectory Optimization in Rician Fading for UAV-Enabled Data Harvesting," *IEEE Trans. Wireless Commun.*, vol. 18, no. 6, pp. 3192–3207, 2019.
- [22] H. D. Tran, T. X. Vu, S. Chatzinotas, S. Shahbazpanahi, and B. Ottersten, "Coarse Trajectory Design for Energy Minimization in UAV-enabled Wireless Communications with Latency Constraints," *IEEE Trans. Veh. Technol.*, pp. 1–1, 2020.
- [23] N. Zhao, X. Pang, Z. Li, *et al.*, "Joint Trajectory and Precoding Optimization for UAV-Assisted NOMA Networks," *IEEE Trans. Commun.*, vol. 67, no. 5, pp. 3723–3735, 2019.
- [24] A. Otto, N. Agatz, J. Campbell, B. Golden, and E. Pesch, "Optimization approaches for civil applications of unmanned aerial vehicles (uavs) or aerial drones: A survey," *Networks*, vol. 72, no. 4, pp. 411–458, 2018.
- [25] L. Chiaraviglio, L. Amorosi, F. Malandrino, C. F. Chiasserini, P. Dell'Olmo, and C. Casetti, "Optimal Throughput Management in UAV-based Networks during Disasters," in *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2019, pp. 307–312.
- [26] H. Hu, C. Zhan, J. An, and Y. Wen, "Optimization for HTTP Adaptive Video Streaming in UAV-Enabled Relaying System," in *IEEE International Conference on Communications (ICC)*, 2019, pp. 1–6.
- [27] W. Feng, J. Wang, Y. Chen, X. Wang, N. Ge, and J. Lu, "UAV-Aided MIMO Communications for 5G Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1731–1740, 2019.
- [28] J. Chakareski, S. Naqvi, N. Mastrorade, J. Xu, F. Afghah, and A. Razi, "An Energy Efficient Framework for UAV-Assisted Millimeter Wave 5G Heterogeneous Cellular Networks," *IEEE Trans. Green Commun. Netw.*, vol. 3, no. 1, pp. 37–44, 2019.
- [29] T. Q. Duong, L. D. Nguyen, and L. K. Nguyen, "Practical optimisation of path planning and completion time of data collection for uav-enabled disaster communications," in *15th International Wireless Communications Mobile Computing Conference (IWCMC)*, Jun. 2019, pp. 372–377.

- [30] J. Boubin and C. Stewart, "SoftwarePilot: Fully Autonomous Aerial Systems Made Easier," in *2020 IEEE International Conference on Autonomic Computing and Self-Organizing Systems Companion (ACSOS-C)*, 2020, pp. 250–251.
- [31] A. Koubâa, B. Qureshi, M.-F. Sriti, *et al.*, "Dronemap Planner: A Service-Priented Cloud-Based Management System for the Internet-of-Drones," *Ad Hoc Netw.*, vol. 86, pp. 46–62, 2019.
- [32] G. Kakamoukas, P. Sarigiannidis, and I. Moscholios, "High Level Drone Application Enabler: An Open Source Architecture," in *2020 12th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP)*, 2020, pp. 1–4.
- [33] E. Frew, K. Glasheen, C. Alexander Hirst, J. Bird, and B. Argrow, "A Dispersed Autonomy Architecture for Information-Gathering Drone Swarms," in *2020 IEEE Aerospace Conference*, 2020, pp. 1–11.
- [34] A. Essameldin and K. A. Harras, "The Hive: An Edge-Based Middleware Solution for Resource Sharing in the Internet of Things," in *3rd Workshop on Experiences with the Design and Implementation of Smart Objects*, 2017, pp. 13–18.
- [35] D. Marcheras, M. Ayaida, N. Messai, and F. Valentin, "A New Middleware for Managing Heterogeneous Robot in Ubiquitous Environments," in *2020 8th International Conference on Wireless Networks and Mobile Communications (WINCOM)*, 2020, pp. 1–5.
- [36] J. Besada, L. Bergesio, I. Campaña, *et al.*, "Drone Mission Definition and Implementation for Automated Infrastructure Inspection Using Airborne Sensors," *Sensors*, vol. 18, no. 4, p. 1170, Apr. 2018, issn: 1424-8220.
- [37] P. Dinh, T. M. Nguyen, S. Sharafeddine, and C. Assi, "Joint Location and Beamforming Design for Cooperative UAVs With Limited Storage Capacity," *IEEE Trans. Commun.*, vol. 67, no. 11, pp. 8112–8123, 2019.
- [38] Y. Zeng, J. Xu, and R. Zhang, "Energy Minimization for Wireless Communication With Rotary-Wing UAV," *IEEE Trans. Wireless Commun.*, vol. 18, no. 4, pp. 2329–2345, 2019.
- [39] L. F. Zuluaga and T. Terlaky, *Modeling and Optimization: Theory and Applications: Selected Contributions from the MOPTA 2012 Conference*. Springer, 2013, vol. 62.
- [40] H. K. Hahn and K. Schoenberger, *The Ordered Distribution of Natural Numbers on the Square Root Spiral*, 2007. [Online]. Available: <https://arxiv.org/abs/0712.2184>.
- [41] 3GPP, "User Equipment (UE) Radio Transmission and Reception; Part 2: Range 2 Standalone (R. 15)," 3rd Generation Partnership Project, Tech. Rep., Jul. 2018.
- [42] —, "Technical Specification Group Radio Access Network; Study on NR-based access to unlicensed spectrum (R. 16)," 3rd Generation Partnership Project, Tech. Rep., Dec. 2018.
- [43] H. Nawaz, H. M. Ali, and A. A. Laghari, "UAV Communication Networks Issues: A Review," *Arch. Comput. Method. E.*, pp. 1–21, 2020.
- [44] S. J. Wright, "Coordinate Descent Algorithms," *Math. Program.*, vol. 151, no. 1, pp. 3–34, 2015.
- [45] M. M. Azari, F. Rosas, K.-C. Chen, and S. Pollin, "Ultra Reliable UAV Communication Using Altitude and Cooperation Diversity," *IEEE Trans. Commun.*, vol. 66, no. 1, pp. 330–344, 2017.

- [46] A. Gil, J. Segura, and N. M. Temme, "The Asymptotic and Numerical Inversion of the Marcum Q-Function," *Studies in Applied Mathematics*, vol. 133, no. 2, pp. 257–278, 2014.
- [47] G. Iacovelli, P. Boccadoro, and L. A. Grieco, "On the Interplay Between Energy and Memory Constraints in Optimized UAV Communications," *IEEE Networking Letters*, vol. 2, no. 4, pp. 203–206, 2020.
- [48] W. Dinkelbach, "On Nonlinear Fractional Programming," *Manage. Sci.*, vol. 13, no. 7, pp. 492–498, 1967.
- [49] Y. Zhao, Y. Li, D. Wu, and N. Ge, "Overlapping Coalition Formation Game for Resource Allocation in Network Coding Aided D2D Communications," *IEEE Trans. Mobile Comput.*, vol. 16, no. 12, pp. 3459–3472, 2017.
- [50] D. Wu, D. I. Arkhipov, M. Kim, *et al.*, "ADDSSEN: Adaptive Data Processing and Dissemination for Drone Swarms in Urban Sensing," *IEEE Transactions on Computers*, vol. 66, no. 2, pp. 183–198, 2017.
- [51] T. Liu, M. Cui, G. Zhang, Q. Wu, X. Chu, and J. Zhang, "3D Trajectory and Transmit Power Optimization for UAV-Enabled Multi-Link Relaying Systems," *IEEE Trans. Green Commun. Netw.*, vol. 5, no. 1, pp. 392–405, 2021.
- [52] H. Ghazzai, M. Ben Ghorbel, A. Kessler, and M. J. Hossain, "Trajectory Optimization for Cooperative Dual-Band UAV Swarms," in *2018 IEEE Global Communications Conference (GLOBECOM)*, 2018, pp. 1–7.
- [53] T. Stützle and M. Dorigo, "A Short Convergence Proof for a Class of Ant Colony Optimization Algorithms," *IEEE Trans. Evol. Comput.*, vol. 6, no. 4, pp. 358–365, 2002.
- [54] M. Grant and S. Boyd, *CVX: MATLAB Software for Disciplined Convex Programming*, 2016. [Online]. Available: <http://cvxr.com/cvx>.
- [55] H. Duan, "Ant Colony Optimization: Principle, Convergence and Application," in *Handbook of Swarm Intelligence*, Springer, 2011, pp. 373–388.
- [56] C. Zhan, Y. Zeng, and R. Zhang, "Energy-Efficient Data Collection in UAV Enabled Wireless Sensor Network," *IEEE Wireless Commun. Lett.*, vol. 7, no. 3, pp. 328–331, 2018.
- [57] Z. Wei, Y. Cai, Z. Sun, *et al.*, "Sum-Rate Maximization for IRS-Assisted UAV OFDMA Communication Systems," *IEEE Trans. Wireless Commun.*, vol. 20, no. 4, pp. 2530–2550, 2021.
- [58] Q. Tao, J. Wang, and C. Zhong, "Performance Analysis of Intelligent Reflecting Surface Aided Communication Systems," *IEEE Commun. Lett.*, vol. 24, no. 11, pp. 2464–2468, 2020.
- [59] A. M. Salhab and M. H. Samuh, "Accurate Performance Analysis of Reconfigurable Intelligent Surfaces Over Rician Fading Channels," *IEEE Wireless Commun. Lett.*, vol. 10, no. 5, pp. 1051–1055, 2021.
- [60] M. A. Al-Jarrah, E. Alsusa, A. Al-Dweik, and M.-S. Alouini, "Performance Analysis of Wireless Mesh Backhauling Using Intelligent Reflecting Surfaces," *IEEE Trans. Wireless Commun.*, vol. 20, no. 6, pp. 3597–3610, 2021.
- [61] M. A. Al-Jarrah, K.-H. Park, A. Al-Dweik, and M.-S. Alouini, "Error Rate Analysis of Amplitude-Coherent Detection over Rician Fading Channels with Receiver Diversity," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 134–147, 2019.
- [62] N. O'Donoghue and J. M. F. Moura, "On the Product of Independent Complex Gaussians," *IEEE Trans. Signal Process.*, vol. 60, no. 3, pp. 1050–1063, 2012.

- [63] A. Seijas-Macías and A. Oliveira, “An Approach to Distribution of the Product of Two Normal Variables,” *Discussiones Mathematicae Probability and Statistics*, vol. 32, no. 1-2, pp. 87–99, 2012.
- [64] K. Tekbiyik, G. Karabulut Kurt, A. R. Ekti, and H. Yanikomeroglu, “Reconfigurable intelligent surfaces in action: For nonterrestrial networks: Employing reconfigurable intelligent surfaces,” *IEEE Vehicular Technology Magazine*, pp. 2–10, 2022.
- [65] G. Iacovelli, A. Coluccia, and L. A. Grieco, “Channel Gain Lower Bound for IRS-Assisted UAV-Aided Communications,” *IEEE Communications Letters*, vol. 25, no. 12, pp. 3805–3809, 2021.
- [66] W. Tang, M. Z. Chen, X. Chen, *et al.*, “Wireless Communications With Reconfigurable Intelligent Surface: Path Loss Modeling and Experimental Measurement,” *IEEE Transactions on Wireless Communications*, vol. 20, no. 1, pp. 421–439, 2021.
- [67] H. Lu, Y. Zeng, S. Jin, and R. Zhang, “Aerial Intelligent Reflecting Surface: Joint Placement and Passive Beamforming Design With 3D Beam Flattening,” *IEEE Trans. Wireless Commun.*, vol. 20, no. 7, pp. 4128–4143, 2021.
- [68] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, *Proximal Policy Optimization Algorithms*, 2017. [Online]. Available: <https://arxiv.org/abs/1707.06347>.
- [69] S. Baidya, Z. Shaikh, and M. Levorato, “FlyNetSim: An Open Source Synchronized UAV Network Simulator Based on Ns-3 and Ardupilot,” in *Proceedings of the 21st ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, ser. MSWIM '18, Montreal, QC, Canada: Association for Computing Machinery, 2018, pp. 37–45, ISBN: 9781450359603.
- [70] N. R. Zema, A. Trotta, G. Sanahuja, E. Natalizio, M. Di Felice, and L. Bononi, “CUS-CUS: An Integrated Simulation Architecture for Distributed Networked Control Systems,” in *2017 14th IEEE Annual Consumer Communications Networking Conference (CCNC)*, 2017, pp. 287–292.
- [71] E. A. Marconato, M. Rodrigues, R. d. M. Pires, D. F. Pigatto, A. R. Pinto, K. R. Branco, *et al.*, “AVENS - A Novel Flying Ad Hoc Network Simulator with Automatic Code Generation for Unmanned Aircraft System,” in *Proceedings of the 50th Hawaii international conference on system sciences*, 2017.
- [72] J. A. Millan-Romera, J. J. Acevedo, À. R. Castaño, H. Perez-Leon, C. Capitán, and A. Ollero, “A UTM Simulator Based on ROS and Gazebo,” in *2019 Workshop on Research, Education and Development of Unmanned Aerial Systems (RED UAS)*, 2019, pp. 132–141.
- [73] M. Tropea, P. Fazio, F. De Rango, and N. Cordeschi, “A New FANET Simulator for Managing Drone Networks and Providing Dynamic Connectivity,” *Electronics*, vol. 9, no. 4, 2020, ISSN: 2079-9292.
- [74] S. Acharya, B. Amrutur, M. Bharatheesha, and Y. Simmhan, *CORNET 2.0: A Co-Simulation Middleware for Robot Networks*, 2021. [Online]. Available: <https://arxiv.org/abs/2109.06979>.
- [75] S. Park, W. G. La, W. Lee, and H. Kim, “Devising a Distributed Co-Simulator for a Multi-UAV Network,” *Sensors*, vol. 20, no. 21, p. 6196, 2020.
- [76] G. Grieco, R. Artuso, P. Boccadoro, G. Piro, and L. A. Grieco, “An Open Source and System-Level Simulator for the Internet of Drones,” in *2019 IEEE 30th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC Workshops)*, 2019, pp. 1–6.

-
- [77] M. Galassi, J. Davies, J. Theiler, *et al.*, *GNU Scientific Library*. Network Theory Limited, 2002.
- [78] M. Yip, “RapidJSON — A Fast JSON Parser/Generator for C++ with Both SAX/DOM Style API,” *THL A29*. [Online]. Available: <https://github.com/miloyip/rapidjson>, 2015.
- [79] C. Shepherd, “Theoretical Design of Primary and Secondary Cells. part 3. Battery Discharge Equation,” Naval Research Lab Washington DC, Tech. Rep., 1963.
- [80] O. Tremblay, L.-A. Dessaint, and A.-I. Dekkiche, “A Generic Battery Model for the Dynamic Simulation of Hybrid Electric Vehicles,” in *2007 IEEE Vehicle Power and Propulsion Conference*, Ieee, 2007, pp. 284–289.
- [81] I. Culjak, D. Abram, T. Pribanic, H. Dzapo, and M. Cifrek, “A Brief Introduction to OpenCV,” in *2012 Proceedings of the 35th International Convention MIPRO*, 2012, pp. 1725–1730.
- [82] S. Senthil Kumaran, *Practical LXC and LXD: linux containers for virtualization and orchestration*. Springer, 2017.