UAVs for Enhanced Communication and Computation

Donevski, Igor

DOI (link to publication from Publisher):
10.54337/aau504489293

Publication date:
2022

Document Version
Publisher's PDF, also known as Version of record

UAVs FOR ENHANCED COMMUNICATION AND COMPUTATION

BY
IGOR DONEVSKI

DISSERTATION SUBMITTED 2022

AALBORG UNIVERSITY
DENMARK

UAVs for Enhanced Communication and Computation

Ph.D. Dissertation

Igor Donevski

Aalborg University
Department of Electronic Systems
Fredrik Bajers Vej 7B
DK-9220 Aalborg

Dissertation submitted: August 17, 2022

PhD supervisor: Assoc. Prof. Jimmy Jessen Nielsen
Department of Electronic Systems
Aalborg University, Denmark

Assistant PhD supervisor: Prof. Petar Popovski
Department of Electronic Systems
Aalborg University, Denmark

PhD committee: Lektor Troels Bundgaard Sørensen (chairman)
Aalborg University, Denmark

Professor Carla Chiasserini
Politecnico di Torino, Italy

Professor Ismail Guvenc
NC State University, USA

PhD Series: Technical Faculty of IT and Design, Aalborg University

Department: Department of Electronic Systems

ISSN (online): 2446-1628
ISBN (online): 978-87-7573-846-5

Published by:
Aalborg University Press
Kroghstræde 3
DK – 9220 Aalborg Ø
Phone: +45 99407140
aauf@forlag.aau.dk
forlag.aau.dk

Printed in Denmark by Stibo Complete, 2022

Abstract

Wireless communication technologies are the key enabler of remote control, and consequently, the existence of unmanned aerial vehicles (UAVs). Vice-versa, UAVs could be used to provide ubiquitous wireless communications services for ground nodes (GNs), by taking the role of UAV base stations (UAVBSs). UAVBSs are highly mobile and modular communications equipment that can be dynamically deployed to provide wireless connectivity on demand. They are thus greatly useful in disaster relief, i.e. war-torn region, wildfires, floods, landslides etc. Their application also extends to mundane scenarios where permanent infrastructure is not feasible. However, UAVBSs excel in their use of mission specific scenarios where more dynamic allocation of communication resources is needed.

The goal of this thesis is to challenge the concept of UAVBS against different types of missions. Using methodologies that span from theoretical modeling and analysis, simplified simulations up to experimental testbeds, the motivation is to test the superiority of UAV systems and play to the benefit of their implementation. Nonetheless, the thesis is critical of non-optimal uses of UAVs and tries to pin them against critical drawbacks such as the energy and weight constraints. Therefore, we initiated with testing general position schemes for 5G traffic types. Following this, we tested the application of UAVBSs for reliable and timely communications through an experimental test-bed. We also studied the importance of UAVBSs in distributing federated learning (FL) computational tasks and more general radio resource allocation techniques to help the learning performance. By using reinforcement learning (RL) techniques we provided optimized path planning for the purpose of fair and energy aware distribution of radio resources, particularly in benefit to FL implementations. Above all, we evaluated the possibility of using sustainable energy generation from renewable sources such as wind and solar to compensate for the energy spent by UAVBS swarms. The results show that the UAVBS hovering position, or trajectory is a key resource in balancing many non-linear optimization criteria and can be exploited to provide exceptional performance in missions that value timeliness and fairness. Commercial UAVBSs applications are however much more nuanced and require thorough investigation of the provincial weather patterns to justify capital investments.

Resumé

Trådløse kommunikationsteknologier er den vigtigste grundsten til fjernstyring og følgelig eksistensen af ubemandede luftfartøjer (UAV'er). Omvendt kan UAV'er i rollen som UAV-basestationer (UAVBS'er) bruges til at levere trådløs kommunikationdækning til jordknudepunkter (GN'er). UAVBS'er er meget mobile og modulære kommunikationsløsninger, der dynamisk kan indsættes til at levere trådløs forbindelse efter behov. De er således meget nyttige i katastrofehjælp, dvs. krigshærgede områder, skovbrande, oversvømmelser, jordskred osv. Deres anvendelse strækker sig også til afsides lokationer, hvor permanent infrastruktur ikke er tilgængelig. UAVBS'er udmærker sig især i missionsspecifikke scenarier, hvor der er behov for mere dynamisk tildeling af kommunikationsressourcer.

Målet med denne afhandling er at udfordre begrebet UAVBS mod forskellige typer af missioner. Ved at bruge metoder, der spænder fra teoretisk modellering og analyse, forenklede simuleringer op til eksperimentelle testbeds, er motivationen at teste UAV-systemers overlegenhed og undersøge holdbare løsningsmodeller. Ikke desto mindre forholder afhandlingen sig kritisk til ikke-optimal brug af UAV'er og forsøger at inddrage kritiske ulemper såsom energi- og vægtbegrænsninger. Derfor indledte vi arbejdet med at teste forskellige positioneringsalgoritmer til 5G trafiktyper. Herefter testede vi anvendelsen af UAVBS'er til pålidelig og rettidig kommunikation gennem en eksperimentel test-bed. Vi undersøgte også betydningen af UAVBS'er i fordelingen af federated learning (FL) beregningsopgaver og mere generelle radioressourceallokeringsteknikker for at forbedre indlæring. Ved at bruge reinforcement learning (RL) leverede vi optimeret ruteplanlægning med henblik på fair og energibevidst fordeling af radioressourcer, især målrettet til FL-implementeringer. Endelig studerede vi muligheden for at bruge bæredygtig energiproduktion fra vedvarende kilder som vind og sol til at levere for den energi, der bruges af UAVBS-sværme. Resultaterne viser, at UAVBS-svæveplacering eller flyvebane er en nøgleressource til at balancere mange ikke-lineære optimeringskriterier og kan udnyttes til at levere enestående ydeevne i missioner, der værdsætter fair og tidslig dækning. Kommercielle UAVBS-applikationer er dog meget mere nuancerede og kræver en grundig undersøgelse af de lokale vejrmønstre for at retfærdiggøre kapitalinvesteringer.

Contents

Thesis Details

Thesis Title:	UAVs for Enhanced Communication and Computation
Ph.D. Student:	Igor Donevski
Supervisors:	Assoc. Prof. Jimmy Jessen Nielsen, Aalborg University
	Prof. Petar Popovski, Aalborg University

The main body of this thesis consist of the following papers.

[A] Igor Donevski, and Jimmy Jessen Nielsen, "Dynamic standalone drone-mounted small cells," in *Proc. of IEEE 2020 European Conference on Networks and Communications (EuCNC)*, pp. 342-347, 2020.

[B] Igor Donevski, Jimmy Jessen Nielsen, and Petar Popovski, "Standalone deployment of a dynamic drone cell for wireless connectivity of two services," in *Proc. of 2021 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1-7, 2021.

[C] Igor Donevski, Christian Raffelsberger, Micha Sende, Aymen Fakhreddine, and Jimmy Jessen Nielsen, "An experimental analysis on drone-mounted access points for improved latency-reliability," in *Proc. of the 7th Workshop on Micro Aerial Vehicle Networks, Systems, and Applications*, pp. 31-36, 2021. *Best Paper Award*

[D] Igor Donevski, Israel Leyva-Mayorga, Jimmy Jessen Nielsen, and Petar Popovski, "Performance Trade-Offs in Cyber–Physical Control Applications With Multi-Connectivity," in *Frontiers in Communications and Networks*, Vol. 2:712973, 2021.

[E] Igor Donevski, Jimmy Jessen Nielsen, and Petar Popovski, "On Addressing Heterogeneity in Federated Learning for Autonomous Vehicles Connected to a Drone Orchestrator," in *Frontiers in Communications and Networks*, Vol. 2:709946.

[F] Igor Donevski, Nithin Babu, Jimmy Jessen Nielsen, Petar Popovski, and Walid Saad, "Federated learning with a drone orchestrator: Path planning for minimized staleness," in *IEEE Open Journal of the Communications Society*, Vol. 2, pp. 1000-1014, 2021.

[G] Nithin Babu, Igor Donevski, Alvaro Valcarce, Petar Popovski, Jimmy Jessen Nielsen, and Constantinos B. Papadias, "Fairness Based Energy-Efficient 3D Path Planning of a Portable Access Point: A Deep Reinforcement Learning Approach," **submitted** in *IEEE Open Journal of the Communications Society*, 2022.

[H] Igor Donevski, Marco Virgili, Nithin Babu, Jimmy Jessen Nielsen, Andrew J. Forsyth, Constantinos B. Papadias, Petar Popovski, "Sustainable Wireless Services with UAV Swarms Tailored to Renewable Energy Sources," **submitted** in *Transactions on Smart Grid*, 2022.

This thesis was supported by the European Union's research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 812991 "PAINLESS" within the Horizon 2020 Program. PAINLESS is multi-partner European Training Network (ETN) project, within the framework of the H2020 Marie Skłodowska-Curie Innovative Training Networks (ITNs), where Igor Donevski participated as an Early Stage Researcher (ESR) and collaborated with 14 other ESRs. PAINLESS relates to H2020-MSCA with a vision to produce the first generation of experts in a radically new wave of energy autonomous networks that will revolutionise the wireless networking technology landscape and the plethora of associated vertical business sectors.

Acknowledgement

Working on a PhD is a unique experience like sailing on the waves of stupidity and genius, motivation and demotivation, successes and failures, acceptances and rejections. To deal with these ups and downs required not only me at the helm, but also the joint effort of the ones around me.

Rightfully, I give my deepest gratitude to Jimmy and Petar. I will always remember the many two-minute talks with Jimmy that ended being invaluable hours, and the invaluable five minute long talks with Petar that were worth hours. They were always there to read the charts of experience and provided the best guidance for both my work and wellbeing. Above all, I thank them for their genuine care to help me become a captain in the sea of science and reach to a more objective view onto the world.

I am grateful of being part of the PAINLESS mission and cherish the times I navigated difficult scientific problems with Nithin and Marco. I thank Christian Raffelsberger and the people at Lakeside-Labs for the most engaging over-the-screen secondment with excellent results. I am also grateful to Branka Vucetic for hosting me at Sydney, which was a once in a lifetime experience to see the communications world from the other side (literally). I am grateful for my crewmates of the connectivity section for all lunch conversations, the fun hangouts and most importantly my office-mates Anders, Radek, and Lam.

Unfortunately, time did not stop and wait for me to finish my thesis voyage, and I was lucky enough to have a lot of dear people around me to send me daily supplies of love and support. Above all, my dearest Aleksandra that has always been my mast; my parents and my sister that show up at the first signal of distress; and my closest friends that no matter distance, have been there like family.

Igor Donevski
Aalborg University, August 17, 2022

Part I

Introductory Chapters

1 Introduction

1.1 Background

Unmanned Aerial Vehicles (UAVs), also known as drones, are useful flying devices that since their conception have had an intricate cross section with wireless communications technologies. In the early developments, wireless communication links were considered essential for providing control over and receiving the feedback from the airborne UAV. Since UAVs were initially designed as military equipment, they were considered useful in military communications [1]. The innovation behind UAVs made way for early visions for providing mass connectivity with UAVs [2], which also brought up the question of optimizing UAV placement for serving ground nodes (GNs) [3]. However, there is an increasing trend of using the devices in more diverse missions where exclusive control is not mandated, such as delivery, automated surveillance, and acting as small base stations.

The use case of UAVs as base stations (UAVBSs) is a subcategory of non-terrestrial networking, that offers flexibility on several domains: planning, deployment, traffic scaling, and user cluster targeting [4, 5]. However, there are different ways in which airbourne objects can keep themselves in the air, through aerodynamics like fixed wing airplanes and rotational wing copters, or through being lighter than air like blimps and airships. Each type has different degree of mobility, where rotational wing copters are the most versatile and capable of fast relocation as well as hovering at the same position. On the other hand, UAVs with limited mobility are named depending on their hovering altitude as low or high altitude platforms (LAPs or HAPs). The sub category of rotational wing drones, multi-copter drones, are the most ubiquitous type of UAV technology, which makes it very easy to prototype and implement. As such, through the rest of this thesis, nearly all references to the names of UAVs or drones is directed towards, but not always limited to multi-copter drones. Unfortunately, the main drawback for multi-copter drones is their limited airtime due to counteracting their weight with a limited energy supply, from which stems another interesting research topic of trajectory planning.

Infrastructure-less Networks

In classical cellular infrastructures, the problem of optimizing the wireless communications services to a given area is divided in two phases: BS positioning as a pre-deployment planning phase with the goal to satisfy customer demand and a post deployment resource optimization of operational wireless resources. However, the densification and adaptability required by the fifth and sixth generations of cellular infrastructure 5G and 6G [6, 7], requires blurring the lines between resources allocated at deployment and operation. This puts UAVs in a very strong position where they are merging the positioning and the resource allocation problems in one, as they can leverage their mobility

to satisfy both problems at once. As such, optimal UAV positioning is the pivotal research problem for deploying communication systems where the communications resources are tailored to the needs of the ground nodes (GNs). This is illustrated in Fig. 1, where the UAV could fly higher and closer to the red user to establish a line of sight (LOS) link, while maintaining the existing LOS link with the blue one.

Moreover, a 52% and 42% capital and operating expenditures (CAPEX and OPEX) respectively are due to infrastructural costs, such as rents and leases, in developed countries [5]. Avoiding a portion of these site management costs can be achieved through using a UAVBS. On the other hand, it is important to note that airborne networks do not offer a big reduction in OPEX due to the power-hungry nature of UAVs. To elaborate, UAVs need to spend large amounts of energy to counteract the downard force of gravity, and therefore maintain their aerial position. To add, mass use of UAVs invokes safety, privacy, and noise concerns. These drawbacks are unfortunately dimming the future of UAVs as a replacement for fixed position BSs. Therefore, UAVBSs should rather be considered in extraordinary scenarios where central infrastructure cannot be relied upon such as disaster relief from floods, earthquakes, wildfires, and socially induced grid failures. To add, UAVBSs are extremely useful for environments that are tough for implementing classical infrastructures, particularly when reliability is of the essence. Finally, UAVBSs are superior to traditional BSs with regards the wireless propagation due to their higher vantage point, which benefits outdoor GNs. However, complex antenna implementations may be infeasible for UAVBSs due to their restrictive energy and weight constraints.

For this reason, the *Energy-autonomous portable access points for infrastructure-less networks* (PAINLESS) project was founded with the goal to provide a thorough investigation of the integration of wireless communication services with the common UAVBS energy constraints. To address the energy constraints, the PAINLESS project intended an analysis of sustainable ways of energy generation with adequate energy storage technologies. Such interdisciplinary themes entailed engaging in collaborations between the project partners. Within the project, this thesis focused onto the verification of the usefulness of UAVs in providing wireless communications services (for various applications) with low or no support of the cellular infrastructure.

Ground-UAV Wireless Connectivity

UAVBSs have an intricate advantage in providing communication services to outdoor users that comes from the predominance of LOS links to GNs. Nonetheless, establishing a LOS link is not guaranteed, also illustrated in Fig. 1, and is dependent on the probability of a LOS occurring P_{LOS}. This probability depends on the elevation angle with regards to the UAV and the physical properties of the built environment [8]. Following

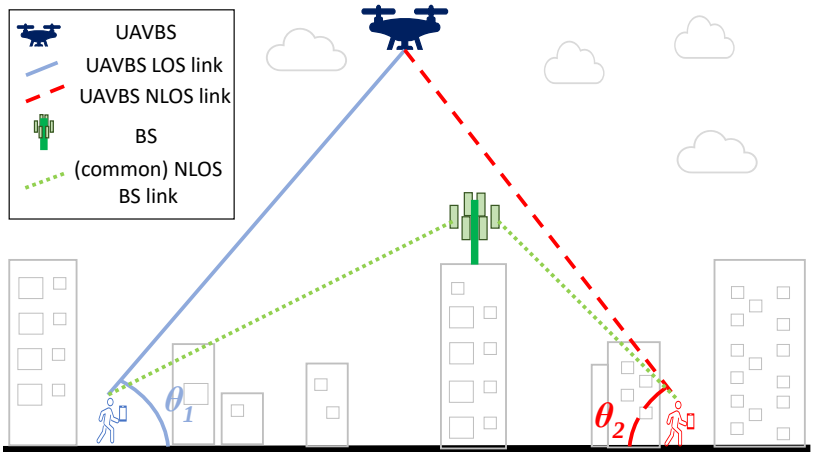


Fig. 1: A generalized illustration of a UAVBS implementation with regards to a classical BS.

the simplified s-curve approximation of the probability the expression is [9]:

$$P_{\mathrm{LOS}} = \frac{1}{1 + a\exp(-b(\theta - a))}, \tag{1}$$

where θ is the elevation angle between the user and the UAV, and a, b are constants that are dependent on the propagation properties of the environment. Since there are only two propagation groups, users with LOS and those with non-LOS (NLOS), the probability of having an NLOS link is $P_{\mathrm{NLOS}} = 1 - P_{\mathrm{LOS}}$. Despite the free space path loss (FSPL) coefficient, there is also a significant impact that the built environment induces for the propagation, η_{LOS} and η_{NLOS}, onto LOS and NLOS links respectively. These η values, commonly referred to as excessive path-loss, are the averages of a normally distributed random variable of the large scale pathloss and fit into the FSPL equations as [9]:

$$\ell_{\mathrm{LOS}} = -10\log_{10}(D_{\mathrm{t}}) + 20\log_{10}(d) + 20\log_{10}(\frac{f4\pi}{c}) + \eta_{\mathrm{LOS}}, \tag{2}$$

$$\ell_{\mathrm{NLOS}} = -10\log_{10}(D_{\mathrm{t}}) + 20\log_{10}(d) + 20\log_{10}(\frac{f4\pi}{c}) + \eta_{\mathrm{NLOS}}, \tag{3}$$

where, both ℓ are expressed in [dB], ℓ_{LOS} is the expected pathloss for a GN with LOS link, ℓ_{NLOS} is the expected pathloss for a GN with NLOS link, D_{t} is the directivity of the UAV mounted antenna, d is the physical distance between the UAV and the GN, and the $\frac{f4\pi}{c}$ term is the FSPL constant of the transmitted wavelength. An interesting perk of UAVBSs is that they can establish an optimal θ, considering that θ is the elevation angle of a user sitting on the edge of the area of coverage. This allows for

solving the maximum coverage, and the optimal altitude problem only through solving the geometry of the system and not the absolute dimensions of the system [9, 10]. In other words, if we need to provide coverage to an area with a fixed size, there is an optimal altitude for the UAV that maximizes the spectral efficiency, and it depends directly on the propagational properties of the environment.

Federated Learning

UAVs have had a fundamental usefulness in problems of distributed computation. Due to the flexibility of having computational services that are portable and easily deployable in remote areas. As such, UAV cloudlet implementations have received a lot of attention from the edge computation community [11, 12]. However, in the field of distributed computation there is a new and emerging concept in the integration of machine learning (ML) implementations across multiple devices. Naturally, the implementation of distributed computation for ML tasks birthed the term of distributed learning (DL). This term, DL, generally referred to networks of learners and server where all learners are identical to each other, and the server has absolute control over the data and task parallelization. However, the DL term got extended to potential systems where each learner is an individual entity with unique data, which birthed the concept of federated Learning (FL) [13]. Thus, the server becomes more of an orchestrator and each learner can have an intricate impact on the ML performance [14], as the learner is the device that performs the computation. As such, the FL implementation is very useful in wireless transmission scenarios where the UAVBS takes the role of an orchestrator. In detail, FL systems have a distinct advantage when orchestrated by UAVs, such as, segmenting cohorts through only positioning, better wireless channels towards outdoor users, more fairly distributed wireless resources, improving fairness through trajectory and positioning, and providing services in difficult-to-reach areas. Moreover, since FL requires a lot less processing power from the orchestrator, UAVs are an ideal candidate for this role as it avoids the energy and weight constraints of flying heavy computational equipment. For this reason, the research topic of FL integration with UAVs has been trending in recent literature [15].

The basic principle behind federated learning is through cyclically passing the ML model, defined by weights ω, between the orchestrator/server and the learners. At the beginning of a cycle, the orchestrator transmits the general weights to each learner $k \in \mathcal{K} = \{1, 2, .., K\}$. The learners proceed with the learning of the task, and returns the ML model to the orchestrator. After all learners have sent their individually trained ML models to the orchestrator, the orchestrator performs a weight aggregation step to compute a new iteration of the general weights ω_g that will be sent to each learner in the new cycle. This FL system most often works on the federated averaging principle for solving a singular learning objective $\min_{\omega} f(\omega)$. The global optimization function $f(\cdot)$ is thus calculated across all learners that have a local optimization function $F_k(\omega)$ that

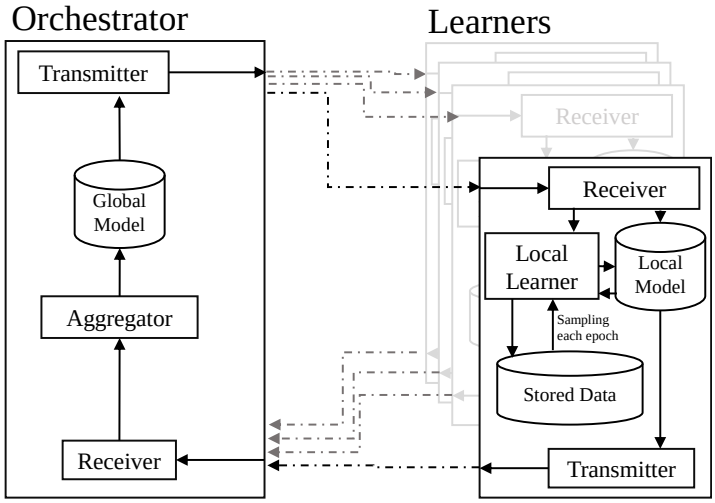


Fig. 2: A simplified illustration of a FL implementation.

is processed each round as [13]:

$$f(\omega_{\mathrm{g}}) = \sum_{k}^{K} p_k F_k(\omega) = \mathbb{E}[F_k(\omega)], \tag{4}$$

where $p_k \geq 0$ and $\sum_k p_k = 1$ are the averaging weights assigned per each learner. Most often, learners are equally weighted to avoid unexpected convergence failures [16].

While this method has been successfully used in many different applications, it fails to provide convergence in a very heterogeneous setting. To elaborate, heterogeneity refers to the uniqueness of each learner, mainly regarding its processing power or data stored. Moreover, when transmitting over wireless channels the difference in communication speed can become drastic and introduce a different kind of heterogeneity. Therefore, it is more useful to use a federated proximal averaging that reduces the impact of the heterogeneity by purposely slowing down the learning process at each learner [14]:

$$F_k(\omega; \omega_{\mathrm{g}}) = L_k(\omega) + \frac{\mu}{2}\|\omega - \omega_{\mathrm{g}}\|, \tag{5}$$

where μ is the new hyperparameter that impacts the intensity of the proximal term, and $L_k(\omega)$ is the training loss function. The introduction of the proximal term $\frac{\mu}{2}\|\omega - \omega_{\mathrm{g}}\|$ diminishes the impact from each learner through calculating the euclidean distance between the instantaneous local model and the global model that was transmitted to the learner at that round. Depending on the value of the hyperparameter μ, the implementation can be the classical federated averaging ($\mu = 0$) to implementations of

proximal federated learning ($\mu > 0$). The strength of μ needs to be properly tuned to the heterogeneity, and FL implementations that have well-balanced hyper parameters converge faster in scenarios where the learners have non-IID data and unequal processing times [14]. Due to this, potential combinations of FL with UAVs that aim to address FL heterogeneity would need to account for the inclusion of proximal FL.

Even in the case of FL, the UAVBSs fulfill their role of learning distribution through the exploitation of wireless resources. As such, UAVBSs have the power to control the learning process in FL networks by properly allocating the wireless resources [17]. However, opposed to a static BS, a UAVBS that has the role of an FL orchestrator has another benefit in which it can also change its physical location.

1.2 Motivation and Objectives

As previously indicated, UAV positioning has a great impact over the channel properties. This has a compounding effect in the problems of radio resource optimization, which has been widely recognised and sustained by the literature [4]. Given this importance of UAV positioning, this thesis considers the problem of position optimization as central in providing a proper investigation for the feasibility of UAVBS systems. Therefore, this thesis is focused towards answering three core research questions (RQs) on three separate themes concerning UAV wireless communications. All three RQs touch on different topics for UAVs that are by themselves very wide and complex fields that do have a strong research foundation. The main incentive behind the phrasing of the questions is to focus on the applicability of the intricate strengths of UAVs, or question the commercial feasibility of UAVBSs.

RQ1 – The question on communication services:

The research community behind UAVBSs has been mostly focused in use-cases where user targeted scenarios or area targeted ones [5]. However, there has been a narrow research gap to investigate the impact of UAVs in scenarios where user targeted communications are offered while a stable coverage needs to be maintained over the whole area. Moreover, dynamic movements in such cases have been a neglected research area to which we intend to uncover, particularly in the cases where UAVBSs have directional transmitters [9, 18]. It goes without saying that it is important to design systems that are in sync with the three proposed traffic classes of the 3$^{\text{rd}}$ Generation Partnership Project (3GPP). According to this, future cellular implementations should include the three traffic classes: enhanced mobile broadband (eMBB), massive machine-type communications (mMTC), and ultra reliable low latency communications (URLLC), that have different reliability, timing and bandwidth requirements [19]. When providing communications services, the UAVs can be specialized in one or several of the traffic classes and aid the main infrastructure or simply focus on guaranteeing the transition towards mission critical communications. One motivating aspect of this is exploiting

the position of the UAV to assign priority classes between any of the traffic categories, eMBB, mMTC and URLLC. Nonetheless, offering latency constrained communications services is of utmost importance when communicating with or through UAVs [20]. The forthcoming URLLC traffic class has very high timing and reliability constraints that are very difficult to establish on multi link systems [21]. Therefore, numerous applications are not limited to the URLLC constraints, such as cyber-physical control applications do not require per-packet reliability [22]. In such use cases, the services define process failures as an outcome of longer delays that most likely due to burst packet errors (loss of several sequential packets) [23]. Finally, since most personal electronic devices contain both cellular and WiFi interfaces, the coexistence of both interfaces can be exploited to reduce the impact of bursty channels onto a cyber-physical implementation [24]. Such implementations are not only limited to uses at UAVs but extend to 5G mMTC use cases. Motivated by the above the first research question of the thesis was formulated as:

- How does UAV mobility impact the wireless communications across many users with various service requirements in a predefined cellular area, and is this practically feasible? In what ways can cyber-physical communications for ground users be improved?

RQ2 – The question on distributed computation:

The boom in distributed computation has been due to two main incentives, maintaining privacy and decreasing the reliance on data-centers [13]. The concept of distributed learning that later evolved to FL includes several hindrances due to different types of diversity among learners [25]. In FL, the distribution of the data used for training most often is not independent and identically distributed (IID) also referred to non-IID (data diversity), and different learners may have different computational capabilities (computational diversity) [14]. These challenges harm the performance of FL that is reflected through the convergence time and accuracy of the models. In short, FL implementations provide additional challenge for having quick and accurate ML solutions, when compared with a classical ML implementation that learn the same task. However, FL does not guarantee privacy by itself and needs to be combined with other privacy preserving methods. Moreover, it is commonplace for new proposals of FL implementations to assume that they can inspect the distribution of data within a learner itself to justify network adaptations ahead of time [26]. This violates privacy and thus motivates us to discover heuristic metrics through which, we can better allocate network resources for FL performance. Moreover, UAVs have a uniquely good position in the world of distributed computation since they are inadequate to carry heavy and energy demanding computational hardware themselves. Thus this provides plenty of motivation to look into the roles of UAVs as task distributors for ML implementations. Motivated by the above, the second research question is formulated as:

- Are UAVs suitable FL orchestrators in the world of privacy oriented ML? Does UAV path planning improve the performance of FL implementations?

RQ3 – The question on energy balancing:

Finally, even though neither UAVBS nor a UAV FL orchestrator require the UAV to carry heavy hardware, the problem of properly managing the UAV resources to achieve increased air-times still stands. This motivates us to look better into ways that improve the UAV's capability in carrying out its task for longer periods of time, while better fulfilling the mission's objective [27]. This research field has received a lot of attention from the scientific literature, justifiably by the complexity of the task. Nonetheless, there are a plethora of viable research directions due to the heavily derivative nature of the subject, where minor changes to the aeronautical model will have significant impact onto the mission objective. Unfortunately, no amount of flight optimization can nullify the energy demand of UAVs which is much greater than that of classical infrastructure. This is one of the main barriers of mainstream usage of UAVBSs and needs to be accounted for, especially considering the global energy impact of large scale implementations of this UAVBS system [28]. This can only be justified if sustainable sources of energy are used to charge the energy hungry UAVs, which become problems of economics and finance. Motivated by the above, the second research question is formulated as:

- How to design UAV trajectory implementations under realistic energy and flight-time constraints? Are the energy requirements of UAV communications systems scalable to bigger swarms when supported by sustainable energy?

1.3 Methodology

In order to achieve the research objective and address the RQs in the best manner, possible research papers were published or submitted at a journal or a conference/workshop proceedings. These papers are founded on various research campaigns that can be based on an analytic, a simulated, or an experimental approach. And in order to reach a higher level of quality, many of the research works were supported by genuine and strong cross collaborations between partners of the PAINLESS research project that have a strong basis in one of the methodologies required to provide answers to the question. Each research question required a slightly different methodological approach as follows:

- RQ1: The research behind the question of communication services for and from UAVs was mainly developed through internal Aalborg University collaborations and a very essential collaboration with Lakeside-Labs GmbH. Moreover, the internal approach to UAVs was mainly analytical, consisting of the use of stochastic analysis such as a point process in combination with classic computational optimization methods to find optimal allocations. To test some of the hypothesis an

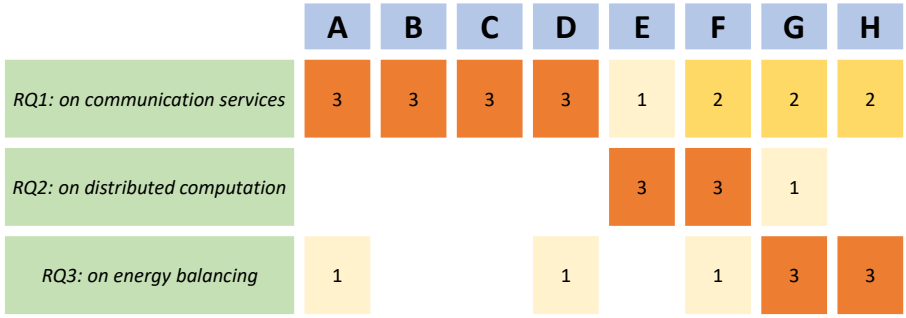


Fig. 3: An overview of the contribution of each paper to each research question, on a scale of 1-3 (least to most related) where non-related works are left blank.

extremely valuable experimental dataset was collected through a UAV testbed. And finally, since discrete-time Markov chains (DTMC) were useful in modelling the system, dynamic programming approaches were used to provide optimal transmission policy in transmission paradigms.

- RQ2: The research behind the question of FL was mainly supported with empirical implementations of the proposed systems. Moreover, FL implementations were done in Python using the TensorFlow package. To find solutions for the complex optimization criteria imposed by resource allocation targets in FL, two approaches were considered. Model-less RL methods were found eligible to provide solutions, along with a model adapted successive convex programming (SCP) approach.
- RQ3: To address the question of energy, a productive collaboration within the members of the PAINLESS research project was conducted. The core methodology was to mathematically model the energy expenditure process of a draining lithium battery through the Peukert effect. A market survey is also essential to understand the pricing of each item as energy supported systems carry high significance through the economical impact of their implementation. Given the complex optimization criteria, RL methods are needed to address this RQ as well. Where eligible, heuristic and custom algorithms that combine several classical algorithms were integrated to produce tractable solutions.

2 Summary of contributions

Since the thesis is organized as a collection of papers, this section is focused on listing and providing a discussion for each of them. This discussion encompasses analyzing the papers' affiliation to each research question (summarized in Fig. 3), but also their

contributions in their respective research fields. Before continuing, there is a need to note that due to the plethora of research on the topic, the terms UAV and drone are used interchangeably, and drone small cell (DSC) is a type of implementation of an UAVBS and are thus interchangeable terms.

Paper A: *Dynamic Standalone Drone-Mounted Small Cells*

Given the assumption that the use of UAVBSs will most likely encompass standalone deployments where it receives no aid from nearby infrastructure, in this paper we investigated the case where UAVs operate on their own (without the aid of cellular infrastructure) for the entire coverage area (CA), but do position opportunisticly in favor of ground user locations. Moreover, this paper is focused on the impact of three different dynamic horizontal opportunistic positioning (D-HOP) techniques over stochastic user positions and having directional antenna equipped on the UAV. Moreover, the paper uses more classical computational geometry approach to calculate fairness based positioning like the smallest bounding circle problem. As such the paper is strongly and directly connected to Q1, and it hints towards future works in Q3 by also investigating the distance travelled for each D-HOP.

Since the interaction between horizontal positioning and directive antennas has seen little attention from the rest of the research community, this paper is the first to question the compatibility of UAVs and directive antennas, especially the beamwidth requirements set by the CA and UAV altitude. We come to realize that antenna directivity is at odds with UAVBSs as it influences the optimal UAV position, and with that impacts the usefulness of UAVBSs. As the UAV altitude is directly influenced by the propagation properties of the environment, different D-HOP implementations reach different rate improvements. The most influential on D-HOP usability is the user density that presented an impact of per-user average rate improvements of up to 20-35% in low-user density scenarios, or 3% - 5% in dense scenarios. These results are impressive considering that the D-HOP implementations operate in the most difficult user distribution, which is uniform as in a Poisson point process (PPP). Finally, as the first work of the thesis, this paper was used to pave the way towards understanding and implementing more advanced UAV systems. As such, it is the basis for most papers included in this thesis.

Paper B: *Standalone Deployment of a Dynamic Drone Cell for Wireless Connectivity of Two Services*

This paper continues the investigation of standalone UAVBSs, but it extends the coverage constraints towards the ones required by the 5G traffic classes. Moreover, we aim to exploit the D-HOP behavior to distinguish two different types of users in a specific CA. In detail, we consider the user requirements of a broadband (BB) devices as a higher

priority user within the CA, within which there are an unknown number of machine-type-communications (MTC) devices that have a random access. The core differences between both user types is that BB users can announce themselves ahead of time and have prolonged data stream sessions, while the MTC users have short messages that come at random that the UAV must be prepared to receive at any time. This research work is obviously and strongly directed to providing answers to RQ1 and does not interact with the other RQs.

The service offered to these users can be optimized by orthogonally slicing the wireless resources and D-HOP, which in turn becomes a tradeoff between guaranteeing the services of each traffic class. Moreover, we treat the D-HOP as a computational geometry function that directly redistributes the BB users within the area to satisfy the fairness bound by locating itself at the center of the smallest bounding circle of active BB users. As such we search for the optimal DSC deployment parameters of radio resource allocation between both active services and the altitude of the UAV that jointly maximize the average rate of BB. Understandably, as in the previous work, the propagation properties of the area directly impact the benefits of the implementation. However, while the application is generally useful for every type of topology and strongly outperforms static UAVBSs, the performance of this type of system is outstanding in urban settings. This performance for deployments in an urban setting can provide average rate improvements of 50-90%. This approach is the first of its kind and is thus a direct contribution towards optimal placement of UAVs in a standalone setting where continuous coverage of two services is required.

Paper C: *An Experimental Analysis on Drone-Mounted Access Points for Improved Latency-Reliability*

Since the previous work was concerned with the BB and MTC traffic classes, there was an obvious incentive to investigate the role of UAVs in service of timing-constrained scenarios such as URLLC. Even though the URLLC requirements are based on the reliability of a single link, UAVs are required to incorporate wireless links on both the fronthaul and the backhaul, which is sometimes called the middlehaul since each transmission would have to undergo the cellular wired backhaul. Since URLLC is very hard to achieve across two wireless links we were motivated to use established mass market communications technologies where the UAV will act as an access point and provide low-latency connections to users on the ground. This role of the UAV has been many times proposed mainly due to the ability of UAVs to establish LOS links to both the Base Station backhaul and the UE fronthaul. However, related research has not performed thorough tests of the efficacy of this system especially when the intricate combination of LTE and Wi-Fi has been introduced in the system. In this work we experimentally evaluated the capability of providing low-latency coverage to individual GNs by hovering right above them. The experiment, done in collaboration with our colleagues of the PAINLESS project at Lakeside-Labs GmbH, encompassed creating a

custom UAV testbed and several measurement campaigns. Same as the previous work, this paper was dedicated solely towards RQ1.

The initial campaign was done to understand the backhaul implications of the flying altitude. As previously shown by other research, there are several unpredictable factors that influence such a deployment such as the propagational pattern of the BS antenna and the interference from other nearby BSs. Thus the initial sweep of measurements have shown that the best flying altitude for the topology of Klagenfurt, Austria was 30m as it was right above the rooftops of nearby buildings and in line of sight with the BSs. To measure the round trip, end to end latency of the entire system an in-house measurement tool by Lakeside-Labs was used that transmits a packet each 100ms. While the average latency for the UAVBS system was identical to having a direct LTE link, this system provided an additional 6.4% of coverage along with considerably smaller latencies for 0.95 reliability (50ms versus the 120ms by a direct LTE connection). As such, this was the first work to investigate the issue of UAVs as relays and discover their eligibility towards more reliable connections before the proliferation of URLLC.

Paper D: *Performance trade-offs in cyber-physical control applications with multi-connectivity*

This paper takes a different perspective and is directly involved with communications services from the ground and has a broader scope than just UAV communications. Nonetheless, this paper proposes using multi-connectivity to the benefit of cyber-physical control systems and reduce the impact of outages such as in transmitting control information to UAVBS systems. Multi-connectivity such as in the case of an implementation of packet cloning through Wi-Fi and LTE interfaces, can also be called interface diversity. Interface diversity offers a solace for communication problems that arise consecutive packet errors due to temporal correlation in a communications channel, or otherwise known error bursts. The use of separate interfaces allows for alleviating the lack of correlation between both channels to resolve the problems of the blocked channel. This paper looks to find an optimal transmission policy for when each transmission requires some form of cost such as in the form of energy or financial expenditure. As such this paper is a firm contribution towards RQ1 and does not cover the other RQs.

To model channels that have temporal correlation, the paper uses the Gilbert-Elliott model that distinguishes a good or bad states that have different transitional properties. The transitional properties for the LTE and the WiFi interfaces were derived from previously conducted measurement campaigns. This model was translatable to a partially observable Markov decision process (POMDP) where each transmission action contributes towards the belief for the unobservable states where no transmission has occurred. To find the optimal policy, we use classical value iteration to calculate Q-values of the underlying MDP. To later find the optimal policy in the POMDP space, we use the Q_{MDP} approach, that weights the actions given some belief after an obser-

vation of the channel. Along with the optimal policy we test a reduced version of the POMDP implementation, called F-POMDP, that has only two belief states instead of the infinite ones in the full PMDP implementation. This reduced implementation is done in favor of reducing computation that may be beneficial in systems that need to recalculate the optimal policy due to a change in the transitional probabilities of the GE channel. As such this paper contributes towards better resource utilization of multi-connectivity/interface-diversity systems particularly when under energy constraints.

Paper E: *On Addressing Heterogeneity in Federated Learning for Autonomous Vehicles Connected to a Drone Orchestrator*

This paper proposes an application scenario for FL implementations where the UAV takes the role of an orchestrator. Moreover, we assume an implementation where the UAVBS also acts as a drone traffic monitor that produces a significant amount of video data. The problem is one of computer vision, more specifically object recognition. To classify the objects in the video data, the monitor needs to send a portion of it to the vehicles that act as supervisors in the FL implementation. Since the unlabelled video data from the UAV itself is useless, each consider each learner as having the data itself. As such, we are generally interested in keeping the supervisory contributions private and we do not invade the privacy of any piece of the data. In such a scenario the FL implementation becomes challenged by the heterogeneous non-IID data and computational equipment that can vary at each learner. Such FL networks introduce much bigger uncertainty with regards to the heterogeneity on all fronts as well as the introduction of very dynamic cohorts of learners that introduce additional impact in the learning process. Therefore, our goal is to test wireless resource allocation methods that use the number of epochs computed at each learner as a proxy metric for impact over the general FL model. As such this paper is strongly related to answering RQ2 and has negligible impact on the other two RQs.

To accelerate the learning process of the FL in the case of unbalanced non-IID data that is distributed among learners with various computational capabilities, we needed to account for asynchronous learning processes at each learner. Such heterogeneity implies partial work from each node that mandates the use of FedProx as a local optimizer. As such FedProx grants some freedom in reducing the effects of heterogeneity through adjusting the parameter μ that controls the intensity of the FedProx intensity in heterogeneity reduction. Given the difficult interaction between the heterogeneity and Fedprox, in this paper we propose three different ways of aiding the learning process: maximizing the total learning time across the whole network (MAX), minimizing the maximum learning contributions between the slowest and fastest learner through average anchored staleness(AAS), and finally a dynamic approach based on estimating the past contributions of each learner and allocating more resources towards the more significant contributors (ACT). To test the implementations, we create a custom distribution of the MNIST and FMNIST datasets among different learners where a single

category of the initial 10 is hidden only at a single learner. This category called a critical object (CO) can be essential in the object detection process and should thus be learnt with the same speed as rest of the classes. From the initial testing we found out that both the reactive approach ACT and FedProx were essential towards speedy system performance. Moreover, the AAS implementation provided excellent CO learning, at the price of slightly slower convergence for the general model, since it treats all learners as equals and has smaller computational requirements. In this way, the paper situated itself as the first one to treat FL heterogeneity with very limited knowledge over the data. Moreover, it is also the first to propose the AAS and ACT metrics for alleviating heterogeneity issues especially when combined with FedProx implementations.

Paper F: ***Federated Learning with a Drone Orchestrator: Path Planning for Minimized Staleness***

In this paper we take the computationally light metric AAS that was tested in the previous paper and strengthened its application to physically dynamic UAV systems. Here, the UAVBS still acts as an orchestrator of an FL implementation where the learners are spatially scattered on the ground and the UAV can fly horizontally within the limits of a CA to balance their needs. In this way, the actual FL schedule of within the duration of a single FL round (downlink-processing-uplink) is controlled through the path of the UAV. As the UAV traverses the path, it controls the distance to each learner, and hence the data rate. As in the previous paper, we use the number of epochs to calculate the average anchored staleness (AAS) and minimize it through path planning. AAS is agnostic to the underlying FL implementation, unlike the ACT one that requires understanding the FL task, and as such it provides a generalized metric of FL performance. Nonetheless, optimizing the trajectory for the entire cycle is a difficult task and requires the implementation of some sort of approximations done through convex theory or deep reinforcement learning. This work was done in collaboration with fellow partners in the PAINLESS project from The American College of Greece (ALBA). In this way, this paper is strongly dedicated towards answering RQ2 and briefly touches upon the RQ1.

The convex theory approach uses several approximation methods, such as a Taylor series, and an aggregate metric for the data rate such as fairness. The resulting approximated solution space was then solved through SCP. On the other hand, we designed an elaborate double deep Q-learning approach where we designed the horizontal space in a hexagonal with seven actions. Both proposed solutions were tested in three very specific scenarios with few nodes that are intended to challenge the trajectory optimization implementations. For all three scenarios, both of the proposed solutions perform better than a static station, such as a BS or a hovering drone (at an optimal altitude) with improvements in the range of 57% - 87.7%. Nonetheless, the RL implementation is superior over using SCP approach given the AAS optimization criteria. However, the SCP solutions require less computational power, especially when compared with the

RL training phase, and still manage to provide eligible solutions. As such, this work provides two perspectives towards serving scattered node FL networks, and addressing them through orchestrator mobility. This is an entirely novel look on FL and as such has contributed towards the transition to superior UAV orchestrated FL networks. Moreover, the RL implementation has a very distinct approach in designing the MDP for the trajectory optimization problem.

Paper G: *Fairness Based Energy-Efficient 3D Path Planning of a Portable Access Point: A Deep Reinforcement Learning Approach*

The goal of this paper is to advance the RL based trajectory optimization such as presented in the previous work. Moreover, this work omits the FL or distributed computing use case and focuses on individual ground nodes that request wireless services from the UAVBS. In such a use case, we value fairness and therefore introduced a novel optimization objective criterion named fair energy efficiency (FEE) that weighs in the energy aspect of the trajectory as well as the fair distribution of data rates between all ground nodes for the duration of the trajectory. Since our goal in this paper is to most accurately replicate the energy conditions that a UAV battery and aerodynamics would impose onto designing a trajectory, we use state-of-the-art aerodynamic models to calculate the energy impact of flying and the non-linear Peukert effect to estimate the impact of voltage and current onto battery capacity. This creates a more difficult non-convex problem with non-tractable constraints where the use of model-free RL implementations are the strongest. This work was enabled by our fellow partners in the PAINLESS project from The American College of Greece (ALBA). As such this paper primarily provides answers to RQ3 and it has small contributions to RQ1.

As it is imposed by RL implementations, firstly the space was defined as an MDP with continuous state an action space. Moreover, opposed to the previous paper that does only 2D trajectory optimization, this work also introduces the altitude and does 3D trajectory optimization. Considering this and all the factors that increase the difficulty of the problem, we use the contemporary implementation of actor-critic deep reinforcement learning called twin delayed deep deterministic policy gradient (TD3). The testing is done in two specific ways, a fixed user location scenario and a more generalized implementation of randomized user locations. The initial takeaway is that not considering the Peukert effect in the trajectory optimization, overestimates the air-time of the UAV which may lead to worse performance in real-world scenarios. Given a static UAV baseline, the FEE improvements expect gains of 33%, 198%, 216% for suburban, urban, and dense urban environments, respectively. As such, this work provides a robust and generalized contribution to the trajectory optimization problem for data transmission between spatially scattered ground nodes and a UAV under realistic energy constraints.

Paper H: *Sustainable Wireless Services with UAV Swarms Tailored to Renewable Energy Sources*

This final paper in the thesis touches upon providing wireless coverage beyond areas that require a single UAV, and may need a deployment of UAV swarms. Such requirements can be imposed by some off-grid scenarios where the electrical and communications infrastructure has failed or may have never existed. Due to this, the foundation of this paper is to supply power, in a sustainable manner, to a main battery located at the center of this CA that offers battery swapping service to each UAV at the end of their air-time. The sustainable energy sources considered in this paper were both solar and wind intensity, collected by photovoltaic panels and wind turbines, respectively. Since the feasibility of such system is defined directly in the financial cost of the collective items of the infrastructure, we use the capital expences (CAPEX) incurred by deploying such a system. This requires the involvement of several intricate models that interact through the energy spent by UAVs while flying. For the system to provide wireless services, we need to consider the way in which UAV positions are organized that maximizes the packing efficiency of each individual coverage area. Moreover, our work also considered the impact of wind intensity onto the energy spent by UAVs, that also provide interesting interaction given that the wind is used to to generate electricity. Considering this, the rate requirements of the UAVs on the ground influences how many UAVs need to participate in the swarm and therefore influence how much energy is being used by the communications service. This provides us with an energy profile for the communications service that needs to be satisfied by the mentioned energy sources. Therefore, the main goal of the paper is to maximize the coverage offered by the swarm while minimizing the CAPEX of the entire system. This is a multi-variate optimization problem that is based on a lot of empirically derived data for each system involved in the analysis. The work in this paper was substantial and was therefore done in collaboration with fellow partners in the PAINLESS project from three institutions, Lyra Electronics Ltd., The University of Manchester, and The American College of Greece (ALBA). As such, this paper is focused on directly answering RQ3 and has negligible impact in answering some aspects of RQ1.

To solve the multi-variate problem, we propose a customized algorithm that combines several classical algorithms that address the complexity of the problem and the size of the solution space. To reduce the solution space of the problem, we use greedy sampling of the solution space that only investigates the best performing coverage areas. Moreover, to reduce the processing complexity we use several techniques such as binary search to better evaluate the bounds that are based on empirical data. For testing we used real world wind and solar data at four different locations in the world that have varying wind or solar intensity. The initial results show an interesting interaction based on the wind intensity. Since both the energy generation and the energy expenditure of the entire system is both non-linear and not a monotonous function of wind intensity there are benefits and drawback to deploying UAV swarms in windy locations. While the cost

of placing such a system in very windy location greatly exceeds that of having reliable solar intensity, there are certain values of systems that seem viable. On the other hand, having reliable solar power in areas with some wind intensity guarantees better system operability and allows for less reliance on battery storage. As such, this work provides a very novel perspective on the deployment of UAVBS swarms in off-grid areas, as well as a quite effective way of finding the optimal parameters of the system.

3 Conclusion

3.1 Revisiting the Objective

Following the research work and the publishing of numerous scientific papers, it is important to revisit the initial research objective and questions. Initially, it is important to re-evaluate the core aim of the thesis and provide several insights that were acquired in hindsight to the overall topic of UAVs and their application. Judging by the progress of literature within the last three years, it is has been correct to assume the UAV's position as the pivotal resource in a plethora of resource optimization problems concerning UAVs. Although there are existing generalized models for the applications of UAVs, there are many valid research directions that are derivations of the UAV positioning problem applicable to many different fields. While the work on UAV applications strictly for wireless communications service applications has been one of many derivative research topics, the implementations of FL in a UAV setting were one of the first of their kind in the topic of UAVs and FL. As such, the FL works have received significantly more attention since their publishing. Finally, the energy aspect of UAV implementation, while necessary, is mostly neglected in relevant literature. As such, early works in this thesis mostly neglected this aspect and a thorough investigation on trajectory optimization for energy balancing and solving the economics of the energy problem have been addressed. In hindsight, applying more energy-aware solutions early in the thesis would have been beneficial to include and improve the quality of early research. Nonetheless, this would have been a very complex and difficult pursuit that may have slowed or diminished the impact of the published research. Thus, we give a sequential summary of the contribution in each research question, as well as the incremental improvements to the overall research topic.

RQ1 – The question on communication services

As previously shown, RQ1 carries the most significance within the thesis since all other services are dependent on the wireless connectivity with and through the UAV, and was the first one to be investigated. The results achieved from Paper A were very useful to provide a better starting point for future research on all RQs. More specifically, this paper directly answered questions for addressing UAVBS mobility in a predefined

cellular area. The key takeaway from Paper A is that antenna directivity, mobility and user positions impact the feasibility of UAVBSs. Since antenna directivity controls the optimal altitude at which the UAVBS should fly at, it directly decreases the usefulness of UAVBS mobility. Moreover, it is concluded that at a certain antenna directivity mobility stops being a factor and UAVBSs can be replaced with low earth orbit (LEO) satellites, or HAPs. This raised a more important question on UAVs with regards to their purpose and usefulness, and the importance for finding an adequate motivation for UAVBS systems. As such, the following Paper B has shown a useful implementation of the UAVBS system for simultaneous service of high-priority eMBB traffic and non-priority mMTC users. Such an implementation is capable of introducing the concept of slicing by using both the wireless resources, and the UAV's position as a resources to be optimized when considering scattered eMBB nodes. This paper provides direct answers for the sub question to exploiting UAV mobility for serving users with various service requirements.

The sequential research dedicated to RQ1 provided more insight in mission critical communications for ground users. Paper C tested a key use-case for UAVBS implementations, and concluded that highly reliable communications are possible with existing technologies (a UAV with WiFi as fronthaul and LTE as backhaul). Moreover, such an implementation provided drastic improvements in communications delay over using a direct link to the LTE base station. The likely use case for such implementation is rescuing robotics equipment that has lost reliable connectivity with the established infrastructure by providing catered wireless access. Nonetheless, in order to guarantee reliability for mission critical operations or cyber-physical control systems a well known approach is to introduce additional diversity in wireless access. Paper D investigated using both WiFi and LTE connections simultaneously to improve the resistance to burst packet errors. Such implementations can help a GN improve reliability at a very low cost to energy.

RQ2 – The question on distributed computation

Once the implications of UAV positioning onto the wireless services were cleared through careful literature review and the aforementioned research, FL specific roles of UAVBSs were considered. Paper E considered a full implementation of proximal FL on a UAV as an orchestrator with the main goal to answer if UAVs are suitable orchestrators for FL. Thorough investigation on FL convergence with realistic privacy considerations showed that fairness is a strong factor on FL implementations. The reactive approach, that evaluates the contributions of learners without peeking into the dataset, was proven more effective than fairness in improving FL convergence. However, learner evaluation was computationally costly and average anchored staleness was more applicable for UAV orchestrators since it reduced the dependence on computational hardware. As such, Paper F proved the usefulness of designing the UAV trajectory for fair distribution of FL and provided a definite answer to RQ2. In summary, using trajectory optimization

for the purpose of addressing AAS in FL implementation proved as a simple and clear solution to improving FL convergence where the FL network is orchestrated by the UAV.

RQ3 – The question on energy balancing

Using the conclusions from previous research required to answer RQ1 and RQ2, the thesis was dedicated towards using UAVs for providing more fair communication services using UAVs while addressing the previously established energy concerns. Paper G directly contributed to solving the trajectory problem for maximizing fair distribution of wireless transferred bits while minimizing the energy spent for that flight. One early takeaway from this paper is that the energy profile does not change monotonically with the velocity of the UAV. In detail, hovering multi-copter UAVs consume a lot of power. However, flying horizontally at mild speeds consumes less power until the drag coefficient becomes a factor to the aerodynamics of the UAV. Due to this, all solutions provided by the reinforcement learning TD3 implementation keep the UAV flying horizontally instead of only hovering above nodes. Paper H is the last paper of the thesis, and as such combines the takeaways from most previous work to construct a more detailed and scaled up implementation of UAV swarms. In this paper, similarly as in Paper G, the velocity of flight impacted the energy consumption by the swarm, however here, we can also account for the impact of wind and altitude onto the implementation. Thus hovering in the presence of wind mimics flying at a constant horizontal velocity. Using wind and solar power generation stored in a ground battery, Paper H extracted the CAPEX feasibility of the system in different environments with varying wind and solar. The results show that the deployment feasibility for windy environments is harder to justify in smaller deployments, while mild wind conditions with reliable solar irradiation are feasible for many different swarm sizes. Finally, Paper H showed an interesting interaction between the impact of wind onto the energy generation and energy draw. Since the wind intensity impacts both the energy generation and expenditure profiles, in a unique and intricate manner. This complex interaction of the energy generation and expenditure system inspires ample future investigations in the compatibility of the systems in windy areas with various wind profiles.

3.2 Forthcoming Research Opportunities

While the thesis managed to answer the laid out RQs in detail and extract additional conclusions outside the scope of the research objective, it also inspired multiple future research directions. Although RQ1 receives plenty of attention from the literature there is insufficient experimental and practical support to the system of UAVBSs. A possible future topic is expanding the testbed from Paper C to test more energy aware positioning and automated deployments. Moreover, follow-up answers on the interaction between papers C and D are essential to understand the role of backhaul in a UAVBS versus

a direct link from the GN to the infrastructure. This would allow to understand the correlation between the ground and air link by the GN and UAV respectively. On the other hand, as FL is a young topic it has many research directions that are yet to receive the proper attention. As such Papers E and F inspire more streamlined testbeds with practical FL implementations such as the one that served as motivation in both papers. Since ML tasks require a lot of data to be collected, such future works may consume a lot of effort to provide productive and unbiased results. More FL specific research would also need to be addressed such as solving the issue of assymetric channels where i.e. downlink capacity may be significantly larger than uplink capacity. Paper E gave way to investigate an interesting trade-off in FL between overfitting, number of ML parameters, and task complexity. Finally, the investigation of energy in RG3, while it has the biggest historical support (in the form of early aeronautical research) it still needs to be adequately tested for the specific purpose of UAVBSs. This gives way for a lot of engineering opportunities to discover the optimal design of a UAVBS. To summarize, nearly all relevant future directions with regards to UAVBSs need to act in the form of verification for the massive theoretical ground work being performed by the UAVBSs and discover other engineering problems that may arise.

References

[1] C. W. Niessen, "Battlefield connectivity via airborne communications nodes," in *Digitization of the Battlefield II*, vol. 3080. SPIE, 1997, pp. 72–78.

[2] G. Djuknic, J. Freidenfelds, and Y. Okunev, "Establishing wireless communications services via high-altitude aeronautical platforms: a concept whose time has come?" *IEEE Communications Magazine*, vol. 35, no. 9, pp. 128–135, 1997.

[3] M. Raissi-Dehkordi, K. Chandrashekar, and J. S. Baras, "Uav placement for enhanced connectivity in wireless ad-hoc networks," Tech. Rep., 2004.

[4] M. Mozaffari, W. Saad, M. Bennis, Y. Nam, and M. Debbah, "A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems," *IEEE Communications Surveys Tutorials*, vol. 21, no. 3, pp. 2334–2360, Mar. , 2019.

[5] A. Fotouhi, H. Qiang, M. Ding, M. Hassan, L. G. Giordano, A. Garcia-Rodriguez, and J. Yuan, "Survey on UAV Cellular Communications: Practical Aspects, Standardization Advancements, Regulation, and Security Challenges," *IEEE Communications Surveys Tutorials*, vol. 21, no. 4, pp. 3417–3442, Mar. , 2019.

[6] Z. Yang, M. Chen, K.-K. Wong, H. V. Poor, and S. Cui, "Federated learning for 6g: Applications, challenges, and opportunities," *arXiv preprint arXiv:2101.01338*.

[7] W. Saad, M. Bennis, and M. Chen, "A vision of 6g wireless systems: Applications, trends, technologies, and open research problems," *IEEE network*, vol. 34, no. 3, pp. 134–142, Oct. , 2019.

[8] A. Al-Hourani, S. Kandeepan, and A. Jamalipour, "Modeling Air-to-Ground Path Loss for Low Altitude Platforms in Urban Environments," in *Proc. of IEEE Global Communications Conference*, Austin, TX, Oct. , 2014, pp. 2898–2904.

[9] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP Altitude for Maximum Coverage," *IEEE Wireless Communications Letters*, vol. 3, no. 6, pp. 569–572, Dec. , 2014.

[10] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage," *IEEE Communications Letters*, vol. 20, no. 8, pp. 1647–1650, 2016.

[11] S. Jeong, O. Simeone, and J. Kang, "Mobile Edge Computing via a UAV-Mounted Cloudlet: Optimization of Bit Allocation and Path Planning," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 3, pp. 2049–2063, Mar. , 2018.

[12] M. Li, N. Cheng, J. Gao, Y. Wang, L. Zhao, and X. Shen, "Energy-Efficient UAV-Assisted Mobile Edge Computing: Resource Allocation and Trajectory Optimization," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 3, pp. 3424–3438, March. , 2020.

[13] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated Learning: Strategies for Improving Communication Efficiency," *arXiv preprint arXiv:1610.05492*, Oct. 2016.

[14] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *arXiv preprint arXiv:1812.06127*, Dec. , 2018.

[15] B. Brik, A. Ksentini, and M. Bouaziz, "Federated learning for uavs-enabled wireless networks: Use cases, challenges, and open problems," *IEEE Access*, vol. 8, pp. 53 841–53 849, 2020.

[16] J. Wang, Z. Charles, Z. Xu, G. Joshi, H. B. McMahan, M. Al-Shedivat, G. Andrew, S. Avestimehr, K. Daly, D. Data *et al.*, "A field guide to federated optimization," *arXiv preprint arXiv:2107.06917*, 2021.

[17] M. M. Amiri and D. Gündüz, "Federated learning over wireless fading channels," *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, pp. 3546–3557, Feb. 2020.

[18] M. Alzenad, A. El-Keyi, F. Lagum, and H. Yanikomeroglu, "3-D Placement of an Unmanned Aerial Vehicle Base Station (UAV-BS) for Energy-Efficient Maximal Coverage," *IEEE Wireless Communications Letters*, vol. 6, no. 4, pp. 434–437, Aug. , 2017.

[19] P. Popovski, K. F. Trillingsgaard, O. Simeone, and G. Durisi, "5g wireless network slicing for embb, urllc, and mmtc: A communication-theoretic view," *Ieee Access*, vol. 6, pp. 55 765–55 779, Spt. 2018.

[20] C. She, C. Liu, T. Q. Quek, C. Yang, and Y. Li, "Ultra-Reliable and Low-Latency Communications in Unmanned Aerial Vehicle Communication Systems," *IEEE Transactions on Communications*, vol. 67, no. 5, pp. 3768–3781, May. , 2019.

[21] P. Popovski, J. J. Nielsen, C. Stefanovic, E. De Carvalho, E. Strom, K. F. Trillingsgaard, A.-S. Bana, D. M. Kim, R. Kotaba, J. Park *et al.*, "Wireless access for ultra-reliable low-latency communication: Principles and building blocks," *Ieee Network*, vol. 32, no. 2, pp. 16–23, Apr. 2018.

[22] 3GPP, "Technical specification TS 22.104 V16.5.0. 5G; Service requirements for cyber-physical control applications in vertical domains," 2021.

[23] P. M. de Sant Ana, N. Marchenko, P. Popovski, and B. Soret, "Wireless control of autonomous guided vehicle using reinforcement learning," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, dec 2020.

[24] J. J. Nielsen, R. Liu, and P. Popovski, "Ultra-reliable low latency communication using interface diversity," *IEEE Transactions on Communications*, vol. 66, no. 3, pp. 1322–1334, 2018.

[25] V. Smith, C.-K. Chiang, M. Sanjabi, and A. S. Talwalkar, "Federated multi-task learning," in *in Proc. of Advances in neural information processing systems*, 2017, pp. 4424–4434.

[26] T. Zeng, O. Semiari, M. Mozaffari, M. Chen, W. Saad, and M. Bennis, "Federated Learning in the Sky: Joint Power Allocation and Scheduling with UAV Swarms," in *Proc. of the IEEE International Conference on Communications (ICC), Next-Generation Networking and Internet Symposium*, Dublin, Ireland, June. , 2020, pp. 1–6.

[27] N. Babu, M. Virgili, C. B. Papadias, P. Popovski, and A. J. Forsyth, "Cost- and Energy-Efficient Aerial Communication Networks with Interleaved Hovering and Flying," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 9, pp. 9077–9087, Jul. 2021.

[28] M. Virgili, A. J. Forysth, and P. James, "A Multi-Objective Genetic Algorithm Methodology for the Design of Standalone Energy Systems," in *2021 IEEE Design Methodologies Conference (DMC)*, Jul. 2021, pp. 1–6.

Part II

Papers

Paper A

Dynamic Standalone Drone-Mounted Small Cells

Igor Donevski, Jimmy Jessen Nielsen

The paper has been published in the
Proceedings of 2020 European Conference on Networks and Communications (EuCNC)
pp. 342–347, 2020.

The layout has been revised.

Abstract

This paper investigates the feasibility of Dynamic Horizontal Opportunistic Positioning (D-HOP) use in Drone Small Cells (DSCs), with a central analysis on the impact of antenna equipment efficiency onto the optimal DSC altitude that has been chosen in favor of maximizing coverage. We extend the common urban propagation model of an isotropic antenna to account for a directional antenna, making it dependent on the antenna's ability to fit the ideal propagation pattern. This leads us to define a closed-form expression for calculating the Rate improvement of D-HOP implementations that maintain constant coverage through antenna tilting. Assuming full knowledge of the uniformly distributed active users' locations, three D-HOP techniques were tested: in the center of the Smallest Bounding Circle (SBC); the point of Maximum Aggregated Rate (MAR); and the Center-Most Point (CMP) out of the two aforementioned. Through analytic study and simulation we infer that DSC D-HOP implementations are feasible when using electrically small and tiltable antennas. Nonetheless, it is possible to achieve average per user average rate increases of up to 20-35% in low user density scenarios, or 3-5% in user-dense scenarios, even when using efficient antennas in a DSC that has been designed for standalone coverage.

1 Introduction

Drone, a.k.a., UAV (Unmanned Aerial Vehicle) usage has excelled in the last decade due to commercial demand for consumer uses such as: photography, entertainment and payload delivery, or public service uses such as search and rescue. In the world of wireless communications, because of their mobility and modularity, the flying devices are considered useful as drone mounted access points that provide or improve localized communication quality. In accord, the need for airborne base stations has been accentuated in the last five years, as it can be noticed from the overabundance in scientific and standardization activity [1, 2]. The concept has been identified as useful in diverse use cases: disaster recovery missions, failures of the main infrastructure, coverage assistance [3] for traffic surges, or sinks for Internet of Things (IoT) devices [4].

1.1 Goals and Motivation

Drones are eligible implementations of Small Cells (DSCs) that offer good coverage in urban areas. Compared to traditional cellular networks, DSCs avoid strong signal shadowing because they are positioned high relative to the user equipment. This benefit is inherent to all Low or High Altitude Platforms (LAPs and HAPs), where DSCs can be considered as a subcategory of LAPs. In accord, there has been a plethora of publications explaining and exploiting the channel improvements due to the relatively high altitude [5–7, 9].

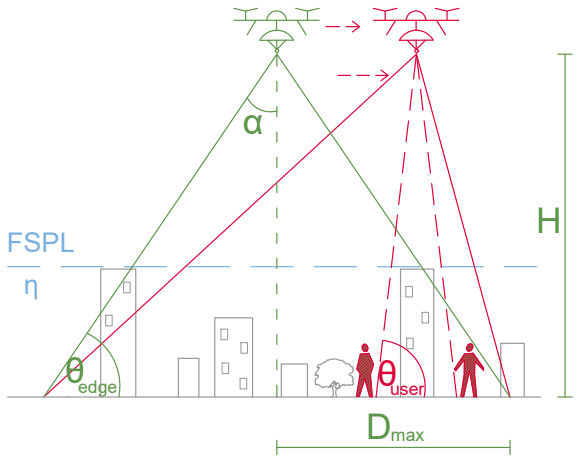


Fig. A.1: Reference cell placement (green), and DSC with D-HOP implementation active (red).

The motivation behind using DSCs is that they offer an alternative that avoids infrastructure costs. In fact, 52% of the Mobile Network Operator's (MNO) CAPEX is spent on site acquisition and construction; in addition to which, site rental dominates the MNO's OPEX, and is increasingly more expensive, with up to 42% in developed countries [1]. Unfortunately, real world deployment feasibility is challenged by the weight and energy limitations to UAV air-time. Dedicated models, designed solely for this purpose should manage to minimize the impact of these drawbacks and rationalize the financial input of investing.

For this purpose, we design a DSC system model that uses a directional antenna in combination with Dynamic Horizontal Opportunistic Positioning (D-HOP) techniques. We consider both elements of the model essential to the concept of DSCs, and not accounting for either will result in misplaced resources and/or unused opportunity. We illustrate a toy example of our approach on Fig. A.1. With green we show a reference deployment of a DSC with no D-HOP applied. Here, the height H and radius $D_{\max}$ are defined during the planning phase in favour of standalone coverage maximization, and are dependent on the region topography and the antenna efficiency to fit the beamwidth of $2 \cdot \alpha$. With red we show how the DSC would apply D-HOP in attempts of avoiding shadowing for two active customers, all the while, it tilts the antenna to maintain a constant coverage of the initially designated area. Our model is based solely on the likelihood of encountering a strong shadowing effect for a specific type of terrestrial topology instead of having full knowledge of the regional infrastructure; since we expect deployments like this need to be easy to deploy, and apply to natural disaster scenarios

as well [8].

1.2 Relation to State of the Art

This is the first work to investigate the combination and compatibility of: using a directional transmitter, the impact of the antenna's efficiency in fitting the beamwidth requirements, and most importantly, its impact on D-HOP improvements for active users. Within this work, we test the performance of three D-HOP techniques, and discuss on choosing the most adequate one for Standalone DSCs. As such, the antenna efficiency has a central role to D-HOP usefulness. The works of [10–12], although considering directional antennas, do not account for the impact of the efficiency of the transmitter, are mainly concerned with 3D placement of the drone and omit analysis on the benefits of different D-HOPs. On the other hand, the works in [3] and [7] are mainly concerned with the location of the drone, assume that they do operate in a non-standalone manner, and omit the impact of directional transmitters altogether. With this, we hope to introduce the reader to the potential benefits and implementation complexities that concern deployment of D-HOP enabled standalone DSCs.

The paper is organized as follows. Sec. 2 describes the considered RAN scenario, explains the directional antenna model, and produces a closed-form expression of the D-HOP model. Sec. 3 explains the tested D-HOP techniques. Sec. 4 presents numerical results from the performed simulations, and discusses the outcomes. Finally, Sec. 5 concludes with the impact of this work.

2 System Model

We begin by assuming that the backhaul link is over-provisioned, and does not have any additional requirements that need to be accounted for. To evaluate the effectiveness of the positioning algorithms in a scenario where the DSCs act as solitary wireless service nodes, we need to assume that the equipment successfully maintains coverage at all times over the whole designated cell with radius $D_{\max}$ by mechanically or electronically tilting the antenna.

2.1 Propagation Model

Looking at Fig. A.1 we notice that buildings may obstruct a user's direct link towards the DSC. We therefore consider the user as belonging to one of two propagation groups, users with Line of Sight (LoS) and No LoS (NLoS) [5].

To express the likelihood of a user device belonging to either of the propagation groups we require a model for the LoS probability. In this service, the ITU has created a model [13] that can be approximated to an s-curve defined by two topology constants a and b as a function of the elevation angle at user side θ_{user} that is expressed in degrees

$0 \leq \theta_{\text{user}} \leq 90$. The model defined in Eq. (A.1), provides good approximations of the ITU model for calculating $P(\text{LoS})$, that is especially accurate for urban scenarios [6].

$$P(\text{LoS}) = \frac{1}{1 + a\exp(-b(\theta_{\text{user}} - a))} \tag{A.1}$$

As shown on Fig. A.1 we define the total path loss as a combination of Free Space Path Loss (FSPL) and the expected shadowing coefficient for each of the propagation groups η_{LoS} and η_{NLoS}. These values represent the means of the normally distributed excessive path loss, that is induced due to the large features of the topology in LoS and NLoS, respectively. Assuming a directional antenna with directivity measure D_t is mounted on the drone, we can define the path loss per propagation group as:

$$L_{\text{LoS}} = -10\log(D_t) + 20\log(d) + 20\log(\frac{f4\pi}{c}) + \eta_{\text{LoS}} \tag{A.2}$$

$$L_{\text{NLoS}} = -10\log(D_t) + 20\log(d) + 20\log(\frac{f4\pi}{c}) + \eta_{\text{NLoS}} \tag{A.3}$$

Knowing $P(\text{LoS})$ and $P(\text{NLoS}) = 1 - P(\text{LoS})$, we can continue to find the expected path loss Λ as:

$$10\log(\Lambda) = L_{\text{LoS}} \cdot P(\text{LoS}) + L_{\text{NLoS}} \cdot P(\text{NLoS}) \tag{A.4}$$

$$= P(\text{LoS})(\eta_{\text{LoS}} - \eta_{\text{NLoS}}) + L_{\text{NLoS}} \tag{A.5}$$

2.2 Antenna Gain

Going back to Fig. A.1 we observe that the usage of a directional antenna at drone-side requires that we fix the proportions of our main lobe to fit the size of the apex angle $2\cdot\alpha$, which fixes the value of θ_{edge} as well. This is done with two main arguments. Firstly, a transmitter limited to its own cell will not contribute to the interference in other cells, therefore diminishing its negative impact. Secondly, using a directional antenna is a practical way of boosting the signal strength within the designated area, while ignoring the users outside the defined borders. In this way, a combination of multiple DSCs can be used, while avoiding strong inter-cell interference, as shown in [9].

We first analyze the directivity D_t element in eqs. (A.2) and (A.3) as an ideal antenna D_I that perfectly covers the designated circular area. We define a value $0 \leq E_r \leq 1$ that is dependent on the type, manufacturing and quality of the antenna, to measure the efficiency of our implemented antenna in reference to the ideal one $D_t = D_I^{E_r}$ for our purpose. This measures the strength of the Main Lobe with relation to the spread outside the assigned coverage area due to sidelobes, inadequate main lobe size or other imperfections. The most adequate antenna type for our application includes, but is not

limited to, phased array antennas, as they are able to quickly adjust the direction of the main lobe. From here we first calculate D_I as [16]:

$$D_I = \frac{4\pi}{\Omega} \tag{A.6}$$

D_I is defined by a beamwidth defined by the solid angle of a perfect cone as $\Omega = 2\pi(1-\cos(\alpha))$. The angle α is half of the apex angle in either of the two two-dimensional propagation planes, as shown in Fig. A.1. By further applying simple trigonometry to our UAV scenario, we reach Eq. (A.7) as the final metric, which is dependent on the elevation angle on the edge of the cell θ_{edge}.

$$D_I = \frac{2}{1-\sin(\theta_{\text{edge}}\frac{\pi}{180})} \tag{A.7}$$

Including this in the final equation for expectation of path loss and expressing all in terms of $D_{\max} > 0$, θ_{user} and θ_{edge} expressed in degrees we get:

$$\begin{aligned} 10\log(\Lambda) = & \\ & \frac{\eta_{\text{LoS}} - \eta_{\text{NLoS}}}{1 + a\exp(-b[\theta_{\text{user}} - a])} + 20\log(\frac{D_{\max}}{\cos(\theta_{\text{user}}\frac{\pi}{180})}) \\ & - E_r 10\log(\frac{2}{1-\sin(\theta_{\text{edge}}\frac{\pi}{180})}) + 20\log(\frac{f4\pi}{c}) + \eta_{\text{NLoS}} \end{aligned} \tag{A.8}$$

2.3 Cell Size and θ_{edge}

We define the cell size by assuming a maximally allowed expected path loss at the point of worst case coverage when the user is located at the cell edge $\theta_{\text{user}} = \theta_{\text{edge}} = \theta$. We analyze the communication in an information theoretic manner and define the expected rate as in Eq. (A.9) in terms of bits/symbol. Since we are maximizing the cell coverage we are interested in meeting the average rate requirement:

$$R_{\text{avg}} = \log_2(1 + \frac{P_{\text{tx}}}{N_0\Lambda}) \tag{A.9}$$

In the following we assume for simplcity that $R_{\text{avg}} = 1$, which gives that the drone needs to be positioned at the point where $\Lambda = \frac{P_{\text{tx}}}{N_0}$, and we maximize the radius of the cell $D_{\max}$ in favor of coverage. In this way, the elevation angle at the edge of the cell is chosen to constrain the radiation radius to the planned radius of the cell but also makes sure to avoid discriminating the users located at the edge of the cell. Going back to Eq. (A.8) to find the maximal radius as a function of the elevation angle $\frac{dD_{\max}}{d\theta}$, we conclude that it is only dependent on the scenario topography relative to our operating

frequency and the efficiency of our antenna, as shown in the derivation in Eq. (A.10).

$$0 = \frac{\pi \tan\left(\theta \frac{\pi}{180}\right)}{9 \log(10)} + \frac{a\, bA \exp(-b(\theta - a))}{a \exp(-b(\theta - a) + 1)^2} - E_r \frac{\pi \cos\left(\theta \frac{\pi}{180}\right)}{18 \log(10)(1 - \sin\left(\theta \frac{\pi}{180}\right))} \quad \text{(A.10)}$$

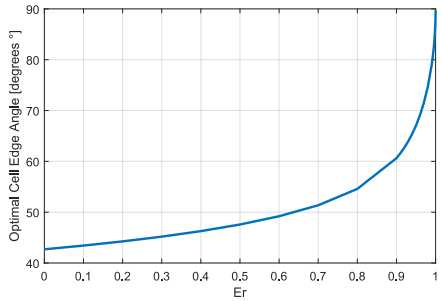


Fig. A.2: The optimal cell edge elevation angle θ_{edge} as a function of E_r for Urban Scenario parameters given in Sec. 4.

Analysing eq. (A.10) from the perspective of an urban scenario we investigate the impact of E_r onto the optimal θ. On Fig. A.2 we follow the evolution of θ_{edge} in the cases of different antenna radiation efficiency. The two extremes set for E_r are 0 for an ideal isotropic antenna and 1 for an ideal conical antenna. Having $E_r = 0$ coefficient cancels the D_t member in Eq. (A.10) and the final θ_{edge} is identical to the one of using an isotropic antenna. Having $E_r = 1$ breaks the point of optimality in Eq. (A.10). In other words, when possessing an ideal transmitter we should be able to establish point to point transmissions with infinitely big gain, meaning that E_r should be strictly smaller than 1 [16]. This dependence of θ_{edge} on E_r defines the principal contribution of our model, and it shows the important details that MNOs need to pay attention to when designing the geometry of the system.

2.4 User Data Rate

In a circular coverage area with radius $D_{\max}$ there are N active users, each i$^{\text{th}}$ user's location within the area is defined with two coordinates (x_i, y_i), where $i \in 1, 2...N$, and the drone occupies a position with coordinates $(x_{\text{d}}, y_{\text{d}})$. We define each user's horizontal distance to the drone with $d_i = \sqrt{(x_i - x_{\text{d}})^2 + (y_i - y_{\text{d}})^2}$ and define a scalar κ_i that represents the distance in relation to the cell radius $d_i = \kappa_i \cdot D_{\max}$, where $2 \geq \kappa_i \geq 0$.

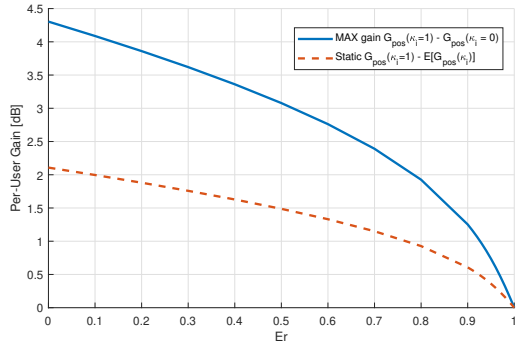


Fig. A.3: Analysis on maximum possible D-HOP gain as a function of E_r.

We reformulate our expected path loss in eq. (A.8) in terms of κ_i, as:

$$10\log(\Lambda) = \frac{\eta_{\text{LoS}} - \eta_{\text{NLoS}}}{1 + a\exp(-b[\theta_{\text{user}} - a])} + 20\log(\sqrt{\kappa_i^2 + \tan(\theta_{\text{edge}}\frac{\pi}{180})^2}) + 20\log(D_{\max}) + C \quad \text{(A.11)}$$

Where θ_{user} is dependent on κ_i and is $\theta_{\text{user}} = \arctan(\frac{h}{\kappa_i D_{\max}}) = \arctan(\frac{\tan(\theta_{\text{edge}}\frac{\pi}{180})}{\kappa_i})$; and parameters that are independent of κ_i or $D_{\max}$ are $C = 20\log(\frac{f4\pi}{c}) + \eta_{\text{NLoS}} - E_r 10\log(D_I)$. Additionally, eq. (A.11) is continuous over the whole range of possible user positions as well as $\kappa_i \geq 0$. For convenience, we group all members of the equation that depend on κ_i to define the horizontal repositioning gain G_{pos} as:

$$G_{\text{pos}}(\kappa_i) = \frac{\eta_{\text{LoS}} - \eta_{\text{NLoS}}}{1 + a\exp(-b[\theta_{\text{user}} - a])} \quad \text{(A.12)}$$

$$+ 10\log(\kappa_i^2 + \tan(\theta_{\text{edge}}\frac{\pi}{180})^2) \quad \text{(A.13)}$$

And receive a final, closed form equation for the expected pathloss of user at distance κ_i in a cell with radius $D_{\max}$ as:

$$10\log(\Lambda(\kappa_i, D_{\max})) = G(\kappa_i) + 20\log(D_{\max}) + 20\log(\frac{f4\pi}{c}) + \eta_{\text{NLoS}} - E_r 10\log(\frac{2}{1 - \sin(\theta_{\text{edge}}\frac{\pi}{180})}) \quad \text{(A.14)}$$

This results in the per-user expected rate being:

$$R_i = \log_2(1 + \frac{\Lambda(\kappa_i = 1, D_{\max})}{\Lambda(\kappa_i, D_{\max})}) \tag{A.15}$$

$$= \log_2(1 + 10^{\frac{G_{\text{pos}}(\kappa_i=1) - G_{\text{pos}}(\kappa_i)}{10}}) \tag{A.16}$$

From the involved parameters, it is obvious that the benefits from D-HOP implementation depend on the cell's geometrical proportions and not on its absolute size. Additionally, on Fig. A.3 we show how the span of possible D-HOP gains evolve as a consequence of the behaviour shown in Fig. A.2 that is dependent on the antenna's efficiency. Since all aforementioned parameters are predefined when planning the communications model, in the next section we focus on lowering the values for κ_i through the means of D-HOP.

3 D-HOP of Drone Small Cells

We open this section by stressing that Drone D-HOP mobility does not affect the coverage area; as it can adjust its antenna angle (physically/electronically) to fully cover the area with radius $D_{\max}$, and inactive users in the area can announce their location and activation time with a rate $R_{\text{acc}} \ll R_{\text{avg}}$. Distortions of the circular coverage field to an oval one due to the angle of the transmitter are considered to be negligible. From here, our goal is to improve the channel condition for the active users in the area without neglecting new requests. We do this by performing constant dynamic movements in the horizontal plane with height $H = D_{\max} \tan(\theta_{\text{edge}})$.

In an arrangement of user locations our goal is to optimize the drone position $(x_{\text{d}}, y_{\text{d}})$ and we identify two significant points, one performance maximization oriented, and the other as fairness oriented. We illustrate this in Fig. A.4.

3.1 Smallest Bounding Circle (SBC)

One approach that carries great geometric significance, is to set the DSC location in the center of the minimum bounding circle of all active users, which is the smallest circle that contains all points inside [7]. With this, the goal is to maximize the fairness of the dynamic system by minimizing $d_{\text{f}} = \max[d_i] \; \forall \; i$. The minimum bounding circle is a well known computational geometry problem falling under the umbrella of facility location, or the 1-center problem.

3.2 Maximum Aggregated Rate (MAR)

The second position of geometric significance is where drone placement would achieve minimal total distance to all active users Min: $d_{\text{m}} = \sum_i d_i$, or, the centroid of all points.

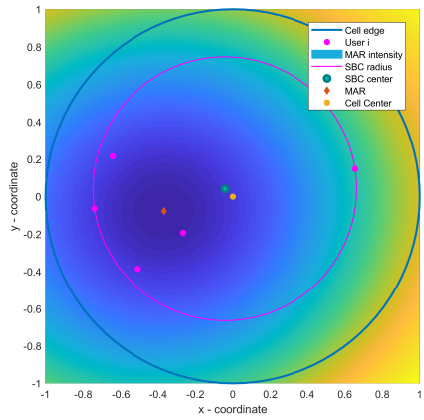


Fig. A.4: An illustration of a single D-HOP scenario, over MAR intensity map.

Although the SBC d_{f} is a universal fairness maximization approach, the centroid d_{m} is not, and we substitute it for a more adequate performance parameter. In its stead we use the aggregate rate improvement, as:

$$\max_{x_{\mathrm{d}}, y_{\mathrm{d}}} \sum_{i=0}^{N} R_i \tag{A.17}$$

3.3 Center-Most Point (CMP)

Finally, limitations regarding the mobility of the UAV need to be taken into account as it cannot instantaneously relocate on every position with every shift in user behaviour. This requires inspecting a more travel distance conservative repositioning technique. In favor of this, we create a repositioning algorithm that puts the drone either at maximum gain, or maximum fairness, depending on which of both points is the Centermost Point. This is done knowing that if averaged over an infinite amount of users and timeslots, the optimal position of the drone is in the very center.

4 Simulation and Results

We consider a snapshot based, simplistic and replicable testing scenario where the positioning of the UE occurs in timeslots, and each timeslot has no correlation to the previous one. No assumptions are done with user mobility in mind, and each user can be

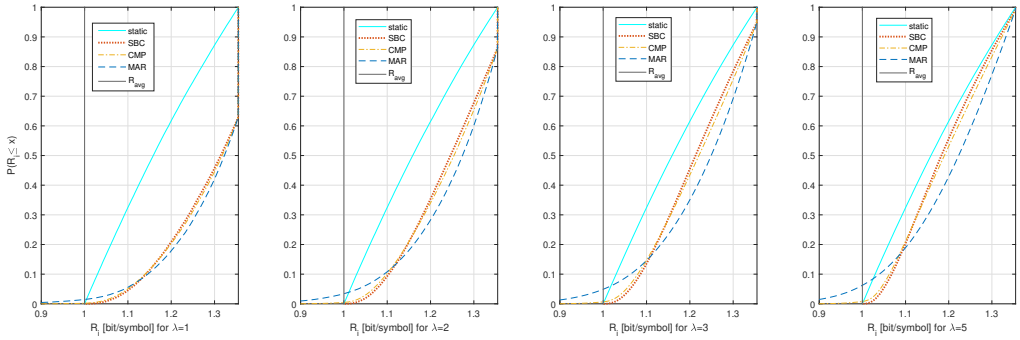


Fig. A.5: The CDF of the probability for a user to receive rates higher than R_{avg} tested for four different user densities.

uniformly located within the cell's limits. We test the system under the Urban scenario conditions, with parameters: $a = 9.61, b = 0.16, \eta_{\text{LoS}} = 1, \eta_{\text{NLoS}} = 20, f = 2$ GHz [5].

On Fig. A.3 we show that E_r has a strong impact on the feasibility of D-HOP implementations. Moreover, dynamic DSCs make more sense in use cases that require antennas with lower E_r. This is due to the dependency of the cell's edge optimal elevation angle θ_{edge} to the E_r coefficient. Luckily, this goes in favor of dynamic DSCs since lighter, cheaper, and electrically small antennas are expected at drone side. We, therefore arbitrarily set $E_r = 0.6$ as an example of an adequately chosen antenna guided by the comments of [16].

We first accentuate the importance of the expected geometric location of the users and that D-HOP finds purpose in unbalanced user formations, especially in the existence of clusters [3]. However, we do not account for user clustering, and we assume that if averaged over an infinite amount of timeslots, the optimal location for positioning the

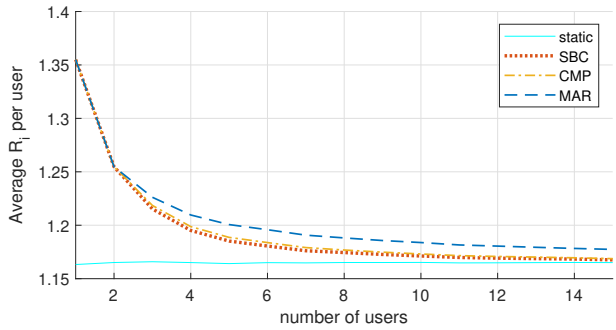


Fig. A.6: Per user expected rate, as a function of the number of active uniformly distributed users.

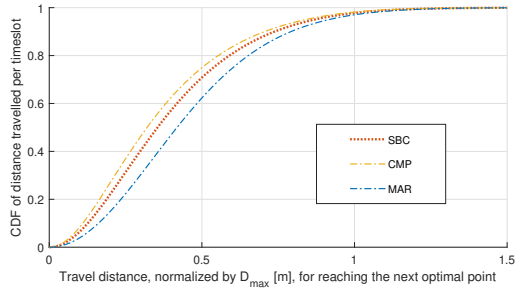


Fig. A.7: Per timeslot drone travel distance CDF, normalized by $D_{\max}$ in $[m]$, for user density of $\lambda = 5$.

drone is in the very center of the cell such as in a well planned cell placement. As a consequence, in the case when the drone takes no action and stays in the center of the cell, we still detect average rate improvements, with regards to the preset average rate R_{avg}, since the users are not always located at the cell's edge. This gain, shown on Figs. A.6 and A.3, is entirely due to the user location distribution, and is imperative for evaluating the usefulness of any repositioning technique.

From the simulation results at Fig. A.6 it is obvious that the more evenly distributed active users are present, the less we exploit the D-HOP advantages. Special cases consist of only two or less active users, where any of the proposed solutions would behave the same. Therefore, Dynamic DSC deployments are well suited for areas with low user density occurrences, and in the best cases can increase the expected per user rates by 17% with regards to static, or 34% for users at the cell edge. Which is a good result considering that we avoid changing the drone's altitude, in service of avoiding coverage holes.

In Fig. A.5, we show how each D-HOP approach modifies the distribution of the distances between the users and the drone and therefore achieve higher rates for most users. Here we imitate four instances of Poisson Point Process (PPP) where every timeslot has N users that are Poisson distributed $N \sim \text{Poisson}(\lambda)$. It is immediately noticeable that the MAR approach has the best performance on average, at the expense of putting roughly 5% of its users at distances $\kappa_i > 1$. Nonetheless, it may be feasible for systems with no hard coverage constraints since when compared to the static it improves the rates by 5.6% for the average user and offers 21.5% increase over the preset average rate. The CMP technique diminishes the impact of users at distances bigger than the Radius of the cell, however, it still does not evade all $\kappa_i > 1$ situations. The SBC approach obviously avoids having such cases at all, while it also improves the expected rate for the lowest fifth percentile of our users by 3% for $\lambda = 5$, and up to 10% for $\lambda = 1$. This makes the bounding circle ideal for DSCs that offer high reliability for all users within the cell.

To account for the limitations of the equipment, we investigate the distributions for displacement requirements per D-HOP. In Fig. A.7 we show that the distances travelled for the SBC and the CMP have an obvious advantage over the MAR approach. The advantage of using the CMP technique with regards to mobility requirements makes it an adequate solution for less rigorous reliability requirements, as it offers a balance between fairness and rate maximization mobility requirements for the drone.

Deriving from Figs. A.3 and A.5, and eq. (A.14) we conclude that the span of possible performance gains are predetermined by the system topology and E_r, while the user distribution and D-HOP technique impact the gains achieved within that span. Since the expected D-HOP gains do not scale with the cell size $D_{\max}$, its implementation is expected to have a higher impact in smaller cells. Additionally, drone travel distance requirements will demand much lesser drone speeds for smaller $D_{\max}$.

Both points reinforce the fact that DSC repositioning benefits from the added complexity in the cases of small or pico cells. For the existence of reliable communications we can conclude that the predefined parameters regarding the radius/altitude of the UAV position or the angle of coverage, are very strict, and fulfill the stringent link budget constraints to establish the desired reliability [14, 15]. Therefore, following this model, we ascertain the D-HOP DSC's inherent eligibility in reliability applications by calling for user-fairness techniques as most effective D-HOP implementations for Drone Small Cells.

5 Conclusions

This work investigates the concept of DSCs and accounts for exploiting all its advantages. We note the importance of knowing the efficiency of the available antenna equipment as it directly influences the optimal geometry of the model. We then quantify its impact over the dynamic repositioning gains, and conclude that the gains achieved from repositioning are mainly beneficial when using small antennas. For the tested urban scenario, we achieve per user average rate improvements of up to 20-35% in low-user density scenarios, or 3% - 5% in dense scenarios. Which for our model is extremely well considering we assume balanced and uniform user positions with standalone and constant coverage over the whole area.

Acknowledgment

The work was supported by the European Union's research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 812991 "PAINLESS" within the Horizon 2020 Program.

The authors would like to thank prof. Petar Popovski for his valuable guidance.

References

[1] A. Fotouhi et al., "Survey on UAV Cellular Communications: Practical Aspects, Standardization Advancements, Regulation, and Security Challenges," in IEEE Communications Surveys & Tutorials 2019.

[2] M. Mozaffari, W. Saad, M. Bennis, Y. Nam and M. Debbah, "A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems," in IEEE Communications Surveys & Tutorials, vol. 21, no. 3, pp. 2334-2360, thirdquarter 2019.

[3] A. Fotouhi, M. Ding and M. Hassan, "Dynamic Base Station Repositioning to Improve Performance of Drone Small Cells," 2016 IEEE Globecom Workshops (GC Wkshps), Washington, DC, 2016, pp. 1-6.

[4] M. Mozaffari, W. Saad, M. Bennis and M. Debbah, "Mobile Unmanned Aerial Vehicles (UAVs) for Energy-Efficient Internet of Things Communications," in IEEE Transactions on Wireless Communications, vol. 16, no. 11, pp. 7574-7589, Nov. 2017.

[5] A. Al-Hourani, S. Kandeepan and S. Lardner, "Optimal LAP Altitude for Maximum Coverage," in IEEE Wireless Communications Letters, vol. 3, no. 6, pp. 569-572, Dec. 2014.

[6] A. Al-Hourani, S. Kandeepan and A. Jamalipour, "Modeling air-to-ground path loss for low altitude platforms in urban environments," 2014 IEEE Global Communications Conference, Austin, TX, 2014, pp. 2898-2904.

[7] M. Alzenad, A. El-Keyi, F. Lagum and H. Yanikomeroglu, "3-D Placement of an Unmanned Aerial Vehicle Base Station (UAV-BS) for Energy-Efficient Maximal Coverage," in IEEE Wireless Communications Letters, vol. 6, no. 4, pp. 434-437, Aug. 2017.

[8] M. Erdelj, E. Natalizio, K. R. Chowdhury and I. F. Akyildiz, "Help from the Sky: Leveraging UAVs for Disaster Management," in IEEE Pervasive Computing, vol. 16, no. 1, pp. 24-32, Jan.-Mar. 2017.

[9] M. Mozaffari, W. Saad, M. Bennis and M. Debbah, "Efficient Deployment of Multiple Unmanned Aerial Vehicles for Optimal Wireless Coverage," in IEEE Communications Letters, vol. 20, no. 8, pp. 1647-1650, Aug. 2016.

[10] B. Galkin, J. Kibiłda and L. A. DaSilva, "A Stochastic Model for UAV Networks Positioned Above Demand Hotspots in Urban Environments," in IEEE Transactions on Vehicular Technology, vol. 68, no. 7, pp. 6985-6996, July 2019.

[11] Z. Yang et al., "Joint Altitude, Beamwidth, Location, and Bandwidth Optimization for UAV-Enabled Communications," in IEEE Communications Letters, vol. 22, no. 8, pp. 1716-1719, Aug. 2018.

[12] M. M. Azari, F. Rosas and S. Pollin, "Cellular Connectivity for UAVs: Network Modeling, Performance Analysis, and Design Guidelines," in IEEE Transactions on Wireless Communications, vol. 18, no. 7, pp. 3366-3381, July 2019.

[13] “Propagation data and prediction methods for the design of terrestrial broadband millimetric radio access systems,” IETF, Fremont, CA, USA, ITU-Recommendation p.1410-2, 2003

[14] P. Popovski et al., "Wireless Access for Ultra-Reliable Low-Latency Communication: Principles and Building Blocks," in IEEE Network, vol. 32, no. 2, pp. 16-23, March-April 2018.

[15] C. She, C. Liu, T. Q. S. Quek, C. Yang and Y. Li, "Ultra-Reliable and Low-Latency Communications in Unmanned Aerial Vehicle Communication Systems," in IEEE Transactions on Communications, vol. 67, no. 5, pp. 3768-3781, May 2019.

[16] W. L. Stutzman, "Estimating directivity and gain of antennas," in IEEE Antennas and Propagation Magazine, vol. 40, no. 4, pp. 7-11, Aug. 1998.

Paper B

Standalone Deployment of a Dynamic Drone Cell for Wireless Connectivity of Two Services

Igor Donevski, Jimmy Jessen Nielsen, Petar Popovski

The paper has been published in the
Proceedings of 2021 IEEE Wireless Communications and Networking Conference (WCNC) pp. 1–7, 2021.

The layout has been revised.

Abstract

We treat a setting in which two priority wireless service classes are offered in a given area by a drone small cell (DSC). Specifically, we consider broadband (BB) user with high priority and reliability requirements that coexists with random access machine-type-communications (MTC) devices. The drone serves both connectivity types with a combination of orthogonal slicing of the wireless resources and dynamic horizontal opportunistic positioning (D-HOP). We treat the D-HOP as a computational geometry function over stochastic BB user locations which requires careful adjustment in the deployment parameters to ensure MTC service at all times. Using an information theoretic approach, we optimize DSC deployment properties and radio resource allocation for the purpose of maximizing the average rate of BB users. While respecting the strict dual service requirements we analyze how system performance is affected by stochastic user positioning and density, topology, and reliability constraints combinations. The numerical results show that this approach outperforms static DSCs that fit the same coverage constraints, with outstanding performance in the urban setting.

1 Introduction

Due to their adeptness, drones or unmanned aerial vehicles (UAVs), have a growing importance in the world of communications as potential aid and substitution to classic cellular infrastructure [1, 2]. Aside extraordinary and unconventional applications, low or high altitude platforms (LAPs or HAPs) and drones as small cells (DSCs) are proven to provide good coverage through avoiding strong signal shadowing due to their high altitude to ground based user equipment [3–7]. Hence, DSC proliferation in the next generation of wireless service has the leverage to lower capital and operating expenditures (CAPEX & OPEX) in developed countries by up to 52% and 42% respectively, through only diminishing site management costs and complexity [1]. The coming generation of 3GPP and IEEE 802.11 communications calls for the coexistence of diverse and heterogeneous services within the same area that are served through the same interface in a concept known as radio access network (RAN) slicing. 5G defines three canonical service types with drastically different requirements, massive machine-type-communications (mMTC), enhanced mobile broadband (eMBB), and ultra reliable low latency communications (URLLC) [9] [10], and for the upcoming IEEE 802.11be protocol, ultra high speed service links and reliable and low-latency communications [11]. In this setting, drones can modify their spatial position in favor of improving channel conditions for priority users when serving two separate service categories. As illustrated in Fig. B.1 a dynamic standalone drone small cell (DSDSC) [3] exploits a dynamic horizontal opportunistic positioning (D-HOP) technique in strict priority for BB users, while successfully maintaining a lower tier reliability service MTC over the whole cel-

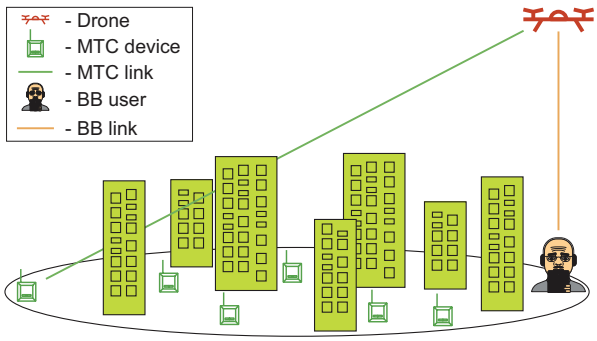


Fig. B.1: Illustration of the drone provided MTC and BB links.

lular area. In this setting, the drone's horizontal location can be treated as stochastic because the DSC relocates in favor of the BB users with stochastic positions and can assume extreme positions as the one illustrated.

1.1 Relation to State of The Art

The problem of adequate DSC positioning is recognized as cardinal in understanding deployment feasibility where the relative proportions of the DSC hold a key role in the performance of the system [1–8]. Additionally, exploiting the dynamic nature of DSCs to favor user locations is bound to improve system performance as per the works of [3, 12]. The work in [13] extends the optimal drone positioning problem to serving multiple Quality of Service (QoS) categories. Regarding the provisioned traffic categories of 5G, the work of [14] investigates the possibility of establishing the challenging traffic type of URLLC, to and from drones in different propagation environments. However, a common proposal towards slicing multiple services with DSCs is by employing tiered networks where drones are part of a complex architecture [15]. In accord, the work of [16] considers the UAV as modular equipment for more granular assistance to the needs of users with heterogeneous coverage demands. In a similar fashion, the works of [17, 18] are concerned with keeping the slicing functionality of the cellular network with mixed special purpose UAVs. Finally, the emerging work of [19] is concerned with maximizing the data rate for eMBB users while optimizing UAVs' total transmit power for drone provided slices.

This work is concerned with providing reliable wireless coverage for areas that demand slicing of two traffic groups, that is enhanced by performing preferential movements to the prioritized group. This is a novel setting that has received a lack of attention especially for a drone that is alone in serving both broadband and random access traffic. Once deployed, the DSDSC provided interface is sliced between both

MTC devices and BB users. Since appropriate deployment parameters assure optimal operation, our goal is maximizing the average rate for priority BB users, by jointly optimizing the slicing allocation per priority class and the DSC interdependent deployment properties of: DSC height, cell size, and antenna beamwidth.

Mainly motivated by the information theoretic study of the case of slicing eMBB and mMTC in [9], this work conducts Monte Carlo simulations to express the complex probability mixture into the potential gains of such an implementation. The proposed model is the first one that to our knowledge considers stochastic horizontal DSDSC positions and is described in detail in Section 2. Followed by a detailed elaboration in maintaining dual coverage in the case of stochastic mobility and problem description in Section 3. The simulation analysis and the adequate discussions are contained in Section 4. The final conclusions are drawn in Section 5.

2 System Model

To evaluate how DSC deployment impacts user service quality we assume an over-provisioned and stable back-haul. The cell is defined by a fixed circular area with radius $D_{\max}$, that contains K BB priority devices, where each $i \in \{1, 2, .., K\}$ is distributed by a Poisson Point Process (PPP) with intensity $\Lambda = A\lambda$ users/km^2. This makes K a Poisson distributed random variable (RV) with density Λ that has a squared relationship with $D_{\max}$ as $A = \pi D_{\max}^2$; and results in uniform BB user location distribution [20]. As shown with red in Fig B.2, the drone position is defined by height H with horizontal coordinates $(x_{\mathrm{d}}, y_{\mathrm{d}})$, that can be off-center due to the operation of the D-HOP technique. The flying height of the drone H is predefined during deployment and remains unchanged during D-HOP service since height modifications are incompatible with directional antenna use [1–3].

Since our goal is to improve the BB service by best adjusting the wireless resources and drone deployment, we approach D-HOP from the perspective of solving a computational geometry problem over the random BB user locations. In this setting, the DSC has full knowledge of each priority device's ground location and activating D-HOP gives the DSC the ability to locate itself in the very center of the smallest bounding circle (SBC) that contains all the active BB users. The SBC is the most fairness oriented D-HOP approach [3] as it manages to achieve the shortest maximum horizontal distance from the drone to any node D_{SBC}. This renders BB user clusters to have no impact over the drone location. Since the number of BB users and their locations are stochastic, D_{SBC} is a RV as well.

In the rest of the paper, we express all horizontal distances between the drone and any point with horizontal coordinates (x, y) as normalized by the cell radius $\kappa = \frac{\sqrt{(x-x_{\mathrm{d}})^2+(y-y_{\mathrm{d}})^2}}{D_{\max}}$, where $0 \leq \kappa \leq 2$. In the same way instead of D_{SBC} we take its normalized version $\kappa_{\mathrm{SBC}} = \frac{D_{\mathrm{SBC}}}{D_{\max}}$ when calculating BB service reliability.

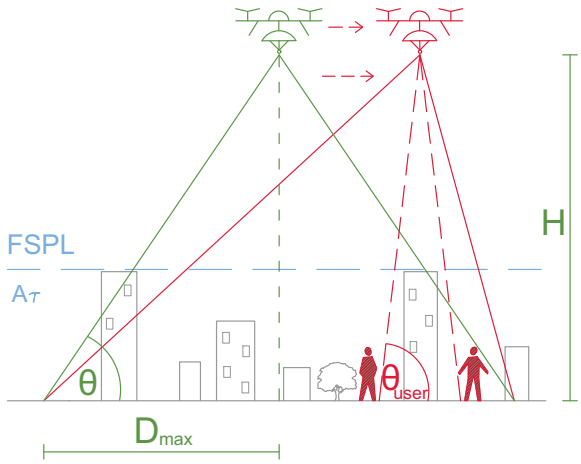


Fig. B.2: Vertical mapping of the propagation environment for reference drone position (green), and a snippet of DSDSC implementation with antenna tilting (red) [3].

2.1 Propagation Model

With green in Fig. B.2 we illustrate the central position of the DSDSC that serves as a good reference through defining the reference elevation angle at the cell edge θ. Shown with red in Fig. B.2 we illustrate how a DSDSC may improve the propagation environment by avoiding large scale shadowing imposed by buildings that obstruct users' line of sight. To account for the positive impact of drone movements, we consider channel propagation impacted by large scale fading in which a user can belong to one of two propagation groups, users with line of sight (LoS) and no LoS (NLoS). In this way, the propagation losses come as a consequence of the free space path loss (FSPL) and excessive path loss [4, 5]. To investigate the channel degradation for some user we need to know that user's elevation angle $\theta \leq \theta_{\text{user}} \leq \frac{\pi}{2}$ and formulate the path loss as:

$$10\log(L) = -G_t(\theta) + 20\log(\theta D_{\max}) - 20\log(\sin(\theta_{\text{user}})) + 20\log(\frac{f4\pi}{c}) + A_\tau(\theta_{\text{user}}), \quad \text{(B.1)}$$

where: log is a shortened version of the common logarithm $\log_{10}$, $G_t(\theta)$ is the gain of the drone's antenna, the terms $20\log(\theta D_{\max}) - 20\log(\sin(\theta_{\text{user}}))$ give the total direct air-to-ground distance between the drone and the user, the $\frac{f4\pi}{c}$ term is the FSPL term dependent on the operating frequency f, and finally, A_τ is a RV giving the excessive path loss value of large scale fading.

Large Scale Fading

defined by A_τ is different for users belonging to either LoS and NLoS propagation group and we represent as:

$$A_\tau(\theta_{\text{user}}) = \tau A_{\tau=1}(\theta_{\text{user}}) + (1-\tau)A_{\tau=0}(\theta_{\text{user}}), \tag{B.2}$$

where $\tau = \{0, 1\}$ is an indicator that can have value 1 with probability $\text{P(LoS)} = \text{P}(\tau = 1)$ or 0 with probability $\text{P(NLoS)} = 1 - \text{P}(\tau = 1) = \text{P}(\tau = 0)$. The P(LoS) value is dependent on the elevation angle of the user θ_{user} given in radians, and topography dependent constants a and b [5]:

$$\text{P(LoS)} = \frac{1}{1 + a\exp(-b(\theta_{\text{user}}\frac{180}{\pi} - a))}. \tag{B.3}$$

The A_τ members are also probabilistic values that are Gaussian-Distributed with mean μ_τ and variance dependent on the elevation angle of the user, and topology constants d_τ and c_τ for each propagation group τ as in [5]:

$$A_\tau(\theta_{\text{user}}) \sim N(\mu_\tau, \sigma_\tau(\theta_{\text{user}})), \tag{B.4}$$

where the variance is given as:

$$\sigma_\tau(\theta_{\text{user}}) = d_\tau \exp(-c_\tau\theta_{\text{user}}\frac{180}{\pi}). \tag{B.5}$$

The antenna Gain

$G_t(\theta)$ for the directional antenna mounted on the drone is taken as uniform gain over the whole aperture. The gain depends on the reference angle θ, but is scaled by its effectiveness $0 < E_r < 1$ in matching an ideal antenna covering the same region, as given in [3]:

$$G_t(\theta) = E_r 10\log(\frac{2}{1-\sin(\theta)}). \tag{B.6}$$

The drone tilts the antenna to cover the whole cellular area while moving, for which, the distortions of the circular shape to an oval one do not influence the analysis.

Path Loss Equivalent

$L_{\text{e}}(\theta, \kappa)$ is a reformulation of Eq. (B.1) that represents the path loss experienced per meter of the cell's radius. In order to adapt this further to the case with stochastic users, we isolate the user positioning gain term and label it as $G_{\text{U}}(\theta_{\text{user}}) = A_\tau(\theta_{\text{user}}) - 20\log(\sin(\theta_{\text{user}}))$, where $\theta_{\text{user}} = \arctan(\frac{\tan(\theta)}{\kappa})$. This leads to the final reformulation to get the path loss equivalence term:

$$L_{\text{e}}(\theta, \kappa) = 10\log(L) - 10\log(C) = -G_t(\theta) + 20\log(\tan(\theta)) + G_{\text{U}}(\theta, \kappa), \tag{B.7}$$

where: $C = \frac{f^2 4^2 \pi^2 {D_{\max}}^2}{c^2}$.

The path loss Equivalent metric is essential in establishing the coverage constraints and performance comparisons to classic DSC implementations that calculate the minimum path loss equivalent (MPLE). In detail, solving the MPLE problem gives the optimal θ for some fixed and specific normalized distance of κ; done as $\min_\theta L_\mathrm{e}$. This approach ties all three deployment parameters H, $D_{\max}$, and antenna beamwidth to be directly represented by the reference elevation angle θ and the radius of the cell $D_{\max}$, where the latter has no impact on the MPLE [3–5]. In contrast to this, $D_{\max}$ does have an impact in choosing the optimal deployment parameters for the case of combined D-HOP and slicing which is addressed at the end of this Section.

2.2 Service Model

We analyze the service requirements by considering that the rate provided to each user is given in bits per symbol. The number of users and scalability for each service can be addressed by bandwidth and power allocation methods that increase the number of symbols used for transmission on the channel, and are not relevant to the analysis. Therefore, the single interface is shared by a BB user and MTC user [9] in an orthogonal multiple access (OMA) manner. Let ω_B where $0 \leq \omega_\mathrm{B} \leq 1$ be the fraction of orthogonal time sharing for the BB user. Then the fraction left for the MTC user is $\omega_m = 1 - \omega_B$. Due to orthogonality, there is no interference between the services, such that the achievable rate R_x for service BB and MTC where $x \in \{B, m\}$ respectively, is:

$$R_x = \omega_x \log_2(1 + \frac{P_\mathrm{t}}{NC10^{L_{\mathrm{e,x}}/10}}), \tag{B.8}$$

where P_t is the transmit power, N is the noise power, and both C and $L_{e,x}$ represent signal attenuation as given in Eq. (B.7).

Service Reliability

for service x can be calculated from probability of the rate of our system to be smaller than a required rate R_x^{req}. To find the $1 - \epsilon_x$ reliability [9, 14], we first look at $P(R_x > R_x^{\mathrm{req}})$. Using intermediary value for targeted user positioning gain G_x as:

$$G_x = -10\log((2^{\frac{R_x^{\mathrm{req}}}{\omega_x}} - 1)\frac{NC}{P_t}) + G_t(\theta) - 20\log(\tan(\theta)), \tag{B.9}$$

we seek if it satisfies $G_\mathrm{U}(\theta, \kappa) < G_x$ for the distribution of $A_\tau(\theta_{\mathrm{user}})$ with system performance of $R_x > R_x^{\mathrm{req}}$. We then calculate the Cumulative Density Function as:

$$\mathrm{P}(G_\mathrm{U}(\theta, \kappa) < G_x) = \tag{B.10}$$

$$= \mathrm{P}(\tau = 0)\mathrm{P}(A_{\tau=0} < u) + \mathrm{P}(\tau = 1)\mathrm{P}(A_{\tau=1} < u), \tag{B.11}$$

where $u = G_x + 20\log(\sin(\arctan(\frac{\tan(\theta)}{\kappa})))$. The sum then unwraps into:

$$\begin{aligned} &\mathrm{P}(\tau=0)\int_{-\infty}^{u}\frac{1}{\sqrt{2\pi\sigma_0^2}}e^{-\frac{(x-\mu_0)^2}{2\sigma_0^2}}dx \\ &+\mathrm{P}(\tau=1)\int_{-\infty}^{u}\frac{1}{\sqrt{2\pi\sigma_1^2}}e^{-\frac{(x-\mu_1)^2}{2\sigma_1^2}}dx = \\ &\frac{1-\mathrm{P}(\tau=1)}{2}\left(1+\mathrm{erf}(\frac{u-\mu_0}{d_0\exp(-c_0\arctan(\frac{\tan(\theta)}{\kappa}\frac{180}{\pi}))\sqrt{2}})\right) \\ &+\frac{\mathrm{P}(\tau=1)}{2}\left(1+\mathrm{erf}(\frac{u-\mu_1}{d_1\exp(-c_1\arctan(\frac{\tan(\theta)}{\kappa}\frac{180}{\pi}))\sqrt{2}})\right), \end{aligned} \tag{B.12}$$

where erf is the error function and the probability $\mathrm{P}(\tau=1) = \mathrm{P}(\mathrm{LoS})$ is taken as defined in Eq. (B.3). An explicit expression for $\mathrm{P}(G_{\mathrm{U}}(\theta,\kappa) < G_x) = 1-\epsilon_x$ in terms of G_x is not possible and requires numerical solutions especially when calculating probability mixtures with κ as a RV.

Reference SNR of Static LAP

is needed to define a baseline testing scenario by mitigating the dependence on $D_{\max}$. For this, we go back to the reference scenario where the drone is static in the center of the cell and solve MPLE for the cell edge $\kappa = 1$, which is trivial and identical to the classic LAP [4] case. The result is a pathloss equivalent named $L_{\text{e-LAP}}$ defined by placement at an optimal LAP cell edge elevation angle θ_{LAP}. Since BB service is demanding $P(R_{\mathrm{B}} \geq R_{\mathrm{B}}^{\mathrm{req}}) = 1-\epsilon_B$, we must be able to satisfy the path loss equivalent $L_{\text{e-LAP}}$ when we transmit with $P_{\max}$ given by signal-to-noise ratio:

$$\frac{P_{\max}}{N} = (2^{\frac{R_{\mathrm{B}}^{\mathrm{req}}}{\omega_{\mathrm{B}}}} - 1)10^{L_{\text{e-LAP}}/10}C. \tag{B.13}$$

This stands as the most basic application of a DSC [4], and it is expected that D-HOP enabled systems should outperform it when using the same power system driven by the identical SNR requirements of $\frac{P_{\max}}{N}$. Finally, this action allows us to isolate the issue of the size of the cell with radius $D_{\max}$ as follows in Section 3.

3 Problem analysis and Coverage Constraints

To avoid misleading the reader that the drone offers perfect service for the priority users when no service is needed at all, we only consider the cases where $K > 0$. An example of the κ_{SBC} distribution in seven different cell sizes for $\lambda = 2$ is shown on Fig B.3, from which we can conclude that increasing the cell radius affects the SBC algorithm to have less certainty in providing low κ distances.

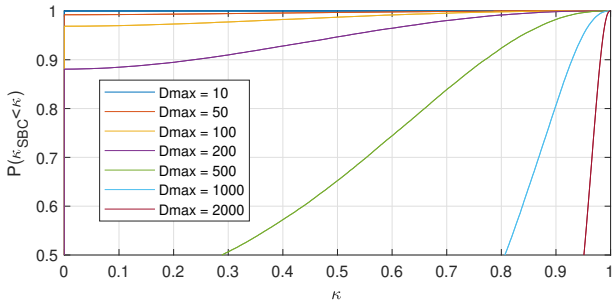


Fig. B.3: Cumulative Probability Distribution Function of κ_{SBC} for $\lambda = 2$ users/km^2 in seven different cell sizes given in meters [m].

3.1 BB Coverage

In an active D-HOP implementation, the worst case distance for two or more BB users is given by κ_{SBC} which is crucial in defining BB coverage. The rate at the edge of the SBC is given by $R_{\mathrm{SBC}}(\theta, \kappa_{\mathrm{SBC}})$, to which we define the BB coverage constraint as:

$$\mathrm{P}(\omega_B \log_2(1 + \frac{P_{\max}}{NC10^{L_{\mathrm{e,B}}(\theta,\kappa_{\mathrm{SBC}})/10}}) \geq R_{\mathrm{B}}^{\mathrm{req}}) \geq 1 - \epsilon_B, \tag{B.14}$$

where the power to noise ratio can be calculated from (B.13) and gives the constraint simplified as:

$$\mathrm{P}(L_{\mathrm{e,B}} \leq L_{\text{e-LAP}}) \geq 1 - \epsilon_B. \tag{B.15}$$

The complexity of calculating this constraint comes from the mixture distribution $G_{\mathrm{U}}(\theta, \kappa_{\mathrm{SBC}})$ in $L_{\mathrm{e,B}}(\theta, \kappa_{\mathrm{SBC}})$ that is given when passing κ_{SBC} trough Eq. (B.10); an operation that must be done numerically.

3.2 MTC Coverage

The second tier MTC service should be offered to the whole cell, at all times, no matter where the drone is positioned. This is done since MTC access times are random and it is unrealistic to expect that we know the devices' positions with regards to the drone's position. We therefore account for the worst case MTC horizontal distance of $\kappa = 2$ when the drone is positioned at the edge of the cell. For a defined reference angle θ, to guarantee the MTC service we first need to calculate the rate as in Eq. (B.8). Since we set a power limit as in the reference LAP case we reach:

$$\mathrm{P}((1 - \omega_B) \log_2(1 + \frac{P_{\max}}{NC10^{L_{\mathrm{e,m}}(\theta,\kappa=2)/10}}) \geq R_{\mathrm{m}}^{\mathrm{req}}) \geq 1 - \epsilon_m. \tag{B.16}$$

Substituting the maximum power requirement for the reference LAP case from Eq. (B.13) we get the CDF for MTC coverage:

$$(2^{\frac{R_{\mathrm{m}}^{\mathrm{req}}}{1-\omega_{\mathrm{B}}}}-1)10^{L_{\mathrm{e}}(\theta,\kappa=2)/10}C \leq (2^{\frac{R_{\mathrm{B}}^{\mathrm{req}}}{\omega_{\mathrm{B}}}}-1)10^{L_{\text{e-LAP}}/10}C, \tag{B.17}$$

which we can input in the final coverage constraint of:

$$\mathrm{P}(L_{\text{e-LAP}} - 10\log(\frac{(2^{\frac{R_{\mathrm{m}}^{\mathrm{req}}}{1-\omega_{\mathrm{B}}}}-1)}{(2^{\frac{R_{\mathrm{B}}^{\mathrm{req}}}{\omega_{\mathrm{B}}}}-1)}) \geq L_{\mathrm{e}}(\theta,\kappa=2)) \geq 1-\epsilon_m. \tag{B.18}$$

It should be noted that in this way, we are massively overprovisioning the MTC link when compared to the classic LAP case. This applies as we support MTC in the absolute worst case scenario of κ which is necessary for guaranteeing random access.

3.3 Maximal RAN slicing ratio

If in the case of Eq. (B.17) we calculate the MPLE value of $L_{\mathrm{e}}(\theta,\kappa=2)$ we get some minimal value that can help us place a constraint for achieving maximum ω_{B}. In other words, if we choose maximum possible value for ω_{B}, there is only one θ at which the drone can operate and maintain MTC service for the whole cell. Therefore, using non-maximal ω_{B} should result in bigger range of possible θ values, at which MTC service is maintained. Finding the optimal θ, can be done numerically through finding MPLE, and reach to a value of minimal path loss for $\kappa=2$ named, $L_{\text{e-RSR}}(\theta_{\mathrm{RSR}},\kappa=2)$. And in order for MTC coverage to be satisfied, the following equation must hold:

$$10^{\frac{L_{\text{e-LAP}}-L_{\text{e-RSR}}}{10}} \geq \frac{(2^{\frac{R_{\mathrm{m}}^{\mathrm{req}}}{1-\omega_{\mathrm{B}}}}-1)}{(2^{\frac{R_{\mathrm{B}}^{\mathrm{req}}}{\omega_{\mathrm{B}}}}-1)}. \tag{B.19}$$

This ratio is an upper bound constraint for the value ω_{B}, with which we can test if the scenario is within the theoretical limits of our system.

3.4 Problem Description

In a specific snapshot of the cell, the active BB user $i \in 1,2,3\ldots K$, has a normalized horizontal distance κ_i^{D} to the D-HOP activated drone, and a normalized horizontal distance of κ_i^{S} to the scenario with a static LAP at the center of the cell. When served by the D-HOP enabled drone, this BB user will experience a rate of:

$$R_i^{\mathrm{D}}(\theta,\omega_{\mathrm{B}}) = \omega_{\mathrm{B}}\log_2(1+(2^{\frac{R_{\mathrm{B}}^{\mathrm{req}}}{\omega_{\mathrm{B}}}}-1)10^{L_{\text{e-LAP}}/10-L_{\mathrm{e}}(\theta,\kappa_i^{\mathrm{D}})/10}), \tag{B.20}$$

Table B.1: The excessive path loss parameters for: Suburban (1), Urban (2), and High Urban (3) environments [5].

	a	b	μ_1	μ_0	d_1	d_0	c_1	c_0
1	4.88	0.43	0.1	21.0	11.25	32.17	0.06	0.03
2	9.61	0.16	1.0	20.0	10.39	29.6	0.05	0.03
3	12.08	0.11	1.6	23.0	8.96	35.97	0.04	0.04

while in the reference LAP system located in the very center of the cell, the same BB user i will experience a rate of:

$$R_i^{\mathrm{S}}(\omega_{\mathrm{B}}) = \omega_{\mathrm{B}} \log_2(1 + (2^{\frac{R_{\mathrm{B}}^{\mathrm{req}}}{\omega_{\mathrm{B}}}} - 1)10^{L_{\text{e-LAP}}/10 - L_{\mathrm{e}}(\theta_{\mathrm{LAP}}, \kappa_i^{\mathrm{S}})/10}). \tag{B.21}$$

Since our goal is to maximize the average rate for all active BB users in the cell we introduce the problem as:

$$\underset{\{\theta, \omega_{\mathrm{B}}\}}{\text{maximize}} \quad \frac{E[R_i^{\mathrm{D}}(\theta, \omega_{\mathrm{B}})]}{E[R_i^{\mathrm{S}}(\omega_{\mathrm{B}})]} \tag{B.22}$$

$$\text{s.t.} \quad 0 < \theta < \frac{\pi}{2} \tag{B.23}$$

$$0 < \omega_{\mathrm{B}} \tag{B.24}$$

$$(\text{B.15}), (\text{B.18}), (\text{B.19}) \tag{B.25}$$

The objective function (B.22) is normalized by (B.21) because an analysis in the non-normalized case would not give productive results as the transmission power given by (B.13) scales with the reference deployment. Furthermore, the real altitude H will directly depend on the deployment radius and not only θ. This would then impact $D_{\max}$, and in order to cover bigger areas, a multi drone provided coverage may be necessary such as in [7, 8]. Therefore, we are only interested in finding an optimal reference elevation angle θ and ω_{B} combination while we try to maximize the average rate improvements to the active BB users. Since the involved functions are monotonic within the given constraints, to efficiently find the maximum we applied a binary-search algorithm to the Monte Carlo simulated estimates of κ_{SBC} used to realize the G_{U} mixture.

4 Simulation and Results

We take a snapshot approach where at each snip the drone is at the SBC center of the BB users while maintaining the full MTC coverage as per the defined constraints. Each snip has no correlation to a previous one and no mobility of the users or trajectories are concerned. The service requirements for both priority groups are tested as per the

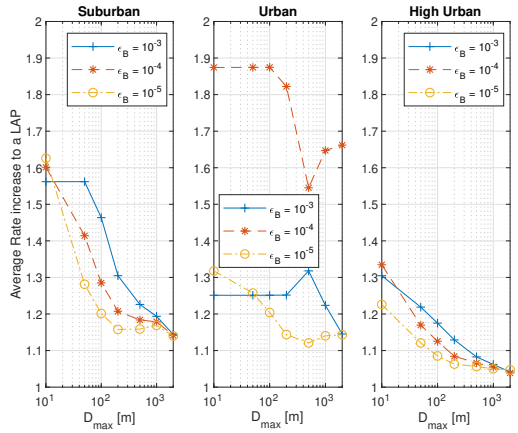


Fig. B.4: ARI gain results for $E_{\mathrm{r}} = 0.6$.

eMBB and mMTC slicing requirements taken from [9]. The mMTC devices require a rate of $R_{\mathrm{m}}^{\mathrm{req}} = 0.3$ with low reliability of $\epsilon_m = 1 - 0.9$, and the rate for serving an eMBB user as $R_{\mathrm{B}}^{\mathrm{req}} = 1$. In addition to medium $\epsilon_{\mathrm{B}} = 1 - 0.999$ (as in [9]), we test under high $\epsilon_{\mathrm{B}} = 1 - 0.9999$ and ultra $\epsilon_{\mathrm{B}} = 1 - 0.99999$ eMBB reliability requirements. We investigate this by testing the previously shown seven different cell sizes of $D_{\max} = [10\mathrm{m}, 50\mathrm{m}, 100\mathrm{m}, 200\mathrm{m}, 500\mathrm{m}, 1000\mathrm{m}, 2000\mathrm{m}]$ in an area with fixed user density of $\lambda = 2$ users/km^2. Additionally, we test three different deployment environments for a 2 GHz carrier, that affects A_τ as shown in Table B.1. The topological constants required for calculating Eq. (B.10) are taken from the Suburban, Urban and High Urban parameters of [5] from where High Urban is renamed from Dense Urban in [5] to avoid confusions regarding user density.

In Fig. B.4 we notice that the deployment feasibility, measured by the average rate increase (ARI), depends strongly on the local topology, where, Suburban and High Urban environments show that there is a strong decrease in D-HOP ARI in user-dense environments with an exception to this rule for the Urban deployments. In the case of High eMBB reliability for Urban deployments, the implementation achieves astonishing performance for all types of user densities, amounting to average rate improvements of 50-90%. This performance is credited to the nature of the fading distribution, and the reliability combination for both eMBB and mMTC users occur very close to a point where the P(LoS) equation (B.3) shows a strong impact in the distribution of Eq. (B.10). Moreover, the DSDSC allows itself to serve mostly LoS links due to its mobility, while the static LAP has to meet the high reliability requirement of mostly, NLoS traffic. The average gains are subject to the much stricter reliability requirements of Eq. (B.15), which results in optimal θ and ω_{B} combination limited by the distribution of the smallest SBC radius. Since κ_{SBC} is directly dependent on the $D_{\max}$ we see

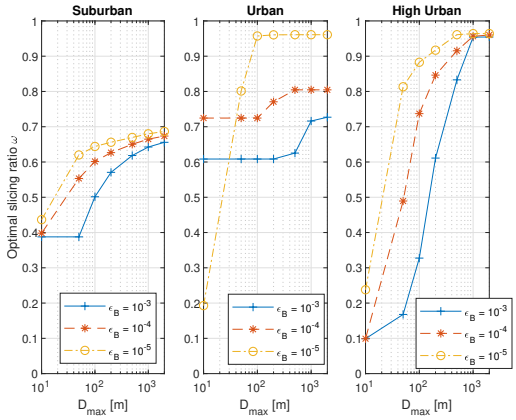


Fig. B.5: Optimal Slicing Ratio ω_{B} evolution for $E_{\mathrm{r}} = 0.6$

that ARI drops off steadily with the increase in cell size. Due to the fact that we measure performance as relative gain over a LAP equivalent setup and maintain cell proportions with θ, the absolute dimensions of the cell do not have a direct impact on the measured rate but only impacts the user density distribution. Finally, by testing the four different $E_{\mathrm{r}} = [0.1, 0.3, 0.6, 0.9]$ we only confirmed that better antenna compatibility results in overall higher optimal θ, which in turn diminishes the advantages of D-HOP implementations and results in lower ARI [3]. When necessary to avoid clutter, we only show the performance for $E_{\mathrm{r}} = 0.6$.

In our optimization problem the value ω_{B} fulfills two very important roles as it controls the rate and impacts the availability of θ for the optimization algorithm. The slicing ratios given in Fig. B.5 show strong correlation between low slicing ratio for eMBB users and high ARI performance. This is due to the relaxation of the mMTC coverage constraint for dynamic system coverage given in Eq. (B.18). By relaxing this value, the SBC distribution allows for achieving lower operating optimal θ as shown in Fig. B.6 which places it closer to the eMBB user and results in higher ARI.

We go back to analyze the most interesting case, the urban DSDSC implementation for high eMBB reliability as it shows strong performance advantages. The reason for this high outlier in the performance of this system can be deduced from Fig. B.7. Here we can notice that the optimal θ for the system does not dramatically vary with neither user density nor antenna efficiency. Comparing the optimal slicing ratios, we notice that because of the point of where we are in the distribution, the system benefits very little from lowering the slicing ratio when compared to the other deployment cases. Hence, the optimal ω_{B} in this case is preferred to be lower for higher E_{r} values, which is inverted from the other deployments. This means that the operating θ where the topology allows for the maximum slicing ratio constraint is very close to the optimal θ for the static

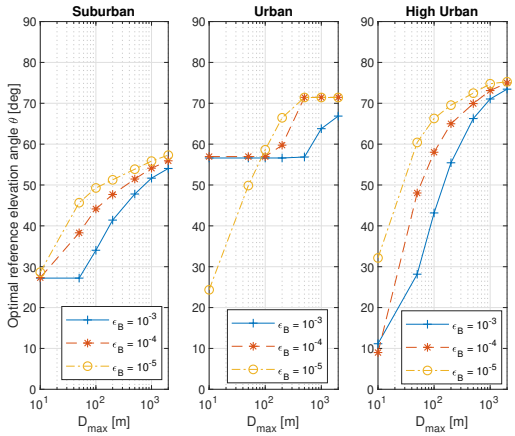


Fig. B.6: Reference elevation angle θ evolution for $E_{\mathrm{r}} = 0.6$

system. In these cases, the slicing ratio is used to compensate for the differences in the use of more efficient antennas, and use the same optimal θ. Such systems find themselves in a *just right* or Goldilocks deployments scenario for the support of both schemes. Which in our case occurs in the urban scenario for slicing support of highly reliable eMBB traffic and low mMTC reliability support.

5 Conclusion

This particular implementation of DSDSCs in slicing of MTC and BB traffic was tested in strict priority for BB users with coverage requirements for both services. Through Monte Carlo analysis we deduced that the use of this system outperforms its static counterpart in many different deployment scenarios, with expected superiority in implementations with low antenna efficiency and sparse BB user activation distributions. However, the implementation showed strong performance in the slicing DSDSC deployment in Urban environments for high reliability requirements for the BB traffic users. Opposed to the other deployment scenarios, this shows significant superiority with regards to static LAP small cells with average rate improvements of 50-90% for the case of antenna with an efficiency coefficient of $E_r = 0.6$. This analysis strongly highlights the importance of beforehand knowing the propagation properties of the environment as central in discovering the feasibility of DSDSC slicing deployments. We finally conclude that as per the given results, standalone deployment of a service slicing DSC can strongly benefit from dynamic and preferential horizontal movements. Finally, this research is limited in that it does not account for the times for which a drone would need to reach the new position, a topic reserved for a future iteration of the work.

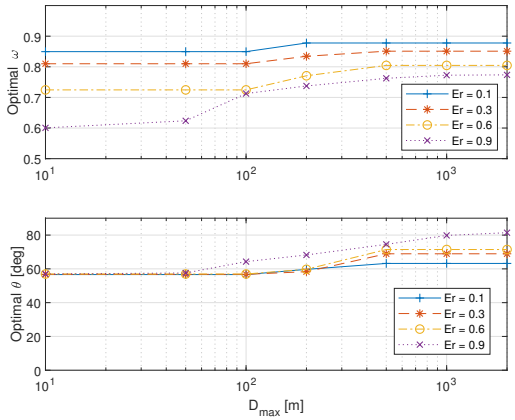


Fig. B.7: Full E_{r} analysis for $\epsilon_B = 1 - 0.9999$ in Urban deployment

Acknowledgment

The work was supported by the European Union's research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 812991 "PAINLESS" within the Horizon 2020 Program.

References

[1] A. Fotouhi et al., "Survey on UAV Cellular Communications: Practical Aspects, Standardization Advancements, Regulation, and Security Challenges," in IEEE Communications Surveys & Tutorials 2019.

[2] M. Mozaffari, W. Saad, M. Bennis, Y. Nam and M. Debbah, "A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems," in IEEE Communications Surveys & Tutorials, vol. 21, no. 3, pp. 2334-2360, thirdquarter 2019.

[3] I. Donevski, J.J. Nielsen, "Dynamic Standalone Drone-mounted Small Cells" 2020 European Conference on Networks and Communications (EuCNC) , Dubrovnik, Croatia. IEEE, 2020.

[4] A. Al-Hourani, S. Kandeepan and S. Lardner, "Optimal LAP Altitude for Maximum Coverage," in IEEE Wireless Communications Letters, vol. 3, no. 6, pp. 569-572, Dec. 2014.

[5] A. Al-Hourani, S. Kandeepan and A. Jamalipour, "Modeling air-to-ground path loss for low altitude platforms in urban environments," 2014 IEEE Global Communications Conference, Austin, TX, 2014, pp. 2898-2904.

[6] M. Alzenad, A. El-Keyi, F. Lagum and H. Yanikomeroglu, "3-D Placement of an Unmanned Aerial Vehicle Base Station (UAV-BS) for Energy-Efficient Maximal Coverage," in IEEE Wireless Communications Letters, vol. 6, no. 4, pp. 434-437, Aug. 2017.

[7] M. Mozaffari, W. Saad, M. Bennis and M. Debbah, "Efficient Deployment of Multiple Unmanned Aerial Vehicles for Optimal Wireless Coverage," in IEEE Communications Letters, vol. 20, no. 8, pp. 1647-1650, Aug. 2016.

[8] N. Babu, C. B. Papadias and P. Popovski, "Energy-Efficient 3D Deployment of Aerial Access Points in a UAV Communication System," in IEEE Communications Letters,

[9] Popovski, Petar, et al. "5G wireless network slicing for eMBB, URLLC, and mMTC: A communication-theoretic view." IEEE Access 6 (2018): 55765-55779.

[10] X. Shen et al., "AI-Assisted Network-Slicing Based Next-Generation Wireless Networks," in IEEE Open Journal of Vehicular Technology, vol. 1, pp. 45-66, 2020.

[11] López-Pérez, David, et al. "IEEE 802.11 be Extremely High Throughput: The Next Generation of Wi-Fi Technology Beyond 802.11 ax." IEEE Communications Magazine 57.9 (2019): 113-119.

[12] B. Galkin, J. Kibiłda and L. A. DaSilva, "A Stochastic Model for UAV Networks Positioned Above Demand Hotspots in Urban Environments," in IEEE Transactions on Vehicular Technology, vol. 68, no. 7, pp. 6985-6996, July 2019.

[13] Alzenad, Mohamed, Amr El-Keyi, and Halim Yanikomeroglu, "3-D placement of an unmanned aerial vehicle base station for maximum coverage of users with different QoS requirements." IEEE Wireless Communications Letters 7.1 (2017): 38-41.

[14] C. She, C. Liu, T. Q. S. Quek, C. Yang and Y. Li, "Ultra-Reliable and Low-Latency Communications in Unmanned Aerial Vehicle Communication Systems," in IEEE Transactions on Communications, vol. 67, no. 5, pp. 3768-3781, May 2019.

[15] I. Bor-Yaliniz and H. Yanikomeroglu, "The New Frontier in RAN Heterogeneity: Multi-Tier Drone-Cells," in IEEE Communications Magazine, vol. 54, no. 11, pp. 48-55, November 2016.

[16] P. Lohan and D. Mishra, "Utility-Aware Optimal Resource Allocation Protocol for UAV-Assisted Small Cells With Heterogeneous Coverage Demands," in IEEE Transactions on Wireless Communications, vol. 19, no. 2, pp. 1221-1236, Feb. 2020.

[17] S. Zhang, W. Quan, J. Li, W. Shi, P. Yang and X. Shen, "Air-Ground Integrated Vehicular Network Slicing With Content Pushing and Caching," in IEEE Journal on Selected Areas in Communications, vol. 36, no. 9, pp. 2114-2127, Sept. 2018,

[18] G. K. Xilouris, M. C. Batistatos, G. E. Athanasiadou, G. Tsoulos, H. B. Pervaiz and C. C. Zarakovitis, "UAV-Assisted 5G Network Architecture with Slicing and Virtualization," 2018 IEEE Globecom Workshops (GC Wkshps), Abu Dhabi, United Arab Emirates, 2018, pp. 1-7

[19] P. Yang, X. Xi, J. Chen, T. Q. S. Quek, X. Cao and D. O. Wu, "RAN Slicing in a UAV Network for eMBB Service Provision." arXiv preprint arXiv:1912.03600 (2019).

[20] M. M. Azari, G. Geraci, A. Garcia-Rodriguez and S. Pollin, "Cellular UAV-to-UAV Communications," 2019 IEEE 30th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Istanbul, Turkey, 2019, pp. 1-7

Paper C

An experimental analysis on drone-mounted access points for improved latency-reliability

Igor Donevski, Christian Raffelsberger, Micha Sende, Aymen Fakhreddine, Jimmy Jessen Nielsen

The paper has been published in the
Proceedings of Proceedings of the 7th Workshop on Micro Aerial Vehicle Networks, Systems, and Applications, pp. 31–36, 2021.

The layout has been revised.

Abstract

The anticipated densification of contemporary communications infrastructure expects the use of drone small cells (DSCs). Thus, we experimentally evaluate the capability of providing local and personalized coverage with a drone mounted Wi-Fi access point that uses the nearby LTE infrastructure as a backhaul in areas with mixed line of sight (LoS) and Non-LoS (NLoS) links to the local cellular infrastructure. To assess the potential of DSCs for reliable and low latency communication of outdoor users, we measure the channel quality and the total round trip latency of the system. For a drone following the ground user, the DSC-provided network extends the coverage for an extra 6.4% when compared to the classical LTE-direct link. Moreover, the DSC setup provides latencies that are consistently smaller than 50 ms for 95% of the experiment. Within the coverage of the LTE-direct connection, we observed a latency ceiling of 120 ms for 95% reliability of the LTE-direct connection. The highest latency observed for the DSC system was 1200 ms, while the LTE-direct link never exceeded 500 ms. As such, DSC setups are not only essential in NLoS situations, but consistently improve the latency of users in outdoor scenarios.

1 Introduction

Unmanned Aerial Vehicles (UAVs) – or drones – are considered a prime candidate for custom solutions to wireless communication coverage in both conventional and unconventional circumstances. Mainly as a benefit to their mobility and their altitude, they are expected to offer good wireless channel conditions towards ground users in outdoor scenarios [1, 2]. Commonly referred to as Drone Small Cells (DSCs) or as Drone provided Access Points, the flying devices that carry wireless equipment are mostly considered as a custom solution for improving regional communication quality [3]. However, it is common among DSC literature to lay strong assumptions for the availability of backhaul and/or the propagation setting of the fronthaul [1]. While this is important for modeling future drone-specific network requirements, we turn the attention towards assessing the current state of commercially available technologies that are capable of drone provided communication services.

In this work we aim to aid the cellular network by analyzing and measuring the impact on latency for a mobile user (UE), located on the ground and in the presence of a DSC that provides a Wi-Fi hotspot, instead of a pure LTE relay. The reason behind designing our DSC like this is that LTE positioning requires careful spectral planning which is not an approachable method for casual users of the DSC technology. Hence, we split the backhaul as LTE provided and the fronthaul offered through a Wi-Fi interface. This is performed by placing the DSC at an altitude that allows the utilization of a nearby Base Station (BS). The goal of this setup is to answer three key questions:

does a drone provided hotspot improve the latency of the application, what are the shortcomings of such a system, and is this technology sufficient for use in reliable and low latency communications? Answering these questions is critical for prospective use of DSCs in cases of ground remote control applications that need to be tolerant to faults of the local cellular coverage.

The paper is organized as follows: Section 2 discusses significant related experimental work with drones. Section 3 describes a route for the horizontal spatial coordinates that is bound to be covered by a mixture of strong LoS and NLoS links for the UE along the route. On this route we forgo two types of approaches, pretesting the LTE propagation conditions for a travelling node on the ground, or following the same route on a specific altitude. In Section 4 we showcase the measurements of the impact on signal and delay when using the DSC network (UE-UAV-BS) and a singular (UE-BS) LTE link. Finally, in Section 6 we draw several conclusions, state the significance of the provided data, and discuss open issues.

2 Related Work

There has been a significant interest in deriving experimental results for cellular connected drones. Mainly, the effect of interference carries a significance due to the likelihood of drones to establish a LoS with neighbouring cells, which would not be visible to ground UE. This causes extra interference, which adds extra complexity to the problem of BS handover, topics that are thoroughly covered in: [4–7]. Moreover, the work of [8] gives an elaborate overview of the common issues plaguing cellular connected drones. In terms of the performance of cellular connected drones, the work of [9] provides detailed experimental analysis of the throughput of drones connected to an LTE-A network. Justifiably, due to rollout phase of 5G infrastructure the work of [10] experimentally evaluates the throughput performance of 5G connected drones.

As for UE-DSC connectivity, there is a strong experimental support of the LoS NLoS model of [11], such as the work of [12]. The work of [13] analyzes the drone-BS connectivity of LTE for its potential use as backhaul in DSCs. However, the experimental works of [14] and [15] are most in line with our goals of testing UE connectivity through a DSC. Moreover, the work of [14] is a trivial example of UAV-to-UAV communications as a relay system. While the work of [15] is most in line with our goals for establishing a UE-DSC-BS relay, its contribution is though limited to a throughput investigation in a small scenario. In addition, all prior UE-UAV-BS connectivity work has majority of the attention focused on measuring signal strength and throughput. Given the vastly different nature of cellular and Wi-Fi connectivity, and the added system complexity, we are interested in the latency and therefore delay quality for UE served by BSs versus DSCs as clarified in the following sections.

Fig. C.1: An image taken from the DSC equipment mounted on a twinFOLD SCIENCE drone.

3 Experimental Setup

The experiments are performed in a relatively controlled setting in the 5G Playground Carinthia testbed in the outskirts of Klagenfurt, Austria. All flights and ground UE patrols were taken along the same identical horizontal path with a constant speed of 1 m/s. The LTE connectivity in the area is provided by a single BS mounted on top of a building, and was predetermined for the experimental analysis. The LTE BS supports a data rate of 150 Mbit/s for each 20 MHz channel supporting a modulation of 256 QAM per carrier. While the BS supports up to two carriers, only one carrier is used during the experimentation.

The drone, Fig. C.1, is a twinFOLD SCIENCE by the Austrian vendor TWINS. The communications equipment mounted on the drone, consists of an LTE modem and a Wi-Fi 802.11ac access point connected to a Raspberry Pi 4 companion board that bridges the two interfaces and captures all traffic that passes through. This allows for a better investigation and separation of the fronthaul, backhaul and most importantly scheduling and processing delays. The Wi-Fi access point (Unifi UAP-AC-M) has a 2x2 WiFi dual band configuration with a maximum link data rate of 867 Mbit/s. The max. transmit power is 20 dBm. The access point is connected to the main board via a 1 Gbit/s Ethernet interface. The Huawei E3372 LTE modem supports LTE Cat4 with

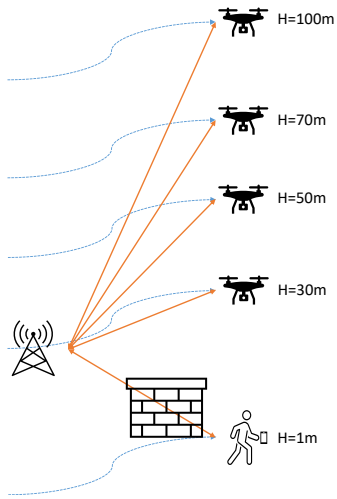


Fig. C.2: Testing LTE signal performance for four altitudes (30 m, 50 m, 70 m, 100 m) and the ground UE-BS link is likely to be NLoS.

a max. downlink rate of 150 Mbit/s and an uplink rate of 50 Mbit/s. It is connected via USB to the companion board. The smartphone is a Samsung S20 5G which supports dual-band Wi-Fi 802.11ac with VHT80 MU-MIMO and LTE Cat20 with 4x4 MIMO.

The measurements are performed with the Cellular Drone Measurement Tool (CDMT) [16] that can record multiple parameters for the cellular connection such as: reference signal received power (RSRP), reference signal received quality (RSRQ), received signal strength indication (RSSI), serving physical cell identity (PCI), and channel quality indicator (CQI) and throughput as averages of one second. To perform latency measurements, the CDMT app sends 10 UDP datagrams per second, containing a sequence number and two timestamp fields (20 byte payload), to a server located in the testbed network. The app logs the timestamp when the packet is sent and when it receives the reply from the server. The timestamp fields are used to store the time when a packet is received at the server and when it is sent back to the client in order to determine the processing time at the server.

3.1 Scenario Investigation

The performance of the DSC is significantly influenced by the UAV-to-BS connectivity. It has been shown that the current cellular BS antenna design is optimized to service ground users and drones experience degraded performance at certain altitudes, since they are only covered by side lobes [9, 10]. To better understand the propagation pattern of the base station, we perform several pre-measurements in order to improve

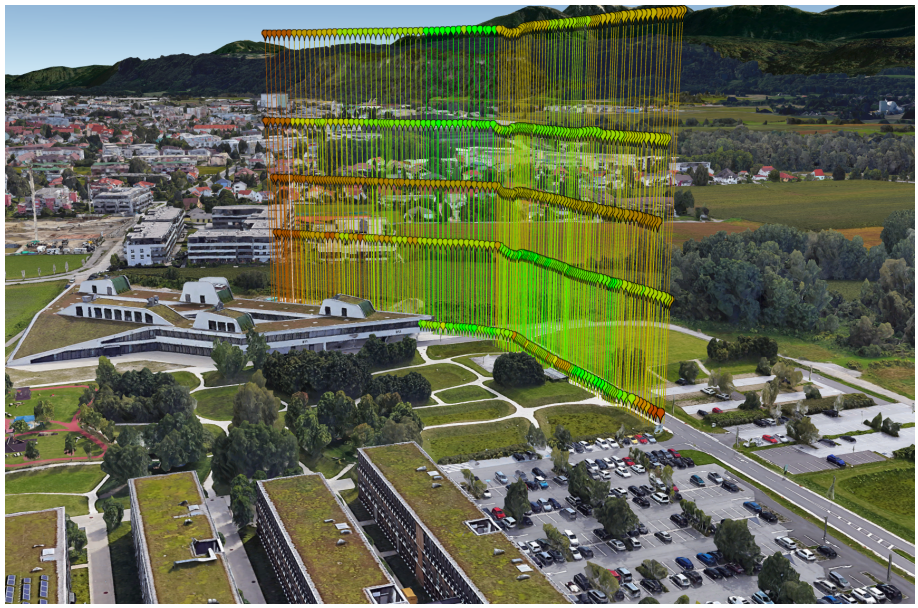


Fig. C.3: RSSI measurement results for the LTE signal performance sweep from the perspective of the BS. Colormap ranges from red for the worst RSSI measured(-107 dB) to green for the best RSSI (-75 dB) measured. (Map data: Google ©2021)

the deployment of the DSC (i.e., determine a suitable altitude for the drone). The pre-measurements start with measuring the performance on the ground, while a user carrying a smartphone walks along the route. Afterwards, we mount the smartphone on a UAV and repeat the same route at a different altitudes. Moreover, even when the drone can move horizontally its performance depends on how big of an area it wants to cover [17]. Hence as per a previous analysis for suburban environments, we derive 30 m as a decent altitude for a drone with an isotropic fronthaul (in our case Wi-Fi) transmitter, and 50 m or 70 m with a directional antenna [18]. Furthermore, we also want to sample the possibility of having drone-to-BS connection with a drone at an altitude of 100 m for UAVs that have an efficient directional antenna. As such we perform a sweep to test the LTE channel performance at different altitudes, as shown in Fig C.2.

In Fig. C.3 we can observe that the altitudes of 30 m and 70 m give the best channel conditions. Since we conduct our experiments with omnidirectional Wi-Fi antennas, we select an altitude of 30 m for the measurements of our DSC. Moreover, lower altitudes suffer less from inter-cell interference and frequent handover challenges [5]. Looking at the RSSI measurements from the opposite side, as shown in Fig. C.4, we can notice that the LTE connectivity of the ground UE severely deteriorates in the NLoS region.

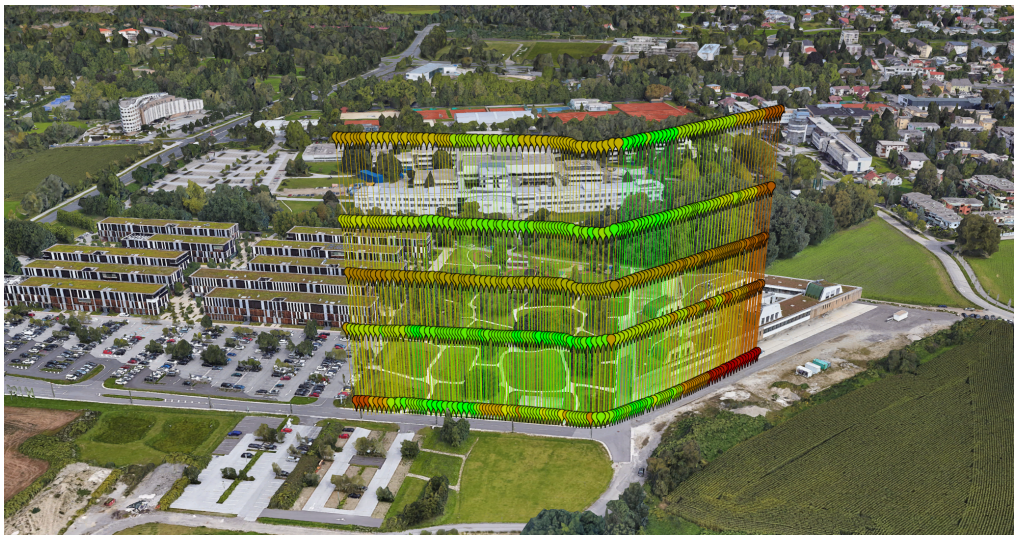


Fig. C.4: RSSI measurement results for the LTE signal performance sweep from the perspective of the DSC. Colormap ranges from red for the worst RSSI measured(-107 dB) to green for the best RSSI (-75 dB) measured. (Map data: Google ©2021)

Table C.1: Mean and variance of RSSI for all measured altitudes

Label	1 m	30 m	50 m	70 m	100 m
mean μ	-89.79	-84.46	-85.98	-84.06	-86.83
variance σ	94.86	31.68	34.73	26.39	30.90

Thus, we expect that DSC-based connectivity should eliminate this issue and improve network performance for users hidden from the direct LoS of the BS.

A final remark of the pre-measurements can be derived from Fig. C.5 where we show the moving evolution of variance and mean RSSI values. As it can be noticed, all flying drone implementations record lower RSSI fluctuations during the route. The signal strength at the moving ground UE was constantly changing, providing additional motivation for stabilizing the performance of the ground UE through a DSC system. The variance and the mean of the RSSI parameter for each pass of the sweep are contained in Table C.1.

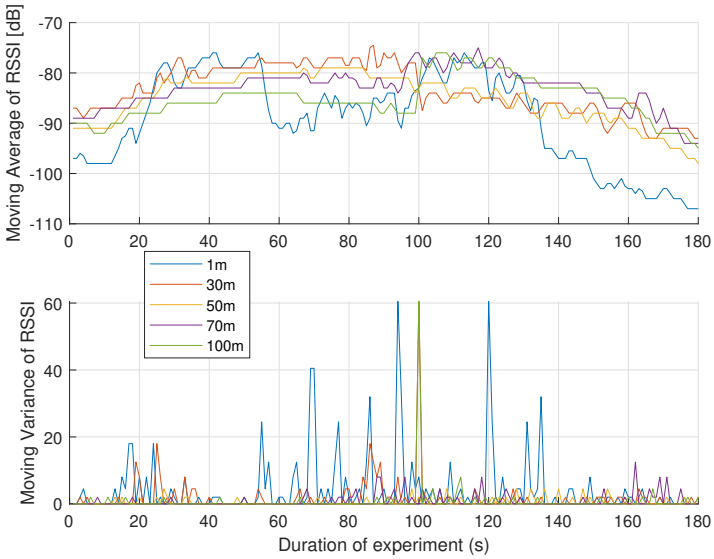


Fig. C.5: 2-point Moving Average and Moving Variance for RSSI values of all 5 experiments.

4 Experimental Results

Since we have established the DSC altitude of 30 m, we proceed with the latency measurements for the system at that particular altitude. As shown on the right in Fig. C.6, one part of the measurement is for the LTE-direct UE-BS link, measured along the route. On the left, in Fig. C.6, we illustrate the second part of the measurements where the Wi-Fi equipped drone provides DSC service to the mobile user, while flying directly above him or her along the same route. To verify the consistency of the measuring setup, the DSC measurements were repeated three times while the LTE-direct measurements were repeated two times. From that we observed consistent behavior across the measurement experiments. For better visibility, we illustrate the duration-time plot in Fig. C.7 only for one sample of each type.

Since all experiments started from the same initial spot and travel with the same speed, in Fig. C.7 we plot the latency evolution of the system with regards to the duration of the experiment. The most notable remark is that LTE-direct measurements lack 10 s of data near the end of the experiment. This was due to the strong NLoS, shown with bright red in Fig. C.6, due to which the ground-UE consistently lost connectivity to the BS near the end of the experiment. This reduces the number of samples for the LTE-direct measurements by 6.4% as for such portion of the time the LTE-direct link is in total outage. However, the DSC setup experiences latency spikes that are

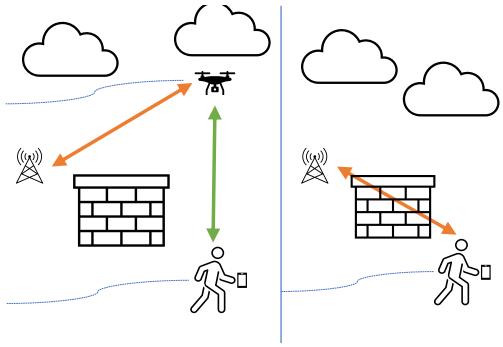


Fig. C.6: Latency measurement scenarios: the drone and UE move at 1m/s together (left), the user moves at 1m/s alone (right)

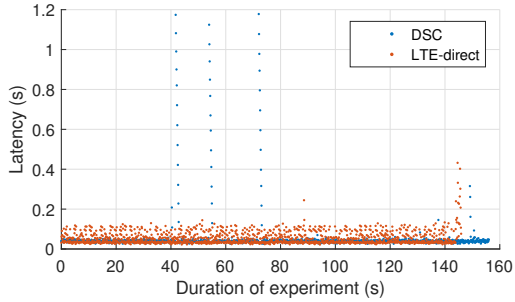


Fig. C.7: The measured latency of the DSC and LTE-direct implementations.

consistently above one second. These spikes occur in areas where the UAV has good LTE channel condition, with strong LoS. From the segmented latency data for the DSC measurement in Fig. C.8, we observe that the latency spikes are due to consistent and lengthy outages of the Wi-Fi interface. These spikes are unrelated to the position of the drone and occur at random times due to the distributed multiple access scheme implemented in Wi-Fi. Therefore, due to the inconsistency of Wi-Fi, the mean latency for both systems (within the area of BS coverage for ground users) is nearly identical, measuring to 54 ms and 53.7 ms of latency for the LTE-direct and the DSC connectivity, respectively. In addition, the Wi-Fi link adds insignificant amount of extra latency for the DSC case when it does not encounter crowding problems due to its distributed nature. Thus, in Fig. C.9 we compare the latency between the UAV-BS LTE link to the UE-BS direct. The major superiority of the DSC system comes as a consequence of the good channel conditions of the flying drone, even when both setups have generally LoS signal. Moreover, the mean LTE delay from the DSC is 33.2 ms while the mean LTE delay from the UE is 54 ms.

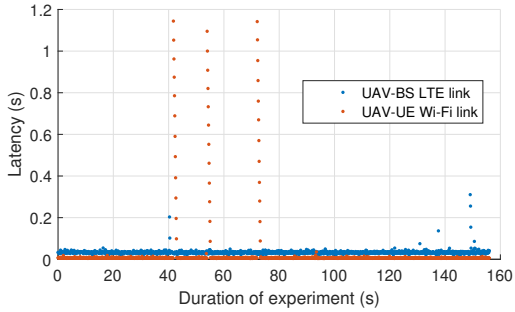


Fig. C.8: Latency comparison of the Wi-Fi link versus the LTE link for the DSC setup.

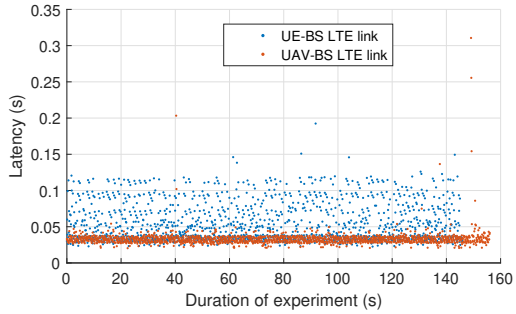


Fig. C.9: Latency comparison of the LTE-direct links from the UE and the UAV/DSC.

Even with this advantage of the UAV, the mean latency for the DSC system is nearly identical. Therefore, in Fig. C.10 we plot the empirical cumulative distribution function (ECDF) starting at the average latency, as taken from all runs. While the average latency is similar for the LTE-direct link and the DSC link, the variations are much higher for the former. In particular, it is much more likely to observe highly varying latencies between 40 ms and 120 ms when having a direct LTE connection. In other words, the LTE-direct link suffers from high jitter. This is not true for the DSC system that offers low jitter and a latency below 50 ms with 0.95 probability. Unfortunately, due to the behavior of CSMA it is likely to observe a latency in the excess of 200 ms, for 2% of the time of operation of the DSC system. We note that this investigation is limited to generally good CSMA congestion conditions, and larger delays can be introduced when sharing the Wi-Fi carrier with other devices. Therefore, through the choice of location we also impact the wireless channel congestion for both the LTE and Wi-Fi spectrum. As such, the analysis presented in this work can be further enhanced by combining the provided measurements with congestion models, which is out of the scope of this paper.

Finally, we observe the latencies in Fig. C.11 that are the result of scheduling and

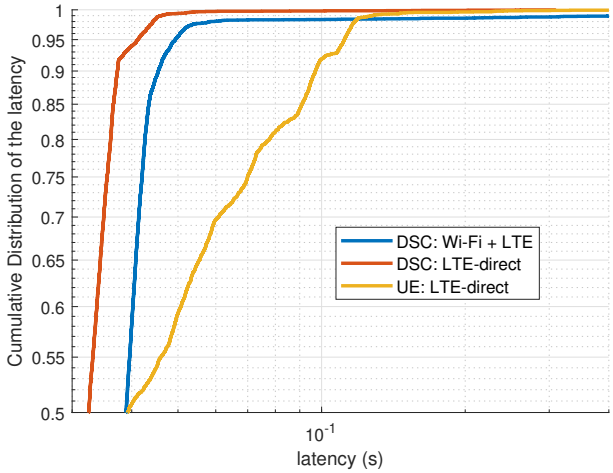


Fig. C.10: Latency-Reliability plot of above average latency for each connection.

processing of our computational system mounted on the drone.

5 Conclusions

In this paper, we conducted and elaborated an experimental analysis of the possibility of addressing latency concerns for ground based UE with DSCs. Initially, we provided a full sweep of measurements at different altitudes to inspect the impact of LoS and NLoS links. In the sweep we concluded that a DSC should operate at an altitude of 30 m to provide a Wi-Fi hotspot for the UE below. In additional experiments, we compared the performance of an LTE-direct link – between the UE and the BS – to a DSC provided hotspot. The experimental analysis shows that the altitude of the DSC allows for reliably lower latency than an LTE-direct link from the ground. Moreover, the DSC system extended the coverage for an extra 6.4% of the route. Even though this number is arbitrary and directly impacted by the choice of the flight and walking trajectories, it showcases that the DSC covers absolute edge cases of full link dropping. The drawbacks of the DSC system come due to the nature of Wi-Fi, as we occasionally observed latency spikes in the excess of 1 s that are due to full outages. This removes the possibility of using the Wi-Fi fronthaul for ultra reliable remote control implementations that require low latency. Finally, this work allows us to segment the collective fronthaul, processing and backhaul latency of a complex DSC system in mixed LoS and NLoS scenarios, limited to outdoor users. In a future work we would like to observe the performance of a similar experimental setup with a 5G backhaul. Such an analysis would be imperative since next-generation communication equipment promises low latency even in NLoS

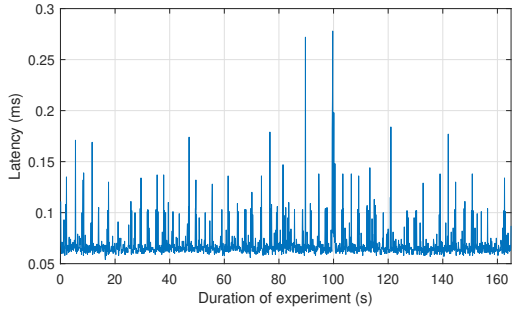


Fig. C.11: Latency, in milliseconds, induced due to total time spent processing in DSC.

environments. In those cases, upgrading the DSC fronthaul to the next-generation of 802.11be might finally achieve URLLC for outdoor ground users. Moreover, to evaluate the feasibility of such a system, we would also strive to reduce the complexity and weight of the mounted equipment in favor of increased drone air time. Such work can finally target the goal of URLLC for outdoor remote control applications in obstacle-dense environments, an evolution towards modular, fault-tolerable and reliable connectivity.

Acknowledgment

The work was supported by the European Union's research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 812991 "PAINLESS" within the Horizon 2020 Program. The experiments within the 5G Playground Carinthia are funded by the Carinthian Agency for Investment Promotion and Public Shareholding (BABEG). The 5G Playground is operated by BABEG and financed by means of the Austrian Federal Ministry for Climate Action, Environment, Energy, Mobility, Innovation and Technology (BMK) and the Carinthian provincial government.

References

[1] A. Fotouhi, H. Qiang, M. Ding, M. Hassan, L. G. Giordano, A. Garcia-Rodriguez, and J. Yuan, "Survey on UAV Cellular Communications: Practical Aspects, Standardization Advancements, Regulation, and Security Challenges," *IEEE Communications Surveys Tutorials*, vol. 21, no. 4, pp. 3417–3442, Mar. , 2019.

[2] M. Mozaffari, W. Saad, M. Bennis, Y. Nam, and M. Debbah, "A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems," *IEEE Communications Surveys Tutorials*, vol. 21, no. 3, pp. 2334–2360, Mar. , 2019.

[3] I. Donevski and J. J. Nielsen, "Dynamic Standalone Drone-Mounted Small Cells," in *Proc. of European Conference on Networks and Communications (EuCNC)*, Dubrovnik, Croatia, Sep. , 2020, pp. 342–347.

[4] I. Kovacs, R. Amorim, H. C. Nguyen, J. Wigard, and P. Mogensen, "Interference analysis for uav connectivity over lte using aerial radio measurements," in *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*. IEEE, 2017, pp. 1–6.

[5] A. Fakhreddine, C. Bettstetter, S. Hayat, R. Muzaffar, and D. Emini, "Handover challenges for cellular-connected drones," in *Proceedings of the 5th Workshop on Micro Aerial Vehicle Networks, Systems, and Applications*, 2019, pp. 9–14.

[6] H. C. Nguyen, R. Amorim, J. Wigard, I. Z. Kovács, T. B. Sørensen, and P. E. Mogensen, "How to ensure reliable connectivity for aerial vehicles over cellular networks," *IEEE Access*, vol. 6, pp. 12 304–12 317, 2018.

[7] B. Van der Bergh, A. Chiumento, and S. Pollin, "Lte in the sky: Trading off propagation benefits with interference costs for aerial nodes," *IEEE Communications Magazine*, vol. 54, no. 5, pp. 44–50, 2016.

[8] X. Lin, R. Wiren, S. Euler, A. Sadam, H.-L. Määttänen, S. Muruganathan, S. Gao, Y.-P. E. Wang, J. Kauppi, Z. Zou *et al.*, "Mobile network-connected drones: Field trials, simulations, and design insights," *IEEE Vehicular Technology Magazine*, vol. 14, no. 3, pp. 115–125, 2019.

[9] S. Hayat, C. Bettstetter, A. Fakhreddine, R. Muzaffar, and D. Emini, "An experimental evaluation of lte-a throughput for drones," in *Proceedings of the 5th Workshop on Micro Aerial Vehicle Networks, Systems, and Applications*, 2019, pp. 3–8.

[10] R. Muzaffar, C. Raffelsberger, A. Fakhreddine, J. L. Luque, D. Emini, and C. Bettstetter, "First experiments with a 5g-connected drone," in *Proceedings of the 6th ACM Workshop on Micro Aerial Vehicle Networks, Systems, and Applications*, 2020, pp. 1–5.

[11] A. Al-Hourani, S. Kandeepan, and A. Jamalipour, "Modeling Air-to-Ground Path Loss for Low Altitude Platforms in Urban Environments," in *Proc. of IEEE Global Communications Conference*, Austin, TX, Oct. , 2014, pp. 2898–2904.

[12] Z. Qiu, X. Chu, C. Calvo-Ramirez, C. Briso, and X. Yin, "Low altitude uav air-to-ground channel measurement and modeling in semiurban environments," *Wireless Communications and Mobile Computing*, vol. 2017, 2017.

[13] G. E. Athanasiadou, M. C. Batistatos, D. A. Zarbouti, and G. V. Tsoulos, "Lte ground-to-air field measurements in the context of flying relays," *IEEE Wireless Communications*, vol. 26, no. 1, pp. 12–17, 2019.

[14] B. Li, Y. Jiang, J. Sun, L. Cai, and C.-Y. Wen, "Development and testing of a two-uav communication relay system," *Sensors*, vol. 16, no. 10, p. 1696, 2016.

[15] R. Gangula, O. Esrafilian, D. Gesbert, C. Roux, F. Kaltenberger, and R. Knopp, "Flying rebots: First results on an autonomous uav-based lte relay using open airinterface," in *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. IEEE, 2018, pp. 1–5.

[16] C. Raffelsberger, R. Muzaffar, and C. Bettstetter, "A performance evaluation tool for drone communications in 4g cellular networks," in *2019 16th International Symposium on Wireless Communication Systems (ISWCS)*. IEEE, 2019, pp. 218–221.

[17] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP Altitude for Maximum Coverage," *IEEE Wireless Communications Letters*, vol. 3, no. 6, pp. 569–572, Dec. , 2014.

[18] I. Donevski, J. J. Nielsen, and P. Popovski, "Standalone Deployment of a Dynamic Drone Cell for Wireless Connectivity of Two Services," in *Proc. of IEEE Wireless Communications and Networking Conference (WCNC)*, no. TBP, Nanjing, China , Apr. , 2021.

Paper D

Performance trade-offs in cyber-physical control applications with multi-connectivity

Igor Donevski, Israel Leyva-Mayorga, Jimmy Jessen Nielsen, Petar Popovski

The paper has been published in the
Frontiers in Communications and Networks Vol. 2, 2021.

The layout has been revised.

Abstract

Modern communication devices are often equipped with multiple wireless communication interfaces with diverse characteristics. This enables exploiting a form of multi-connectivity *known as* interface diversity *to provide path diversity with multiple communication interfaces. Interface diversity helps to combat the problems suffered by single-interface systems due to* error bursts *in the link, which are a consequence of temporal correlation in the wireless channel. The length of an error burst is an essential performance indicator for cyber-physical control applications with periodic traffic, as these define the period in which the control link is unavailable. However, the available interfaces must be correctly orchestrated to achieve an adequate trade-off between latency, reliability, and energy consumption. This work investigates how the packet error statistics from different interfaces impacts the overall latency-reliability characteristics and explores mechanisms to derive adequate interface diversity policies. For this, we model the optimization problem as a partially observable Markov Decision Process (POMDP), where the state of each interface is determined by a Gilbert-Elliott model whose parameters are estimated based on experimental measurement traces from LTE and Wi-Fi. Our results show that the POMDP approach provides an all-round adaptable solution, whose performance is only* 0.1 *% below the absolute upper bound, dictated by the optimal policy under the impractical assumption of full observability.*

1 Introduction

In the rise of the Industry 4.0, the fourth industrial revolution, there is an amassing interest for reliable wireless remote control operations. Moreover, the application of connected robotics, such as in cyber-physical control, is one of the main driver for technological innovation towards the sixth generation of mobile networks [1]. In accord, one of the main use cases for the fifth generation of mobile networks (5G) is Ultra-Reliable and Low-Latency Communication (URLLC) [2]. Reliability and latency requirements for this use case are in the order of $1 - 10^{-5}$ and of a few milliseconds, respectively. The combination of these two conflicting requirements makes URLLC challenging. For instance, hybrid automatic repeat request (HARQ) retransmission mechanisms provide high reliability, but cannot guarantee the stringent latency requirements of URLLC. To solve this, recent 3GPP releases have supported dual- and multi-connectivity, in which data packet duplicates are transmitted simultaneously via two or more paths between a user and a number of eNBs. Hereby, reliability can be improved without sacrificing latency by utilizing several links pertaining to the same wireless technology – 4G or 5G – but at the cost of wasted time-frequency resources [3, 4]. However, modern wireless communication devices, such as smart phones, usually possess numerous wireless interfaces that can be used to establish an equal number of communication

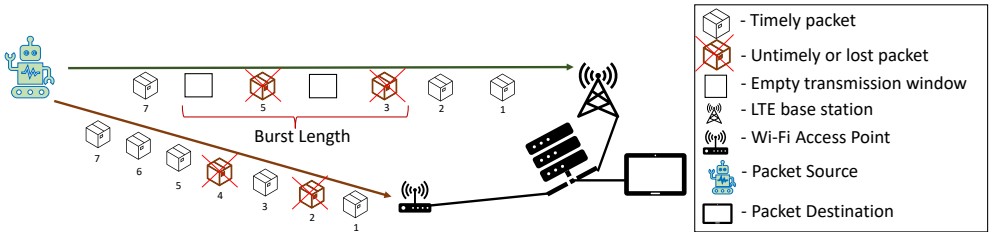


Fig. D.1: An illustration of the scenario investigating interface diversity where the sender duplicates each packet. The sender would sometimes skip transmission windows in favor of conserving energy. Here, only packet 4 was lost, packets 2,3,5,6 were saved, and packets 1, 7 arrived with a redundant copy.

paths. Recent work has proposed *Interface diversity* [5], which expands the concept of dual and multi-connectivity to the case where a different technology per interface can be used. Thereby, lower cost connectivity options can help to increase communication reliability. Since constant packet duplication leads to a large waste of resources, the transmission policies in multi-connectivity and interface diversity systems must be carefully designed to meet the performance requirements while avoiding resource wastage and over-provisioning. Furthermore, as we will observe on the results section, acquiring sufficient knowledge on the channel statistics is essential to attain adequate trade-offs between resource efficiency and reliability in interface diversity systems.

From its definition, the URLLC use case treats each packet individually and, hence, does not capture the performance requirements of numerous applications. For instance, the operation of cyber-physical control applications, that transmit updates of an ongoing process, is usually not affected by individual packets that violate the latency requirements (i.e., untimely packets). Instead, these applications define a *survival time*: the time that the system is able to operate without a required message [6]. Hence, the reliability of communication in such cyber-physical systems is defined by the statistics of consecutive untimely packets, that is, the length of *error bursts*. Hence, in cyber-physical systems, having multiple interfaces with diverse characteristics is greatly valuable, as it allows to select the appropriate interface based on the requirements of the task at hand. For example, while an LTE-based system with multi-connectivity capabilities and selective packet duplication could satisfy the requirements of the application, it seems likely that a combination of unlicensed (e.g. Wi-Fi) and licensed (e.g. LTE) technologies could lead to similar performance guarantees while achieving a lower usage of scarce licensed spectrum and reduce overall costs.

In this paper, we therefore study the performance of interface diversity in terms of burst error distribution in a source-destination system, where we consider two fundamentally different technologies: 1) LTE, which is based on orthogonal frequency division multiple access (OFDMA), operates in licensed spectrum, and where the base station

(BS) schedules the uplink resources for communication; and 2) Wi-Fi, which is based on carrier sense multiple access (CSMA) and operates in unlicensed spectrum. The goal of the proposed interface diversity system, as illustrated in Fig. D.1, is addressing the survival time in cyber-physical control applications [6]. In particular, we investigate the trade-offs between system-lifetime (the time until the system reaches the end of the survival time and operation must be interrupted) and energy consumption. Given the nature of periodic traffic, the survival time can be expressed as the maximum tolerable number of consecutively lost or untimely information packets.

To consider the effect of channel correlation in consecutive errors, we use the Gilbert-Elliott [7] model that is well suited for representing time-correlated transmissions [8]. Using this approach, we formulate the problem as a partially-observable Markov decision process (POMDP) that takes into account the limited observability of the inactive interfaces. Hence, based on the observations and the belief states, we can calculate the optimal transmission policy even for devices with extremely limited computational power. We observed that the performance trade-offs achieved with the POMDP approach are greatly similar when compared the ones achieved with an idealized fully observable MDP. The key contributions of this work are:

- The formulation of an interface diversity problem for energy-constrained devices as a POMDP. Hence, our approach considers the limited observability of the inactive interfaces: those that do not transmit, and thus do not receive feedback. While our results are presented for a device using an LTE and a Wi-Fi interface, our model is sufficiently general and, hence, can be applied to cases with more than two interfaces and to different technologies.

- The analysis of interface diversity policies for cyber-physical control applications, where a certain number of untimely packets are tolerated and with error burst due to the temporal correlation in the wireless channel are considered by means of a Gilbert-Elliott model.

- It illustrates that a computationally simple solution, the Q-MDP value method, can be used for solving the POMDP. Using this method we obtain results that closely follow the performance of the fully observable MDP. Specifically, the expected loss in the reward is only around 0.1%.

The rest of the paper is organized as follows. We initially present an elaborate explanation of multi-connectivity and interface diversity's role in timeliness in Section 2. Next, we present the system model in Section 3, followed by the analysis of the scenario and our proposed method to solve the POMDP in Section 4. Then, we present the numerical results in Section 5. Finally, we conclude the paper with a summary of the work in Section 6.

2 Literature Review

Multi-connectivity has been studied from different perspectives. For instance, [3] studied a scenario with one user equipment (UE) connected to multiple BSs and with multiple simultaneous connections to the same BS. The benefits of this approach are assessed in terms of transmit power reduction, achieved by increasing the signal-to-noise ratio (SNR). Following a similar multi-connectivity approach, a matching problem is formulated by [9], where the number of UEs in the network and the limited wireless resources are considered. The objective is to provide the desired reliability to numerous users by assigning only the necessary amount of resources to each of them. [10] investigated a similar problem in a heterogeneous network scenario with a small cell and a macro cell. Their results show that multi-connectivity is particularly useful for cell-edge UEs connected to the small cell, and provides even greater benefits when URLLC and enhanced mobile broadband (eMBB) traffic coexist. [11] considered multi-connectivity for URLLC as a combination of device-to-device and cellular links, where correlated shadowing is considered. They achieved remarkable increases in the availability ranges for both interfaces. In our previous work [5] we studied the benefits of interface diversity in terms of reliability for a given error probability. Finally, [12] assessed the performance scheduling schemes such as packet duplication and load balancing in order to achieve latency and reliability improvements. The authors exploited a combination of a local Wi-Fi and a private LTE network, that was tested under traffic patterns that are expected to appear in an industrial communications setting.

In the studies mentioned above, only stationary error probabilities are considered. Moreover, [13] provided a thorough investigation of switching off a singular interface that has an unreliable channel, based on channel feedback. The goal of the authors is thus aligned with ours since they aim for an energy-efficient transmission policy given bursty channels, for reliable connectivity of synchronous services. However, the use of different interfaces provides unique benefits for URLLC, especially in the case of bursty wireless errors. For instance, different interfaces are likely to present different burst error distributions, and the correlation of errors between different interfaces is expected to be much lower compared to the correlation between multiple links using the same wireless interface. Despite these evident benefits, and the thorough investigation of burst errors in past research [14], little research has been conducted on interface diversity with error bursts. Specifically, our previous work presents one of the few analyses of this kind [15]. However, it was limited to the benefits of interface diversity in the length of error and success bursts without considering the impact on resource efficiency.

Cyber-physical control applications can belong to one of two major categories depending on the traffic direction requirements: in downlink or uplink only (open-loop control) or the combined uplink and downlink (closed-loop control) requirements [6]. Moreover, a closed-loop control application needs to process incoming events, and thus give appropriate instruction commands to those events [16]. In such scenarios, timeli-

ness is critical to avoid violating the system imposed latency requirements, which leads to executing outdated actions. Therefore, being untimely is the equivalent of a failure in communication service availability.

Open-loop control applications with periodic commands appear frequently in industrial applications and are considered representative of cyber-physical control systems. In these applications, failing a specific number of consecutive updates directly corresponds to exceeding the survival time and, hence, to an error in the system. For example, it has been observed that the number of consecutive errors impacts the stability of the system and leads to a considerable decrease in safety of autonomous guided vehicles [17]. As in the present model and in our previous work [15], a Gilbert-Elliott model was considered in [17] to introduce correlation in the wireless channel. Finally, the novelty of this work comes from investigating the problem of interface diversity for timely packet arrivals for cyber-physical control applications in a burst error channel, where the reliability of the system comes as a trade-off of energy.

3 System model

We consider a point-to-point communication between a user and a BS in an industrial scenario. The user samples a given set of physical phenomena and generates data periodically, where T_s is the sampling period. The sampled data is immediately transmitted to the BS, where it is used for control purposes, so that it must be received within a pre-defined latency constraint $\theta \leq T_s$. Hence, it is now convenient to introduce the definition of the latency-reliability function, which stands for the probability of being able to transmit a data packet from a source to a destination with a given latency deadline [5].

Let L be the RV that defines the packet latency. Then, for a given interface i and latency deadline θ, the latency-reliability function is defined as

$$F_i(\theta) = \Pr(L \leq \theta \mid i). \tag{D.1}$$

As such, the latency-reliability function is a CDF of the interface's latency, where lost packets have the equivalent of infinite latency. Thus, the error probability becomes a specific value (of deadline) Θ in the latency-reliability function, and we define the probability of error for interface i as

$$P_e^{(i)} = 1 - F_i(\theta). \tag{D.2}$$

It should be noted that the traditional definition of the probability of error is obtained for the case $\theta \to \infty$ and that the distribution of L can be updated continuously to reflect the changes in the wireless channel.

We consider the case where the interface diversity system uses packet cloning, where a full packet is transmitted via each of the N interfaces. Next, by assuming that

errors across the multiple available interfaces occur independently, the end-to-end error probability can be calculated as in [5, 18]:

$$P_e^{\mathrm{E2E}} = \prod_{i=1}^{N} (1 - F_i(\theta)) = \prod_{i=1}^{N} P_e^{(i)}. \tag{D.3}$$

Note that the correlation of the large-scale fading across the interfaces is captured by the model described above through the distribution of the RV L. The assumption of errors occurring independently across interfaces holds since correlation in the fast fading may only occur if the antenna elements within an array have insufficient spacing and/or if the concurrent transmissions occur in frequencies that are separated by less than one coherence bandwidth [19]. In out case, the use of two different technologies and frequency bands for WiFi (unlicensed ISM bands) and LTE (licensed spectrum) ensure that the transmissions are sufficiently separated in frequency to avoid correlation.

The BS sends individual feedback per interface to the user after each transmission attempt. If the data is not received within θ, it is declared as missing and the user receives a NACK. The system tolerates a maximum number of missed transmissions. Specifically, if the number of missed transmissions is N, the system declares a failure and operation must be interrupted. Otherwise, the system is able to continue normal operation whenever the number of missed transmissions is $n \leq N$ (smaller than the survival time).

In the following, we define our interface diversity problem as a POMDP denoted as the tuple $(\mathcal{S}, \mathcal{A}, T, \mathcal{R}, \Omega, \mathcal{O})$. Here, $\mathcal{S}$ is the set of states, $\mathcal{A}$ is the set of actions, T is the transition probability to the next state of the *environment* given a state-action pair, $\mathcal{R} \subset \mathbb{R}$ is the set of immediate rewards, Ω is the set of possible observations, and $\mathcal{O}$ is the observation probability when transitioning.

3.1 The environment

The user, i.e. the agent, interacts with the *environment* at discrete time steps $t \in \mathbb{N}$ by sampling and transmitting data to the BS. It is equipped with two distinct communication interfaces, $i \in \{1, 2\}$, where the generated data can be transmitted. Throughout this paper, we assume that these interfaces are completely independent from each other and that each interface is accurately modeled by a two-state Gilbert-Elliott (GE) model [7]. The GE model was selected due to its simplicity and to the ability to capture temporal correlation.

In our GE model, the details of the implemented protocol and the wireless conditions – interference, noise, and fading – are simplified and related to two possible states in a discrete-time Markov chain (DTMC). These are the *good* state G and the *bad* state B. Hence, we model the state space for the GE model for interface i as $\mathcal{S}_i = \{\mathrm{G}, \mathrm{B}\}$. At any given point in time, an interface is in the G state if the protocol and the wireless

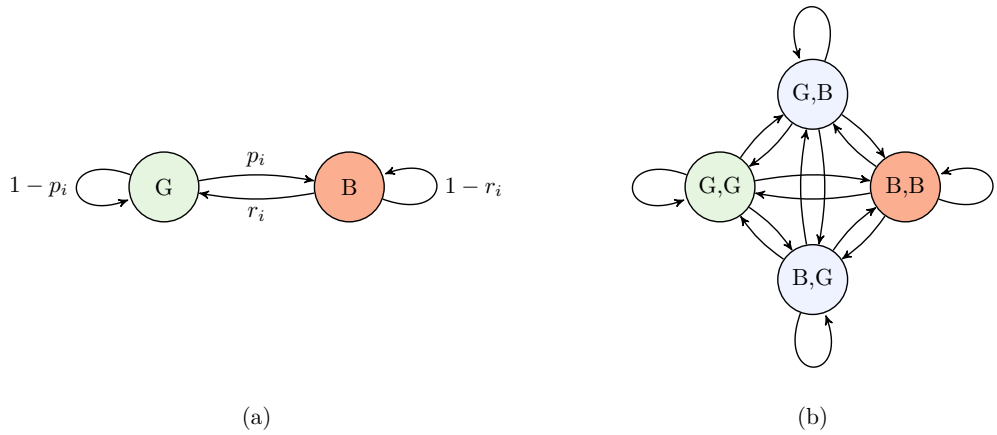


Fig. D.2: (a) Two-state GE model for interface i and (b) four-state GE model for user with two interfaces.

channel conditions are such that allow for a transmission to be received within the latency constraint θ. Otherwise, the interface is in the B state.

This simple GE model has two parameters, namely p_i and r_i that determine the transition probabilities and, hence, the steady-state error probability and burst lengths [7]. Hence, these are system- and environment-specific, and can only be learned after deployment by collecting statistics of the packet transmissions. An additional benefit of using the GE model is that through continuous tuning of the statistical parameters, it allows to capture cross-interface correlation due to large-scale fading or traffic surges.

We denote the state of interface i at time step $t-1$ as s_i and as s_i' at time step t. Parameter p_i represents a transition from state G to B and r_i from B to G (i.e., a recovery from the bad state). Hence, the transition probabilities are defined as

$$\Pr\left(s_i' = \mathrm{G} \mid s_i = \mathrm{G}\right) = 1 - p_i, \tag{D.4}$$
$$\Pr\left(s_i' = \mathrm{G} \mid s_i = \mathrm{B}\right) = r_i, \tag{D.5}$$
$$\Pr\left(s_i' = \mathrm{B} \mid s_i = \mathrm{G}\right) = p_i, \tag{D.6}$$
$$\Pr\left(s_i' = \mathrm{B} \mid s_i = \mathrm{B}\right) = 1 - r_i, \tag{D.7}$$

Fig. D.2a illustrates the GE model with one interface, whose transition probability in a matrix form is:

$$\mathbf{P}_i = \begin{bmatrix} 1 - p_i & p_i \\ r_i & 1 - r_i \end{bmatrix}. \tag{D.8}$$

Building on this, the state of a system with two interfaces is defined by the four-

state GE model illustrated in Fig. D.2b, where transition labels are omitted for brevity. To elaborate, transition probabilities are calculated under the assumption of the two interfaces being independent, for example, the transition from state G,G to state B,B has probability $p_1 p_2$.

Besides the status of each interface, knowing the number of consecutive missed data n is essential for the operation of the system. Therefore, to build a respective Markov decision process (MDP), we define the state space as $\mathcal{S} = \{(s_1, s_2, n)\}$, where $n \in \{0, 1, 2, \ldots, N\}$. Note that the true state of the interface is observable only after concluding the transmission at time step $t-1$, and before the following transmission attempt at time t. Thus, the state of the environment at time t denoted as $S \in \mathcal{S}$ is defined by the outcome of the transmission in the last attempt at $t-1$. Hence, all states $S \in \mathcal{S} : n = 0$ indicate that a transmission at time $t-1$ was received successfully. Furthermore, all states $S \in \mathcal{S} : n = N$ are absorbing states and, consequently, this is a finite MDP with episodic tasks.

Ultimately, the goal of the system is to navigate the MDP in a way that decreases the amount of errors, and altogether reduces the likelihood of having N consecutive errors. Thus, the system should be incentivised to maximize its expected lifetime, while optimizing the costs associated with each transmission. This is done through proper allocation of rewards for each action in the MDP. However, the main challenge for solving the issue comes as a product of the limited observability of the defined MDP system when an interface is switched off. The details for this are encompassed in the following subsection.

3.2 Actions, rewards, and uncertainty

At each time step t (i.e., data transmission instant) the user takes an action $A \in \mathcal{A} = \{(a_1, a_2)\}$, where $a_i \in \{0, 1\}$; $a_i = 1$ indicates transmission and $a_i = 0$ indicates no transmission for interface i. Hereafter we denote that interface i is on when $a_i = 1$ and off otherwise. Note that in our case $\mathcal{A}(S) = \mathcal{A}$ for all $S \in \mathcal{S}$; that is, the set of actions is the same in every possible state. This totals to three different actions – leaving out the option to turn off all interfaces altogether $A = (0, 0)$ – and have either interface off, or both interfaces turned on.

Having taken action A when in state S there is a probability $T(S, A, S')$ to end up in state S', therefore it must apply that $\sum_{S'} T(S, A, S') = 1$. Note that S' represents the true state of both interfaces at time t, that is revealed only after taking action A. A missed transmission can thus occur when both interfaces are transmitting $A = (1, 1)$ but are in the bad state $S' = (B, B, n)$, or a single interface is transmitting that is in its corresponding bad state – $A = (0, 1)$ when $S' = (G, B, n)$, or $A = (1, 0)$ when $S' = (B, G, n)$.

Therefore, following a transmission action A, the user receives a reward R that also depends on the feedback by the BS given before time $t + 1$. Having arrived at state

S' by taking action A when in state S, yields a reward $R(S, A, S')$ that is a function $r(n)$, where $r(n = 0) = 1$ is a successful transmission and $r(n > 0) = -1$ is a missed transmission. In the overall reward allocation, we also account for the cost of using interface i, specifically,

$$R(S, A, S') = r(n) - c(A) = r(n) - a_1 c_1 - a_2 c_2, \tag{D.9}$$

where $c(A)$ is the cost of taking action A.

In an MDP, a policy $\boldsymbol{\pi}$ is a function that maps each state $S \in \mathcal{S}$ to an action $A \in \mathcal{A}(s)$. Therefore, given a policy $\boldsymbol{\pi}(S)$, the agent will choose action A, once it finds itself in state S. Our objective is to find an optimal policy $\boldsymbol{\pi}^*$ that selects the best action given some state S. Let K denote the total number of time steps until the system transitions to an absorbing state. The best action at time t is the one that maximizes the discounted return

$$G_t = \sum_{k=0}^{K-t-1} \gamma^k R_{t+k+1}, \tag{D.10}$$

where $0 < \gamma < 1$ is the discount factor, and the system lifetime is $K < \infty$ as there is always a non-zero probability of transitioning to the absorbing states in a finite number of steps. Thus, the value of the MDP process, when starting from initial state S_{init} under a policy $\boldsymbol{\pi}$, is:

$$V_\pi(s) = \mathbb{E}_\pi \left[G | S = S_{\text{init}} \right]. \tag{D.11}$$

Next, we define the set of observations as $\Omega = \{(o_1, o_2, n)\}$, where $o_i \in \mathcal{S}_i \cup \{0\}$ is the set of observations for an interface i, and the number of consecutive missed deadlines n is always observable. The transition from S to S' following action A provides a deterministic observation $O = (o_1, o_2, n) \in \Omega$ in the following manner: (Since all observations are deterministic with each action, the set $\mathcal{O}$ is irrelevant to the analysis.) I) Having an interface on, namely $a_i = 1$, allows for fully observing the state of that interface $o_i = s_i$. II) On the other hand, having an interface off, namely $a_i = 0$, provides no observation of the state of that interface, i.e. $o_i = 0$, unless additional mechanisms are available to correctly estimate the state, for example, based on the exchange of control messages. Building on this, we define the following function.

$$o_i = \begin{cases} s_i & \text{if } a_i = 1 \text{ OR } \alpha = 1, \\ 0 & \text{otherwise,} \end{cases} \tag{D.12}$$

where $\alpha = 1$ indicates that the BS has additional mechanisms to perform the observation. Fig. D.3 illustrates the action space along with the associated transitions to states S' and observations from an arbitrary non-absorbing state S.

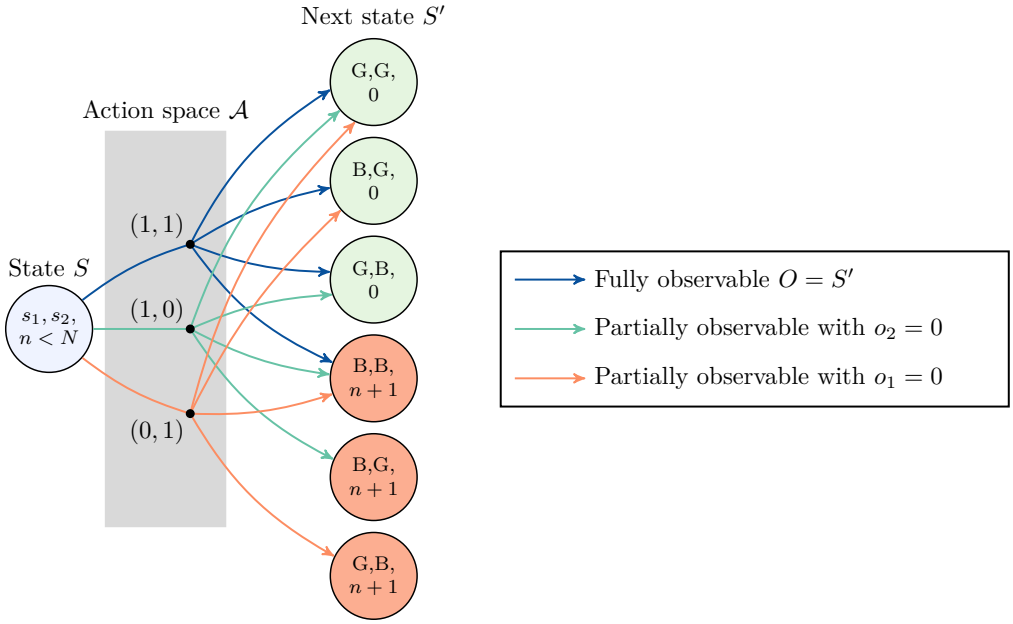


Fig. D.3: One-step transitions from an arbitrary non-absorbing state (i.e., $n < N$) for the partially observable Markov decision process (POMDP) with two interfaces.

4 Analysis

As a first step, we derive the steady-state probabilities of the good and bad states from the transition matrix $\mathbf{P}_i$ as [7]:

$$\pi_{i,\mathrm{G}} = \frac{r_i}{p_i + r_i} \tag{D.13}$$

$$\pi_{i,\mathrm{B}} = \frac{p_i}{r_i + p_i} \tag{D.14}$$

where $\pi_{i,\mathrm{G}} + \pi_{i,\mathrm{B}} = 1$.

Assuming that the system has both interfaces turned on during initialization, the initial state of the MDP S_{init} is chosen randomly among the set of states $\{(G,G,0), (G,B,0), (B,G,0), (B,B,1)\}$ based on the steady state probabilities $\pi_{i,\mathrm{G}}$ and $\pi_{i,\mathrm{B}}$.

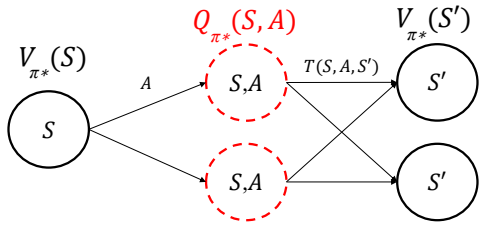


Fig. D.4: An illustration of a single step of the value iteration process for state S that tests the Q-function for all state-action pairs, where future states follow the optimal policy π^*

4.1 Policy Utility Through the Value and Q functions

Let $V_\pi(S)$ be the expected utility received by following policy π from state S, as in:

$$V_\pi(S) = \begin{cases} Q_\pi(S,A) & \text{if } S \notin \{(s_1, s_2, N)\}, \\ 0 & \text{otherwise,} \end{cases} \tag{D.15}$$

where $Q_\pi(S,A)$ is the expected utility of taking action A from state S , and then following policy π [20]. Starting from state S_{init} our goal is to find the optimal policy π^* that results in the maximum value that can be obtained through any policy $V_{\pi^*}(S_{\text{init}})$. As illustrated in Fig. D.4, when not following the optimal policy π^*, but sampling the value of the Q-functions for each action A in the space of values $V_{\pi^*}(S')$, we get:

$$Q_{\pi^*}(S,A) = \sum_{S'} T(S,A,S')[R(S,A,S') + \gamma V_{\pi^*}(S')], \tag{D.16}$$

where, γ is the discount factor that controls the importance of short term rewards (γ values close to 0), or long term rewards (γ values close to 1), where the anticipated rewards are represented through the value of the state as:

$$V_{\pi^*}(S) = \begin{cases} \max\limits_{A \in \mathcal{A}} Q_{\pi^*}(S,A) & \text{if } S \notin \{(s_1, s_2, N)\}, \\ 0 & \text{otherwise,} \end{cases} \tag{D.17}$$

Unless the user is in an N state that is absorbing, the values $V_{\pi^*}(S)$ are recurring and can be approximated through the iterative process of [20]:

$$V^k_{\pi^*}(S) \longleftarrow \max_{A \in \mathcal{A}} \sum_{S'} T(S,A,S')[R(S,A,S') + \gamma V^{k-1}_{\pi^*}(S')], \tag{D.18}$$

that continues until it converges to some predefined precision ϵ of the past and current value:

$$\max_{S \in \mathcal{S}} |V_\pi(S)^k - V_\pi(S)^{k-1}| \leq \epsilon. \tag{D.19}$$

Thus the method of value iteration guarantees finding the optimal value for an MDP that is a function of the optimal policy. Given a κ number of iterations to converge to a solution, the complexity of this algorithm is $O(\kappa SAS')$, which given our small MDP, is insignificant. Unfortunately, this method does not directly produce an optimal policy for a POMDP, however, this can be addressed by the Q_{MDP} value method.

4.2 Belief Averaged Q_{MDP} Value Method

Due to the limited information on the channel properties for each interface our Markov process is a POMDP where we cannot fully observe the true state space for time t. Therefore, the agent maintains a belief b on the state of the system $S \in \mathcal{S}$ based on the observation $O \in \Omega$. By observing that there is no uncertainty on the value of n and that the state of the interfaces $\mathcal{S}_i$ is independent, we can define the belief as:

$$b(S, O) = \Pr\left(S = (s_1, s_2, n) \mid O = (o_1, o_2, n)\right) = b_1(s_1, o_1) b_2(s_2, o_2), \tag{D.20}$$

where b_i is the belief for interface i to be in state s_i, given its observation o_i. The b_i values are updated recursively with each following observation as in:

$$b_i(s_i, o_i) \longleftarrow \Pr\left(s_i \mid o_i\right) = \begin{cases} 1 & \text{if } s_i = o_i \text{ and } o_i \neq 0 \\ f_{s_i}(b_i) & \text{if } o_i = 0, \\ 0 & \text{otherwise,} \end{cases} \tag{D.21}$$

where $f_{s_i}(b_i)$ is a function for calculating the probability of being in state s_i as a function of the previously held beliefs in:

$$f_G(b_i) = (1 - p_i) b_i(G, 0) + r_i b_i(B, 0), \tag{D.22}$$

$$f_B(b_i) = p_i b_i(G, 0) + (1 - r_i) b_i(B, 0). \tag{D.23}$$

Therefore, knowing the belief $b(S, O) \forall S \in \mathcal{S}$ we can proceed with finding an optimal policy for the underlying POMDP through the following two steps.

Step 1: Ignore the observation model and compute the Q-values $Q_{\pi^*}(S, A)$ given directly from the state-action pairs. These are denoted as $Q_{\text{MDP}}(S, A)$ and are obtained through calculating the Bellman operator in the value iteration method.

Step 2: Calculate the belief averaged Q-values for each action and belief $b(S, O)$ as:

$$Q_A(b) = \sum_{S \in \mathcal{S}} b(S, O)\, Q_{\text{MDP}}(S, A)\,. \tag{D.24}$$

The optimal policy now becomes a function of the belief, instead of the current state, and is

$$\pi^*(b) = \max_{A \in \mathcal{A}} Q_A(b). \tag{D.25}$$

Note that this is a method that does not incentivize updating the belief state, but optimizes with the assumption that we will have full observability following the transmission at time t [21].

4.3 Parameter Tuning

For the MDP to optimize the operation of the underlying communications system we require a proper assignment of the rewards and costs for the MDP. Therefore, the reward and punishment for a successful or a missed transmission were fixed to 1 and -1, respectively. Conversely, the value of c_i, the cost of using interface i, greatly depends on the specific characteristics of the system and on the individual notion of resource efficiency. Moreover, the cost of using an interface is directly related to the consumption of resources that would otherwise be available to other services. Throughout the rest of the paper, we consider that the cost of using an interface is given by the energy consumption. However, other parameters can be used to define the cost of each interface when adapting our methods to a specific system.

Given a transmission power E_{LTE} and $E_{\text{Wi-Fi}}$ for the LTE and the Wi-Fi interface, respectively, we calculate the cost for interface i as

$$c_i = \eta \frac{E_i}{\sum_i E_i}, \tag{D.26}$$

where η is a cost scaling factor that serves to reduce/increase the importance of the energy transmission costs with regards to the initial rewards. The scaling factor η was sampled across several values in the range $0 \leq \eta \leq 1$, which resulted in five different optimal policies, one for each different $\eta \in \{0, 0.03, 0.07, 0.2, 1\}$.

4.4 Latency measurements for modeling Wi-Fi and LTE

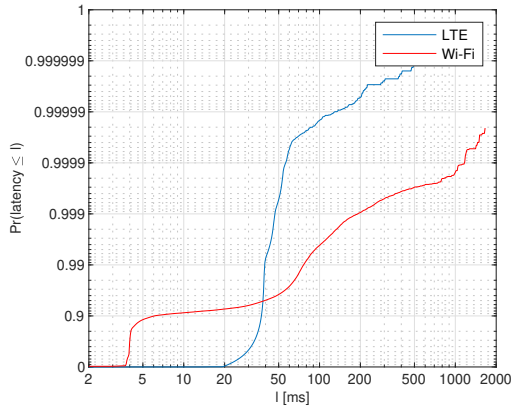


Fig. D.5: Empirical latency CDFs of considered interfaces.

Traces of latency measurements for different communication technologies were obtained by sending small (128 bytes) UDP packets every 100 ms between a pair of GPS

time-synchronized devices through the considered interface (LTE, or Wi-Fi) during the course of a few work days at Aalborg University campus. A statistical perspective of this data is given by the latency CDFs in Fig. D.5, which clearly outlines some key differences between the performance of the LTE and Wi-Fi interfaces. While Wi-Fi can achieve down to 5 ms one-way uplink latency for 90% of packets, it needs approx. 80 ms to guarantee delivery of 99% of packets. For LTE, on the other hand, there is hardly any difference between the latency of 90% and 99% delivery rates, approx. 36 ms and 40 ms, respectively. Since the measurements for both LTE and Wi-Fi were recorded in good high-SNR radio conditions, we expect that the differences between LTE and Wi-Fi can, to a large extent, be attributed to the inherent differences in the protocol operation and the fact that LTE operates in licensed spectrum whereas Wi-Fi has to contend for spectrum access in the unlicensed spectrum.

4.5 Performance evaluation

To conduct the performance evaluation of the policies obtained with the POMDP, we define the following benchmarks.

- *Fully observable system:* Assumes an inherent ability of the BS to inform the user about the interface that is turned off, for example, by using pilots that precede the transmissions. In this case $\alpha = 1$, making the POMDP collapse to an MDP. We denote the policy with full observability as $\boldsymbol{\pi}^*_{\alpha=1}$.
- *Forgetful POMDP (F-POMDP):* Maintains a single state of partial belief and, afterwards, assumes the steady state probability $\pi_{i,\mathrm{G}}$, $\pi_{i,\mathrm{B}}$ for the inactive interface. This forgetful approach collapses to a small MDP where belief does not need to be continuously computed.
- *Hidden MDP (H-MDP):* Is the fully reduced MDP of the forgetful approach, where the belief averages in F-POMDP are joint in a single state. Here, the transition probabilities for the inactive interface directly become the steady state probabilities $\pi_{i,\mathrm{G}}$, $\pi_{i,\mathrm{B}}$.

The obtained policies are evaluated based on the following performance indicators. First, the distribution of the number of consecutive errors n. Second, the utilization of the LTE interface, defined as the ratio of time slots when the LTE interface is turned on $u_{\mathrm{LTE}}(\boldsymbol{\pi}) = \Pr(a_1 = 1 \mid \boldsymbol{\pi})$. Third, the expected system-lifetime, defined as the number of time steps from initialization until the system transitions into an absorbing state. For the latter, let $\overline{K}(\boldsymbol{\pi})$ be the expected system lifetime with policy $\boldsymbol{\pi}$. Finally, we define the expected total reward of the system with policy $\boldsymbol{\pi}$, from initialization until absorption, as $\overline{R}(\boldsymbol{\pi})$.

Building on this, we assess the policies derived with partial observability w.r.t. the policy with full observability based on:

- *System lifetime delta:* Denotes the relative increase of the expected system lifetime w.r.t. the MDP with $\alpha = 1$ (i.e., full observability), denoted as:

$$\Delta K = \frac{\left(\overline{K}(\boldsymbol{\pi}) - \overline{K}(\boldsymbol{\pi}^*_{\alpha=1})\right)}{\overline{K}(\boldsymbol{\pi}^*_{\alpha=1})}. \tag{D.27}$$

  Hence, positive values of ΔK indicate an increase in the system lifetime w.r.t. the optimal policy with full observability.

- *Policy deviation:* Measures the relative change in the expected system lifetime K and expected transmission cost as

$$\Delta\boldsymbol{\pi} = |\Delta K| + |u_{\mathrm{LTE}}(\boldsymbol{\pi}) - u_{\mathrm{LTE}}(\boldsymbol{\pi}^*_{\alpha=1})| \cdot c_{\mathrm{LTE}}. \tag{D.28}$$

  Note that this measures the difference in behavior between w.r.t. to optimal policy but does not necessarily reflect a proportional decrease in performance. Instead, this is an measure of the normalized collective error, as in common estimators that try to project the optimal LTE-usage and system lifetime.

- *Relative reward loss:* Defines the relative loss in the expected total reward $\overline{R}(\boldsymbol{\pi})$ with policy $\boldsymbol{\pi}$ w.r.t. $\boldsymbol{\pi}^*_{\alpha=1}$ as

$$\mathcal{L}(\boldsymbol{\pi}) = \frac{\left|\overline{R}(\boldsymbol{\pi}^*_{\alpha=1}) - \overline{R}(\boldsymbol{\pi})\right|}{\overline{R}(\boldsymbol{\pi}^*_{\alpha=1})} \tag{D.29}$$

The results with the fully observable MDP were obtained analytically. In order to evaluate the performance of the POMDP and forgetful methods, analytical results were obtained for extreme values of parameter η. For all other cases, we performed Monte-Carlo simulations of 20000 episodes. The duration of each episode depends on the system lifetime which could last up to several million time steps.

5 Results

In this section we investigate the performance of the modelled system. The investigation in this section is guided by the use of interface diversity in the case a combination of Wi-FI and LTE. The performance of the aforementioned system where $i = 1$ is LTE and $i = 2$ is Wi-Fi was evaluated by a Monte-Carlo Matlab simulation (when necessary) where the calculation of the statistical properties for the GE model are derived from experimental latency measurements.

We tested the system for all 5 different values $\eta = 0, 0.03, 0.07, 0.2, 1$ where the fully observable MDP system had different transmission policies. Given the measurements and the characteristics of our measurement setup, we tuned the simulation parameters to the values in Table D.1.

Table D.1: Parameters for evaluation

Label	Definition	Value
θ	Latency Constraint	38.25 ms
p_{LTE}	LTE's p-transition probability	0.0178
$p_{\text{Wi-Fi}}$	Wi-Fi's p-transition probability	0.0515
r_{LTE}	LTE's r-transition probability	0.2577
$r_{\text{Wi-Fi}}$	Wi-Fi's r-transition probability	0.9468
$E_{\text{Wi-Fi}}$	Power consumption of Wi-Fi interface	15.85 mW
E_{LTE}	Power consumption of LTE interface	200 mW
N	Maximum number of consecutive missed transmissions	4
ϵ	Value iteration convergence criteria	10^{-11}
$k_{\max}$	Value iteration maximum number of iterations	10^5
γ	Discount factor	0.99999

5.1 Extreme Policies

As a starting point, we describe and evaluate the policies obtained in the cases where the value of parameter $\eta \in (-\infty, \infty)$ is set to a extremely low or high value (i.e., at either of the extremes of its range). When adjusting the scaling factor to its lowest possible value $\eta \to -\infty$, the cost of using each interface is omitted and the MDP optimizes in favor of not losing any transmissions. Thus, we observe an extreme behavior that is not affected by the belief, or the POMDP behavior. Specifically, the optimal policy maintains both interfaces turned on no matter the current state $\pi^*(S) = A(1,1)\,\forall S \in \mathcal{S}$. Since the utilization of both interfaces is 100%, this gives the upper bound on burst error performance for the whole system. Even in this case, there is a non-zero probability to end up in the absorbing state which happens with an expectation of $5.0738 \cdot 10^6$ transitions. This is the defined lifetime of the system, and before turning-off due to failure the system maintains successful transmissions for 99.67% of the time, 0.32% of transmissions have a single error, 0.0127% have two consecutive errors, and $4.9971 \cdot 10^{-4}$% of all burst errors have three consecutive errors.

On the other hand, setting $\eta \to \infty$ creates a lower bound of the system performance that aims to minimize the cost of operation at the expense of decreasing the system lifetime. This is the result of scaling the cost of performing a transmission to be higher than the reward of maintaining successful transmissions. Since the action space is restricted to use at least one interface for transmission at all times, in such a cost

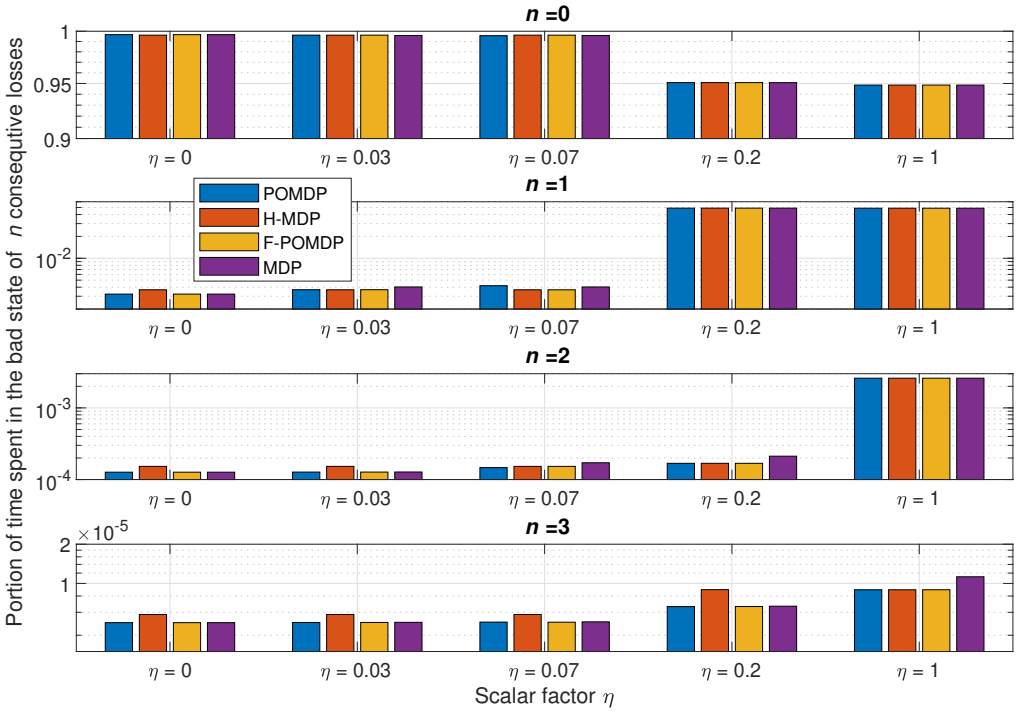


Fig. D.6: Portion of time spent in state with n consecutive untimely packets for all simulated and analytically extracted data.

restraint system, it is reasonable to only allow for the utilization of the Wi-Fi interface $\pi^*(S) = A(0,1)\,\forall S \in \mathcal{S}$, due to the high cost of using LTE. Since we have 100% utilization of Wi-Fi and 0% utilization of LTE the system lifetime of the system decreases drastically to $1.3633 \cdot 10^5$. During operation, the system maintains successful transmissions for 94.84% of the time, 4.89% of transmissions have a single error, 0.26% have two consecutive errors, and 0.0138% of all burst errors have three consecutive errors. All the implementations with other values of η result in policies that exploit mixtures of actions and could not be analytically extracted – for the POMDP and H-MDP implementations – and are thus extracted through Monte-Carlo simulations, as detailed in the previous section.

5.2 Optimal Policies with Scaled Costs

The portion of time spent in states with n consecutive untimely packets, obtained from the simulations, are shown in Fig. D.6 as a function of η, from which we can extract

several conclusions. Initially, we notice a sharp decay for the portion of time spent in good states when comparing the values with $\eta = 0.07$ and $\eta = 0.2$. In accord, we notice a sharp increase increase in all bad states, that is most significant for single errors. This manifests in the optimal policy, as a reluctance of mitigating single burst errors ($n = 1$). Notwithstanding this increase in single errors, all approaches still mitigate higher orders of error bursts ($n > 1$) when $\eta = 0.2$. This is not true for $\eta = 1$, where all approaches focus solely on mitigating the last error that may lead to exceeding the survival time N.

It is important to notice that due to the fact that H-MDP treats the GE model as hidden when a portion of it is unobservable, turning off an interface results in fully losing the state for that interface. This leads to a behavior where the H-MDP would intentionally turn off the interface that observes a bad state, even when there is no negative incentive to keeping that interface on, in favor of the more likely transition to the steady state of the good state for that interface. Due to this, the optimal policy for H-MDP is never $\pi^*(S) = A(1,1)\,\forall S \in \mathcal{S}$, even when $\eta = 0$. Interestingly, H-MDP uses the same policy for all three $\eta = 0, 0.03, 0.07$, but is best fit for $\eta = 0.07$. This makes H-MDP the only suboptimal approach – out of all four – for $\eta = 0$.

Due to the low cost of using the Wi-Fi interface, and generally superior r probability, all policies keep the Wi-Fi interface at 100% utilization. On the other hand, LTE utilization is the only that varies for each approach, and is shown in Fig D.7. However,

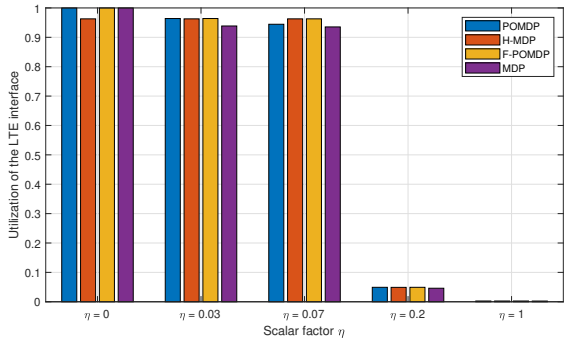


Fig. D.7: Portion of time spent using the LTE interface $i = 1$.

we are interested in investigating the lifetime of the system given burst error tolerance of N. The system lifetime for the different scaling factors η is given in Fig. D.8 as a difference from the optimal system lifetime. Therefore, the goal of each approach with limited observability is to follow the performance of the optimal, fully observable, approach as closely as possible. Thus, when a policy improves the system-lifetime, it is a sign of energy-inefficiency that comes in the form of extra LTE-interface utilization. Accordingly, the goal of all three approaches that have to work with limited information

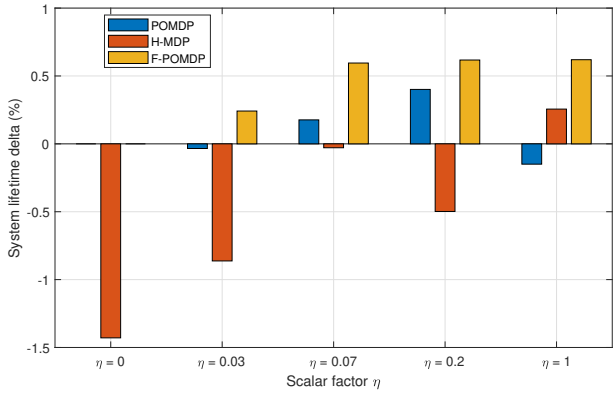


Fig. D.8: Relative increase in the system lifetime ΔK (i.e., in the time to reach one of the absorbing states with N consecutive errors) w.r.t. to the fully observable, optimal policy of MDP .

is to achieve $\Delta K \approx 0$. Looking at Fig. D.8, we can also notice that aside from the case for $\eta = 0.07$, the POMDP approach gives the least deviations with regards to the other two approaches. Moreover, the good performance of the H-MDP approach in the case for $\eta = 0.07$ is a simple coincidence since this approach applies exactly the same policy for $0 \leq \eta \leq 0.07$, where both the POMDP and the F-POMDP tend to vary and adapt. Additionally, we notice that the F-POMDP approach is highly focused towards increasing the system lifetime which, as shown in Fig. D.7, comes at the cost of using the LTE more often than with of the optimal approach. Since this behavior is quite consistent, we can safely say that the F-POMDP is a system-lifetime conservative approach. The POMDP approach is however more adaptable, and consistently bests the F-POMDP in replicating the system lifetime of the fully observable MDP.

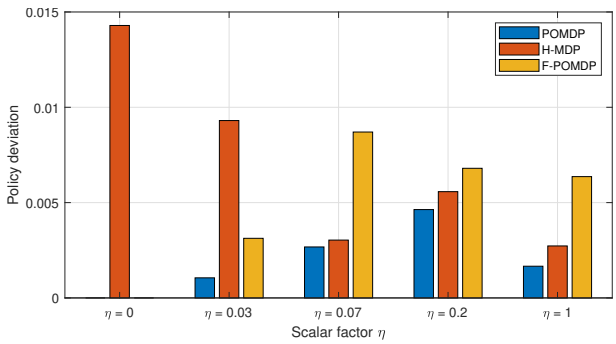


Fig. D.9: The deviation from the optimal MDP policy of full observably.

Finally, in Fig. D.9 we show the aggregate deviation, in terms of system lifetime and energy, as calculated as in (D.28). Here we can see that the POMDP approach provides the most-adaptable behavior, that best resembles the policy when having full observability. Treating the system as a hidden MDP does yield some adaptability, however, the approach can lead to large deviations from the optimal behavior, as it can be seen for $\eta = 0, 0.03$. In these cases, since the stochastic process was treated as hidden to the MDP, the H-MDP optimal solution would intentionally turn off the LTE interface when it is in the bad state. With this, H-MDP fully loses the information of the LTE interface, in favor of the better stationary state probabilities. Due to this, we consider the H-MDP approach as unsuitable. On the other hand, the F-POMDP was always conservative with regards to the system-lifetime. Thus, F-POMDP approach is ideal for implementation in devices with extreme power limitations, as it does not require re-computation of the belief states continuously. Finally, the POMDP approach provides the best solutions that most closely follow the optimal policy. Hence, it presents the best solution given that an accurate model of the environment is available and should be adopted if the energy consumption of the computational circuit, that is dedicated for updating the belief states, is not an issue. In the following, we present a sensitivity analysis of the considered methods under an imperfect model of the environment.

5.3 Sensitivity Analysis

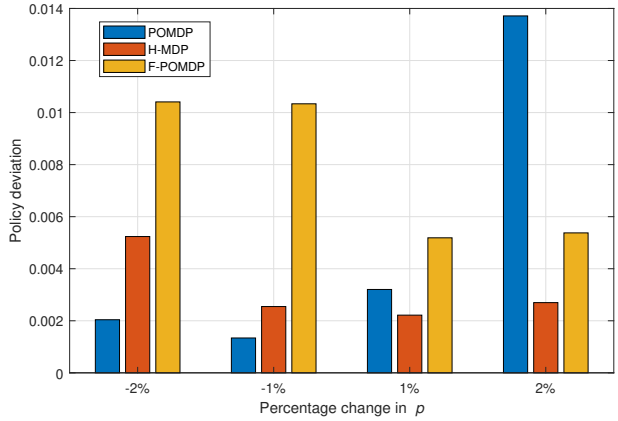


Fig. D.10: The deviation from the optimal MDP policy of full observably when having an error in estimating the p and r values, for the case of $\eta = 0.07$.

We conclude the section by evaluating the impact of the estimation error regarding the p and r values for $\eta = 0.07$. We do this by adding a percentage of error to the p value, while the r is calculated to maintain the same steady state probabilities $\pi_{i,\mathrm{B}}$

and $\pi_{i,\mathrm{G}}$ for each interface i as derived from the values in Table D.1. In this way, the true probabilities of the Markov system are hidden from the decision processes. As Fig. D.10 shows, in the case of a negative percentage change for the system, F-POMDP approach greatly deviates from the optimal policy for $\eta = 0.07$. Additionally, the POMDP approach deviates considerably when an error of 2% is introduced. Moreover, the H-MDP is the most robust as it is more reluctant to change policies in the presence of different parameters, which shows best in the case of positive errors. We can conclude that while belief mechanics help adapt to the optimal policy in the case where the model of the environment is perfectly known, however, such implementations can lead to bad results in particular scenarios where the true probabilities of the system are hidden from the agent. On the other hand, the H-MDP system does not show a big disadvantage in those cases since it already treats the Markov process as hidden.

To conclude, we show the relative reward loss $\mathcal{L}(\boldsymbol{\pi})$ for the same cases of estimation error, as relative to the optimal MDP policy, in Fig. D.11. Here we observe that, even though the F-POMDP method deviates considerably from the optimal MDP policy in the negative estimations of -1% -2% (see Fig. D.10), its rewards are close to those with the optimal policy. The reward loss with the H-MDP are relatively stable and do not exceed 0.6%. Finally, we see that the POMDP implementation generally achieves a small reward loss, which is around 0.1% for no error and around 0.4% for -2%. Nevertheless, a high loss is achieved with 2% error. In this case, the adaptability of the POMDP has a negative effect since it scales the policy in accord with the erroneous p and r values.

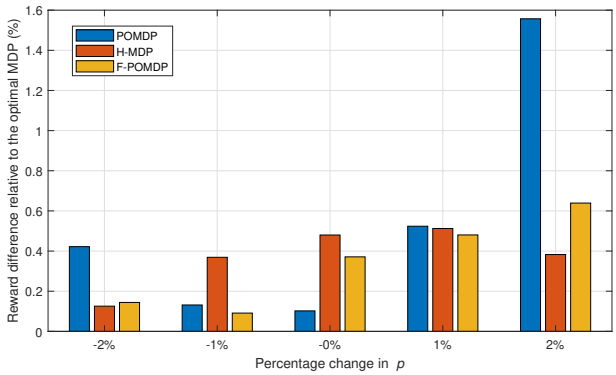


Fig. D.11: The system rewards with regards to the optimal MDP policy of full observably when having an error in estimating the p and r values, for the case of $\eta = 0.07$.

6 Conclusion

Motivated by the recent requirements for cyber-physical systems, we analyzed the problem of addressing error bursts by using two different wireless interfaces. We model the problem as a Gilbert-Elliott model with good and bad states for each interface. Given limited energy resources on our device, we derived and evaluated transmission policies to achieve an adequate trade-off between system lifetime and energy consumption with limited channel information. For this reason, we modelled the system as a POMDP that memorizes and calculates its belief for the observable states. Using Value Iteration to extract the Q-values from the MDP, we update the policy for the POMDP through the Q_{MDP} technique. Our results show that the POMDP approach indeed produces near-optimal policies when the environment is accurately characterized. As such, this is a computationally inexpensive solution that closely follows the performance of the optimal policy, even in cases with various and mixed state-action pairs. We also propose a forgetful F-POMDP approach with only two finite belief states. This approach performs worse than the classic POMDP, with affinity to increase system-lifetime, but is well suited for approaches that are under extreme energy limitations. Finally, in future works we would like to practically validate the usefulness of the system onto several application scenarios, and address dynamic systems in non-stationary or non-characterized environments.

Conflict of Interest Statement

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Author Contributions

ID: formal analysis and software development; ID,ILM,JJN: conceptualization, investigation, writing; ID,ILM,JJN,PP: review, editing; PP,JJN: resources, funding acquisition, supervision, and project administration.

Funding

The work was supported by the European Union's research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 812991 "PAINLESS" within the Horizon 2020 Program.

References

[1] W. Saad, M. Bennis, and M. Chen, "A vision of 6g wireless systems: Applications, trends, technologies, and open research problems," *IEEE Network*, vol. 34, no. 3, pp. 134–142, Oct. 2020.

[2] P. Popovski, Č. Stefanović, J. J. Nielsen, E. De Carvalho, M. Angjelichinoski, K. F. Trillingsgaard, and A.-S. Bana, "Wireless access in ultra-reliable low-latency communication (URLLC)," *IEEE Transactions on Communications*, vol. 67, no. 8, pp. 5783–5801, 2019.

[3] A. Wolf, P. Schulz, M. Dorpinghaus, J. C. S. Santos Filho, and G. Fettweis, "How reliable and capable is multi-connectivity?" *IEEE Transactions on Communications*, vol. 67, no. 2, pp. 1506–1520, feb 2019.

[4] M.-T. Suer, C. Thein, H. Tchouankem, and L. Wolf, "Multi-connectivity as an enabler for reliable low latency communications—an overview," *IEEE Communications Surveys Tutorials*, vol. 22, no. 1, pp. 156–169, 2020.

[5] J. J. Nielsen, R. Liu, and P. Popovski, "Ultra-reliable low latency communication using interface diversity," *IEEE Transactions on Communications*, vol. 66, no. 3, pp. 1322–1334, 2018.

[6] 3GPP, "Technical specification TS 22.104 V16.5.0. 5G; Service requirements for cyber-physical control applications in vertical domains," 2021.

[7] G. Haßlinger and O. Hohlfeld, "The Gilbert-Elliott model for packet loss in real time services on the Internet," in *14th GI/ITG Conference-Measurement, Modelling and Evaluation of Computer and Communication Systems*. VDE, 2008, pp. 1–15.

[8] A. Willig, M. Kubisch, C. Hoene, and A. Wolisz, "Measurements of a wireless link in an industrial environment using an IEEE 802.11-compliant physical layer," *IEEE Transactions on Industrial Electronics*, vol. 49, no. 6, pp. 1265–1282, 2002.

[9] M. Simsek, T. Hößler, E. Jorswieck, H. Klessig, and G. Fettweis, "Multiconnectivity in multicellular, multiuser systems: A matching-based approach," *Proceedings of the IEEE*, vol. 107, no. 2, pp. 394–413, 2019.

[10] N. H. Mahmood, M. Lopez, D. Laselva, K. Pedersen, and G. Berardinelli, "Reliability oriented dual connectivity for URLLC services in 5G new radio," in *Proc. International Symposium on Wireless Communication Systems (ISWCS)*, 2018, pp. 1–6.

[11] C. She, Z. Chen, C. Yang, T. Q. S. Quek, Y. Li, and B. Vucetic, "Improving network availability of ultra-reliable and low-latency communications with multi-connectivity," *IEEE Transactions on Communications*, vol. 66, no. 11, pp. 5482–5496, 2018.

[12] M.-T. Suer, C. Thein, H. Tchouankem, and L. Wolf, "Evaluation of multi-connectivity schemes for URLLC traffic over WiFi and LTE," in *Proc. 2020 IEEE Wireless Communications and Networking Conference (WCNC)*, 2020, pp. 1–7.

[13] D. Dzung, R. Guerraoui, D. Kozhaya, and Y.-A. Pignolet, "To transmit now or not to transmit now," in *2015 IEEE 34th Symposium on Reliable Distributed Systems (SRDS)*, 2015, pp. 246–255.

[14] M. Yajnik, S. Moon, J. Kurose, and D. Towsley, "Measurement and modelling of the temporal dependence in packet loss," in *Proc. IEEE INFOCOM '99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No.99CH36320)*, vol. 1, 1999, pp. 345–352 vol.1.

[15] J. J. Nielsen, I. Leyva-Mayorga, and P. Popovski, "Reliability and error burst length analysis of wireless multi-connectivity," in *Proc. International Symposium on Wireless Communication Systems (ISWCS)*, aug 2019, pp. 107–111.

[16] N. Ploplys, P. Kawka, and A. Alleyne, "Closed-loop control over wireless networks," *IEEE Control Systems Magazine*, vol. 24, no. 3, pp. 58–71, 2004.

[17] P. M. de Sant Ana, N. Marchenko, P. Popovski, and B. Soret, "Wireless control of autonomous guided vehicle using reinforcement learning," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, dec 2020.

[18] R. Billinton and R. N. Allan, *Reliability evaluation of engineering systems.* Springer, 1992.

[19] Y. Chen, A. Wolf, M. Dörpinghaus, J. C. S. S. Filho, and G. P. Fettweis, "Impact of correlated fading on multi-connectivity," *IEEE Transactions on Wireless Communications*, vol. 20, no. 2, pp. 1011–1022, Oct. 2021.

[20] M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming.* John Wiley & Sons, 2014.

[21] M. L. Littman, A. R. Cassandra, and L. P. Kaelbling, "Learning policies for partially observable environments: Scaling up," in *Machine Learning Proceedings 1995*, A. Prieditis and S. Russell, Eds. San Francisco (CA): Morgan Kaufmann, 1995, pp. 362–370. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9781558603776500529

Paper E

On Addressing Heterogeneity in Federated Learning for Autonomous Vehicles Connected to a Drone Orchestrator

Igor Donevski, Jimmy Jessen Nielsen, Petar Popovski

The paper has been published in the
Frontiers in Communications and Networks Vol. 2, 2021.

The layout has been revised.

Abstract

In this paper we envision a federated learning (FL) scenario in service of amending the performance of autonomous road vehicles, through a drone traffic monitor (DTM), that also acts as an orchestrator. Expecting non-IID data distribution, we focus on the issue of accelerating the learning of a particular class of critical object (CO), that may harm the nominal operation of an autonomous vehicle. This can be done through proper allocation of the wireless resources for addressing learner and data heterogeneity. Thus, we propose a reactive method for the allocation of wireless resources, that happens dynamically each FL round, and is based on each learner's contribution to the general model. In addition to this, we explore the use of static methods that remain constant across all rounds. Since we expect partial work from each learner, we use the FedProx FL algorithm, in the task of computer vision. For testing, we construct a non-IID data distribution of the MNIST and FMNIST datasets among four types of learners, in scenarios that represent the quickly changing environment. The results show that proactive measures are effective and versatile at improving system accuracy, and quickly learning the CO class when underrepresented in the network. Furthermore, the experiments show a tradeoff between FedProx intensity and resource allocation efforts. Nonetheless, a well adjusted FedProx local optimizer allows for an even better overall accuracy, particularly when using deeper neural network (NN) implementations.

1 Introduction

The adoption of ubiquitous Level-5 fully independent system autonomy in road vehicles (as per the SAE ranking system [1]) is barred from progress due to the omnipresence of chaotic traffic in legacy traffic situations. Moreover, a 38% share of prospective users are sceptical of the performance of the autonomous driving systems [2]. As such, lowering the number of negative outcome outliers in autonomous vehicle operation, particularly ones that lead to fatal incidents, can be addressed with an overabundance of statistically relevant data [3]. Thus, given the privacy requirements and the abundance of the data that is produced by road vehicles and/or unmanned aerial vehicles (UAVs) in the role of traffic monitors, the machine learning (ML) problem can be addressed by treating the participatory vehicles as learners in a federated learning (FL) network.

In more detail, FL is an ML technique that distributes the learning across many learners. In this way, many separate models are aggregated in order to acquire one general model at server side [4]. In FL, each learner does not have to send heaps of data to a common server for processing, but maintains the data privately. As such, the concept of FL is an extension of distributed ML with four important distinctions: (1) the training data distributions across devices can be non-IID; (2) not all devices have similar computational hardware; (3) FL scales for networks of just few devices to vast

networks of millions; (4) FL can be engineered in a way in which privacy is conserved. Given the vast complexity of implementing FL in autonomous vehicular traffic, particularly related to the quickly changing environment, in this paper we focus on solving the issues of non-IID data learnt across several devices with unequal processing power. We proceed with a review on relevant FL literature below.

1.1 State of the Art

FL is an emergent field that has gained immense popularity in the last five years. From the relevant literature we highlight several works. [5] covers the state of the art regarding computational models, [6] contains a clear understanding of the FL potential and its most prominent applications. [7] and [8] provide comprehensive coverage on the communications challenges for the novel edge computation, [9] analyzes scenarios of FL where learners use wireless connectivity. Challenges and future directions of FL systems in the context of the future 6G systems is given in [10], while [11] elaborates upon the applications of FL on connected automated vehicles and collaborative robotics. [12] covers resource allocation and incentive mechanisms in FL implementations. Most of the works on FL concerning UAVs treat the devices as learners [13–15]. This requires mounting heavy computational equipment on-board, and therefore it is an energy inefficient way of exploiting drones. In contrast, in our prior work [16] we have investigated techniques for reducing staleness when a UAV acts as an orchestrator by optimizing its flying trajectory.

There is also an interest in wireless resource allocation optimization for FL networks, as covered in the topics that follow. The work of [17] proposes a detailed communications framework for resource allocation given complex wireless conditions and an FL implementation on IID data. This work has a strong contribution to the topic of convergence analysis of wireless implementations of FL with very detailed channel model. The work of [18] does a detailed convex analysis for distributed stochastic gradient decent (SGD) and optimizes the power allocation for minimizing FL convergence times. The work of [19] formulates FL over wireless network as an optimization problem and conducts numerical analysis given the subdivided optimization criteria. However, the aforementioned works perform their analysis on SGD which has been shown to suffer in the presence of non-IID data and unequal work times [20]. The novel local subproblem that includes a proximal optimizer in [20] achieves 22% improvements in the presence of unequal work at each node.

The learning of both single task and multi task objectives in the presence of unequal learner contributions is a difficult challenge and has received a lot of attention, e.g. in the works of [21–23]. This also leads to the question of analyzing contributions among many learners with vastly different hardware that is considered in works covering FL incentive mechanisms, by [12, 24–27]. The incentive based FL implementations rely on

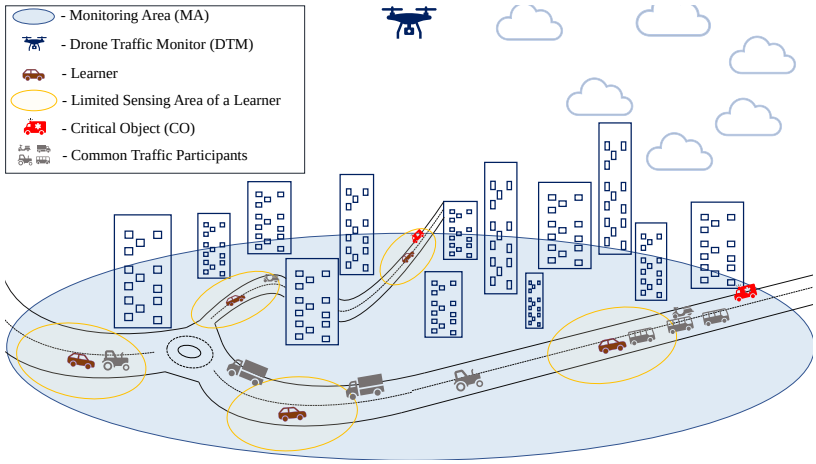


Fig. E.1: Illustration of the DTM covered monitoring area, with five scattered learners.

estimating each learner's contribution and rewarding them for doing the work. Hence calculating appropriate rewards becomes a difficult challenge that also comes at the price of computation and communications as shown by [28]. Such mechanisms are useful when orchestrating an FL where learners would collect strongly non-IID data and learn with vastly different processing capabilities.

1.2 Drone Traffic Monitors as FL Orchestrators

Unmanned aerial vehicles (UAVs) or drones could provide an essential aid to the vehicular communication networks by carrying wireless base stations (BSs). In combination with the 5G standardisation and the emerging 6G connectivity, drone-aided vehicular networks (DAVNs) [29] are capable of providing ultra reliable and low latency communications (URLLC) [30, 31] when issuing prioritized and timely alarms. In accord, most benefits of DAVNs come as consequence of the UAV's capability to establish line of sight (LOS) with very high probability [32]. The good LOS perspective also benefits visual surveillance, hence enabling UAVs to offer just-in-time warnings for critical objects (COs) that can endanger the nominal work of autonomous vehicles. Though DAVNs expect many roles from the drone, we draw inspiration from UAVs in the role of drone traffic monitors (DTMs) that continuously improve and learn to perform timely and reliable detections of COs. To avoid requiring a plethora of drone-perspective camera footage of the traffic, we propose DTMs that take the role of a federated learning (FL) orchestrator, and autonomous vehicles participate as learners.

This FL architecture with a drone orchestrator, illustrated in Fig. E.1, exploits

the processing and sensing enabled vehicles contained in the monitoring area (MA) to participate both as learners and supervisors. The vehicle-learners receive the drone provided footage, and do the heavy computational work of ML training for the task of computer vision. This is possible since the vehicle-learners have robust sensing capabilities, and when they have the CO in view, can contribute to the learning process due to their secondary perspective [33] on the object, and their deeper knowledge of traffic classes. However, even when assuming perfect supervision by the learners, FL is not an easy feat since some knowledge can be obfuscated among omnipresent information and/or contained at computationally inferior straggler learners. In accord, we use a combination of state of the art FL implementation with a novel resource aware solution for balancing work times and learner contributions, which are described in the overview that follows.

1.3 Main Contributions

In this paper, we provide a novel perspective on continuous DTM improvements through an FL implementation onto vehicle-learners. Moreover, we aim to provide a robust and adaptable resource allocation method for improved FL performance in the presence of chaotic, quickly changing, and most importantly imbalanced and non-IID data. Since both computational and data bias cannot be analytically extracted before sampling the ML model received from each learner, we assume heuristic measures such as maximizing the epochs computed, or equalizing the epochs computed across the learners. Moreover, the core contribution of this work is a dynamic resource allocation method based on each learner's past contributions. To provide full compatibility with heterogeneous learners and non-IID data, we employ these methods in combination with the FedProx algorithm. Finally, we developed an experimental analysis in which the performance is evaluated through its capability to learn an underrepresented class of the dataset, while also balancing overall system accuracy.

The paper organisation goes as follows. Section 2 introduces the learning setup and the communications resource allocation setup. Section 3 defines the optimization problem and lists several static and reactive heuristic measures for improving the learning performance, and introduces the learner contribution calculations. This is followed by Section 4 where the experimental setup and the results from the setup are presented. The final, Section 5 summarizes the outcomes and discusses future directions.

2 System Model

The setup is depicted in Fig. E.2, where we show the orchestrator block that sends and receives the models through wireless connections, while simultaneously broadcasts

Table E.1: Relevant symbols of variables, constants and functions.

Symbol	Definition
$h_{\mathrm{MAX}}()$	Utility function that maximizes the number of computed epochs.
$h_{\mathrm{AAS}}()$	Utility function that minimizes the average anchored staleness.
$h_{\mathrm{ACT}}()$	Utility function that maximizes based on the estimated contributions from each learner.
$F()$	Local machine learning optimization function
$f()$	Global (network-wide) optimization function
$E()$	Model evaluation function
$\omega_{\mathrm{g}\setminus\{k\},i}$	Custom model aggregator that excludes the k learner's model
i	An integer indicating the FL cycle/round
k	Learner index number
K	Total number of leanrners in the MA
$\mathbf{T}_i$	Vector representation of the epochs computed across all learners for round i
$\mathbf{G}_i$	Vector representation of the contributions computed, for all learners, for round i
$\mathbf{S}_i$	Vector representation of bandwidth allocated for each learner for round i
$G_{k,i}$	Estimated contribution for learner k, at round i
$\omega_{\mathrm{g},i}$	The global ML model weights for round i
$\omega_{k,i}$	The ML model weights produced at learner k for round i
$\tau_{k,i}$	Epochs computed at learner k for round i
B	The size of the batch computed at each epoch
μ	Proximal term intensity in the FedProx FL implementation
f_k	Processing capability of learner k in terms of epochs per millisecond
W	Total bandwidth allocated for the system
D	Total data transmitted in both directions to a single learner within a single round
R_{avg}	Channel data rate in symbols per hertz
$S_{k,i}$	Bandwidth allocation coefficient for learner k and round i
α	Computation phase duration coefficient (in milliseconds)
β	Communication phase duration coefficient (in milliseconds)
$S_{\min}$	The lower bound of the bandwidth allocation coefficient
$S_{\max}$	The extreme bound of the bandwidth allocation coefficient

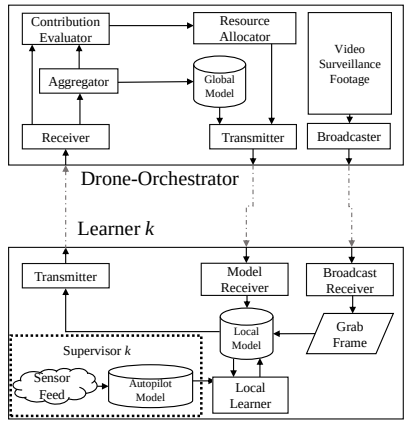


Fig. E.2: System model illustration.

the unsupervised video surveillance footage at a constant data rate for all vehicles inside the MA. We assume that each vehicle acts as an ideal supervisor for the objects which are represented both in the broadcasted video and their sensor feed. Given some deadline of completion T, the learner needs to return its locally learnt model to the drone-orchestrator. After receiving the model, the orchestrator aggregates the K models, after which it can also evaluate the contribution of each learner separately. Each learner k has a contribution, that the contribution estimator estimates to be $G_{k,i}$, for some FL cycle/round i. Finally, the orchestrator contains a resource allocator module that based on the aforementioned information can readjust the wireless resources for the next round, in a way that it improves the FL process.

2.1 Federated Learning

The FL process starts when the orchestrator sends its weights to all K learners, where each learner $k \in \mathcal{K} = \{1, 2, .., K\}$ is present in the MA. The goal of FL methods [4] is to coordinate the optimization of a single global learning objective $\min_\omega f(\omega)$, where the function $f()$ is calculated across the whole network at each round i as:

$$f(\omega) = \sum_{k}^{K} p_k F_k(\omega) = \mathbb{E}[F_k(\omega)], \tag{E.1}$$

where ω are the instantaneous value of the local model weights, $F_k(\omega)$ is the local optimization function at each node, $p_k \geq 0$ and $\sum_k p_k = 1$ is the averaging weight when aggregating. In a single FL round $i \in \mathbb{Z}^+$, a server, i.e. the DTM-orchestrator,

has a global model with weights $\omega_{g,i}$. On round i each k-th learner receives the model and computes $\tau_{k,i}$ epochs of solving the local optimization function $F_k()$, with data batches of size B. Each batch represents a sample of items that have been sensed and collected from that learner's surroundings. The distributed training process produces a new set of weights $\omega_{k,i}$ at each k that totals to K different ML models. Hence, cycle i concludes when all $\omega_{k,i}$ are aggregated to a signle set of weights $\omega_{g,i+1}$, that serve as the collective model for the next iteration. The two most prominent approaches to solve the FL problem are Fedavg [4] and Fedprox [20] and differentiate mainly in the local optimization problem $F_k()$ at each device.

Using stochastic gradient descent (SGD) as a local solver $F_k()$, federated averaging (FedAvg) locks the amount of local epochs for each device to a fixed value. As such, each learner is fixed on computing the same $F_k()$ with the same learning rate of SGD for the same amount of epochs. For the successful operation of this system, it is essential to tune the optimization hyperparameters properly including the amount of epochs. The tradeoff in FedAvg becomes one of computation and communication since computing more local epochs reduces communication overhead at the expense of diversifying the local objectives as each system converges to a local optima given their portion of the non-IID data.

Due to the expected heterogeneity in the network of learners in the proposed FL implementation, we use the FedProx algorithm. The benefit of FedProx is that it can converge and provide good general models even under partial work and very dissimilar amounts of $\tau_{k,i}$. This is done by introducing a proximal term $\|\omega - \omega_{g,i}\|$ that alleviates the negative impact of the heterogeneity as:

$$F_k(\omega; \omega_{g,i}) = L_k(\omega) + \frac{\mu}{2}\|\omega - \omega_{g,i}\|, \tag{E.2}$$

where ω is the instantaneous value of the local model weights at the local optimizer, $L_k(\omega)$ is a local cost function for the estimation losses, μ is a hyperparameter controlling the impact of the proximal term. The role of the proximal term here is that it prevents the local optimiser from straying far from the global model at round i. Moreover, we can control the local optimization problem to vary from a FedAvg ($\mu = 0$) to FedProx ($\mu > 0$). We note that even when using Fedprox, too much local work causes the local optimizers to diverge from the global objective [20]. Finally, using (E.2) for minimizing the local sub-problem $\min_\omega F_k(\omega; \omega_{g,i})$ the FL converges to a solution even in the presence of heterogeneity and non-IID data distribution [20]. Therefore, we use the FedProx algorithm to allow for full flexibility in data and processing heterogeneity, in combination with the resource allocation module that follows.

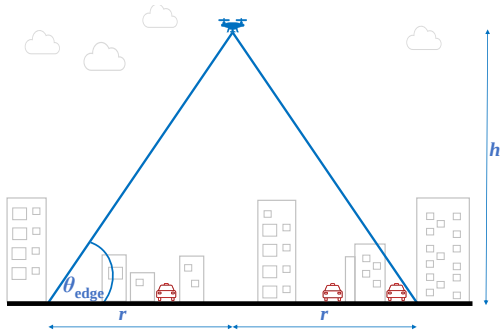


Fig. E.3: Illustration of the drone position and geometry, in the communications setting.

2.2 Allocation of Wireless Resources

Though the work of [17] covers a detailed cellular model for FL connectivity, drone provided connectivity is generally uniform and can be designed to be predominantly line of sight [34]. As we illustrate in Fig. E.3 the drone height h and the projected coverage on the ground with radius r impact the elevation angle at the edge of the MA, θ_{edge}. The steepness of the elevation can be derived from the evironmental parameters while also accounting for the directivity of the antenna mounted on the drone, as in [35], and the service reliability that needs to be achieved [36].

Since our goal of a DMT implementation is to improve the worst case performance of autonomous traffic, we also model the communications system through θ_{edge} as a worst case design parameter. θ_{edge} is decided upon deployment as it plays an important role of controlling the likelihood of establishing line of sight with the ground vehicles at the edge of the cell as in:

$$\mathrm{P}_{\mathrm{DLoS}} = \frac{1}{1 + a\exp(-b(\theta_{\text{edge}} - a))}, \tag{E.3}$$

where a and b are constants defined by the propagation topology of the environment, as given by [37]. Through θ_{edge} in (E.3) a system designer controls not only the probability of detecting a CO but also the average quality of the communications channel at the edge of the MA as:

$$\Lambda = L_{\mathrm{LoS}} \cdot \mathrm{P}_{\mathrm{DLoS}} + L_{\mathrm{NLoS}} \cdot (1 - \mathrm{P}_{\mathrm{DLoS}}), \tag{E.4}$$

where L_{LoS} and L_{NLoS} are the pathloss coefficients when LOS is established or lost, respectively. As such, we arrive to the average rate for the user located at the edge of the cell by:

$$R_{\text{avg}} = \log_2(1 + \frac{P_{\text{tx}}}{N\Lambda}), \tag{E.5}$$

where P_{tx} is the transmission power, and N is the noise power. As FL model transmissions usually take several seconds depending on the size of the model, we omit small scale fading as an impactful factor in the analysis and assume that the drone provided links are symmetrical in both directions and offer each learner k a rate of $\frac{W}{K} \cdot R_{\text{avg}}$, where W is the total bandwidth dedicated for the FL model passing. W may be represented as discrete resource blocks or a band of spectrum that is left over after portioning part of it for the purpose of video broadcasting. Like this, R_{avg} acts as a lower bound guarantee for the amount of time spent learning at each ground device.

As the size of the processing batch is fixed to B, each device k is tasked with an equal number of floating point operations (FLO) for each epoch, and computes $\tau_{k,i}$ epochs. However, for each learner k we introduce a coefficient f_k that represents the learners' computational power with regards to the model size, and is a unit of amount of epochs computed per unit time. Having full information on f_k is generally trivial since it depends on the processing capabilities of the learner, which should be publicly available in the device specifications.

Given an equal bandwidth allocation to all devices, the total number of epochs is a linear function of f_k. This results in the following equation for $\tau_{k,i}$:

$$\frac{\tau_{k,i}}{f_k} = T - \frac{KD}{WR_{\text{avg}}}, \tag{E.6}$$

where, D is the total amount of data that needs to be sent in both directions within the deadline of T. We convert the problem to a step-wise nomenclature that gives the relationship between each learner, independent of the length of T but as a relative inter-learner metric:

$$\begin{aligned} \tau_{k,i} - \tau_{l,i} &= Tf_k - \frac{KDf_k}{WR_{\text{avg}}} - Tf_l + \frac{KDf_l}{WR_{\text{avg}}}, \\ \tau_{k,i} - \tau_{l,i} &= T(f_k - f_l) - (f_k - f_l)\frac{KD}{WR_{\text{avg}}}, \\ \frac{\tau_{k,i} - \tau_{l,i}}{f_k - f_l} &= T - \frac{KD}{WR_{\text{avg}}}, \end{aligned} \tag{E.7}$$

where $\forall k, l \in \mathcal{K}, l \neq k$. We then perform the substitution:

$$\begin{aligned} \alpha &= \frac{\tau_{k,i} - \tau_{l,i}}{f_k - f_l}, \ \forall k, l \in \mathcal{K}, l \neq k, \\ T &= \alpha + \frac{KD}{WR_{\text{avg}}}, \end{aligned} \tag{E.8}$$

where α is the nominal time reserved for learning, and it is directly influenced by the

amount of FLOPs required to compute one epoch. This simplifies to:

$$\begin{aligned}\frac{\tau_{k,i}}{f_k} &= \alpha + \frac{KD}{WR_{\text{avg}}} - \frac{KD}{S_{k,i}WR_{\text{avg}}},\\ \frac{\tau_{k,i}}{f_k} &= \alpha + \frac{KD}{WR_{\text{avg}}}(\frac{S_{k,i}-1}{S_{k,i}}),\end{aligned} \tag{E.9}$$

where $S_{k,i} \geq 0$ and $\sum_k^K S_{k,i} = K$ is the bandwidth allocation for learner k in round i, represented as the portion of the average spectrum $\frac{W}{K}$ occupied (i.e. $S_{k,i} = K$ is the full spectrum, and $S_{k,i} = 1$ is the average spectrum). We continue with the substitution:

$$\frac{KD}{WR_{\text{avg}}} = \beta, \tag{E.10}$$

where β is the portion of time spent transmitting within one round. As per β, it is obvious that it is much more important to investigate the ratio of data load on the channel instead of solely focusing on the achieved rate R_{avg}. Moreover, the time spent learning at each device becomes more significant the more we load the resources, in both number of learners and the size of the model. This results in the final representation of epochs computed for learner k as a function of the bandwidth allocated to them:

$$\tau_{k,i} = f_k\alpha + f_k\beta(\frac{S_{k,i}-1}{S_{k,i}}), \tag{E.11}$$

Given a no-drop policy (each learner must complete at least one epoch $\tau \geq 1$), the lower bound on $S_{k,i}$ becomes:

$$S_{\min} = -\frac{\beta f_k}{1-\alpha f_k - \beta f_k}, \tag{E.12}$$

and the extreme upper bound of $S_{k,i}$ is therefore:

$$S_{\max} = K + \sum_{l}^{K-1} \frac{\beta f_l}{1-\alpha f_l - \beta f_l},\ \forall l \in \mathcal{K}, l \neq k. \tag{E.13}$$

The behaviour of the resource function for a single $\tau_{k,i}$ when adjusting β and $S_{k,i}$ within the bounds of (E.12) and (E.13), is:

$$S_{\min} \leq S_{k,i} \leq S_{\max}, \tag{E.14}$$

The entire communications setup is reducible to the analysis of combinations of α and β, as both parameters directly determine the impact that resource allocation has on the system. Moreover, the parameter β modifies the impact of resource allocation for each learner, where systems with high β values stand to benefit the most, while low β values indicate near instantaneous model transfers which cannot be influenced by modifying the bandwidth. On the other hand, α is a system design hyperparameter that indicates the amount of epochs computed within a single round, by an average learner, and it is fully customizable before or even during operation.

3 Analysis

Our goal is to improve the learning of a particular class among the network of FL devices, that may represent a CO, without harming the overall accuracy of the system. Thus, each round i we exploit our control over the wireless resources and optimize the bandwidth allocated to each device $S_{k,i}$. The vector representation of the bandwidth allocation for each round becomes $\mathbf{S}_i = (S_{1,i}, S_{2,i}...S_{K,i})$. In the same way, the number of epochs computed in round i and the contribution estimations are reformulated into vectors: $\mathbf{T}_i = (\tau_{1,i}, \tau_{2,i}...\tau_{K,i})$ and $\mathbf{G}_i = (G_{1,i}, G_{2,i}...G_{K,i})$ respectively, where $G_{k,i}$ is an estimate of the contribution of learner k based of its learning performance in the past. Due to the rapidly changing environment around each learner, we cannot assume having information about the size or distribution of the data stored at each learner. Therefore, we can assume a function of utility from both aforementioned parameters $h_{\mathrm{X}}(\tau_i, \mathbf{G}_i)$, where X is a placeholder for the name of the approach. Given this function, the optimization problem of maximizing the utility X can be defined as:

$$\begin{aligned} \max_{\mathbf{S}_i} \; & h_{\mathrm{X}}(\mathbf{T}_i, \mathbf{G}_i), \\ & \sum_{k}^{K} S_{k,i} = K, \\ & \tau_{k,i} \in \mathbb{Z}^+, \\ & \text{(E.11)}, \text{(E.12)}, \text{(E.13)}, \text{(E.14)}. \end{aligned} \tag{E.15}$$

Extracting the direct impact of $G_{k,i}$ and τ_i onto the future accuracy of the model, and under non-IID data distribution, is non-trivial and hence requires that we form several heuristic functions for $h_{\mathrm{X}}()$ to be tested on an experimental setup. Therefore we compare three different solutions for (E.15) by swapping the utility function $h_{\mathrm{X}}()$ with the ones named as $X \in \{\mathrm{MAX}, \mathrm{AAS}, \mathrm{ACT}\}$. The first two versions of the optimization problem (MAX and AAS) apply a static method that computes utility only as a function of the epochs that will be computed for that round for each learner. The third approach (ACT) is a novel reactive method, that extracts the utility of a learning round as a product of the estimated contribution by each learner and the epochs that will be computed by that learner. The details for each method follow below.

3.1 Static Resource Allocation Measures

The naive way of improving the convergence in a heterogeneous setting is maximizing the total amount of work done by all learners as in:

$$h_{\mathrm{MAX}}(\mathbf{T}_i, 0) = \sum_{k}^{K} \tau_{k,i}. \tag{E.16}$$

This optimization criteria maximizes the epochs computed across the whole network given the limited radio resources. Since (E.16) implies asyncronous amount of work performed among the learners, it may not be considered as a potential maximization metric when using classical FedAvg implementations. However, since we use FedProx as a local optimizer, this is a sufficient naive solution that represents an exploitative behavior from the orchestrator.

Furthermore, given the work on asynchronous FL and the issues of diverse computational hardware in the network [38, 39] we identify maximum *staleness* [16] as an important criterion towards the precision of the model. We define this as the maximal difference between the fastest and slowest learner:

$$s = \max(|\tau_{k,i} - \tau_{l,i}|) \ \forall k, l \in \mathcal{K}, l \neq k. \tag{E.17}$$

Nonetheless, minimizing staleness does not extract the full potential of our setup. Therefore, as in [16] we convene s and the average of the anticipated epochs to a more balanced heuristic metric, named Average Anchored Staleness (AAS) as an optimization metric:

$$h_{\text{AAS}}(\mathbf{T}_i, 0) = \frac{1}{K} \sum_{k}^{K} \tau_{k,i} - s. \tag{E.18}$$

AAS gives a good general overview that is data-agnostic, without the need to assume the impact of data at some particular learner and solely on spatial and computational performance. Like this, AAS provides a resource allocation objective function that serves an equally balanced amount of learning and *staleness*.

3.2 Contribution Estimation for Reactive Resource Allocation

In the case of DTMs, the considered vehicle supervisors/learners can find themselves in the presence of vastly different objects, and the data they sense changes constantly while they operate. Given the aforementioned, the contribution of each learner is hard to estimate especially in the presence of noisy samples. Hence, we assume that separating the important CO information ahead of time is impossible and only consider reactive approaches such as incentive mechanisms. To use incentive mechanisms we must assume that the validation dataset that is present at the orchestrator has equal representation of all classes. Hence, based on such validation data we can pass the weights ω through an evaluation function $E(\omega)$ which can be based on accuracy or loss evaluations of the model **(we choose accuracy)**. To calculate the contribution for each round i we define:

$$G_{k,i} = \frac{E(\omega_{\text{g},i}) - E(\omega_{\text{g}\setminus\{k\},i})}{\sum_{k}^{K} |E(\omega_{\text{g},i}) - E(\omega_{\text{g}\setminus\{k\},i})|}, \tag{E.19}$$

where $\omega_{g\setminus\{k\},i}$ is a model aggregator that constructs a new model that is an aggregate of all recieved models except the one of k. Hence the difference in accuracy between the fully aggregated model and the $\omega_{g\setminus\{k\},i}$ [27] gives the added value (the uniqueness) of the learning done by learner k. Like this, the contribution estimator is capable of discovering the overall contribution from each learner for that round, without the capability of sampling for contributions on each detection class separately, or discern which object is underrepresented or is the CO. This is a central feature of our method, since we aim to improve CO learning without tailoring the solution to discern which class is the CO.

We note that the $\omega_{g\setminus\{k\},i}$ function needs to be called for each learner in order to produce K different contribution estimations. In addition to having to compute an additional parameter, there is one extra set of weights that needs to be aggregated for the calculation of $\omega_{g\setminus\{k\},i}$ for all other learners, thus making the complexity of the estimator scale as a square of the number of nodes in the system K. Even though the computational complexity of this technique can escalate in big FL implementations, in the architecture that we propose there should be several active learners inside the MA. Thus, even aside the limited computational power on the drone, the estimator module should not experience lengthy computational times.

Following the first round, each device k provides its model to the DTM-orchestrator. After which, the aggregator provides the first aggregate model weights $\omega_{g,i}$. The resource allocator module in the orchestrator receives the contributions for each of the participating learners and hence can decide to adjust the resources based on $G_{k,i}$. Since $G_{k,i}$ is an estimation of the contributions for the past round, the goal is to maximize the total contribution of the upcoming round by introducing the following optimization function:

$$h_{\mathrm{ACT}}(\mathbf{T}_i, \mathbf{G}_i) = \sum_{k}^{K} \tau_{k,i} g(G_{k,i}), \tag{E.20}$$

where $g()$ is a utility function that scales the contributions to match the impact of the number of computed epochs. Introducing a utility function is necessary to properly scale each learner's impact since $-1 \leq G_{k,i} \leq 1$ and $\tau_{k,i} \in \mathbb{Z}^+$. Since in an average scenario $\mathbb{E}[\tau_{k,i}] = \alpha\mathbb{E}[f_k]$, and $\mathbb{E}[f_k] = 1$ we scale our utility function as per the average epochs computed for that round as $g(G_{k,i}) = \alpha^{G_{k,i}}$. The bounds of the function become $\frac{1}{\alpha} \leq g(G_{k,i}) \leq \alpha$, and the nominal non-contributive learners produce $g(0) = 1$. Thus the heuristic exponential optimization function for the reactive solution can be calculated as the contribution corrected maximum epochs computed as in:

$$h_{\mathrm{ACT}}(\mathbf{T}_i, \mathbf{G}_i) = \sum_{k}^{K} \tau_{k,i} \alpha^{G_{k,i}}. \tag{E.21}$$

In the case of constantly equal contributions from all learners, the heuristic maximization criteria is reduced to the epoch maximization problem defined in (E.16). With h_{ACT}

Table E.2: The non-IID distribution of data among learners, and their computational coefficients f_k.

Learner	Classes Stored (out of 0-9)	f_k
1	3, 4, 5, 6	0.15
2	0, 1, 2, 3, 4	0.7
3	4, 6, 7, 8, 9	1.0
4	0, 1, 2, 6, 7, 8, 9	1.3
5	0, 1, 2, 7, 8, 9	1.3
6	0, 1, 2, 7, 8, 9	1.0
7	3, 4, 6	0.7

defined as in (E.21) we maintain the problem within the bounds of mixed integer linear programming since the utility is applied only to $G_{k,i}$ that remains constant for the whole round i.

4 Results

4.1 Experimental Setup

For a set of learners that are scattered along the MA, our goal is to as closely as possible generate an experimental setup that simulates a realistic learner given the system model in Section 2. Since each learner has a very short amount of time to do the learning for the DTM, we approach the data as fleeting (stored very briefly) and concealed (cannot be known beforehand). Due to the complexity and the issues of reliably simulating the FL performance for full scale traffic footage, we test the performance of the proposed methods through simple and easily accessible computer vision datasets. Each testing scenario was built using either the MNIST dataset [40] of handwritten digits, or the FMNIST [41] dataset consisting of 10 different grayscale icons of fashion accessories.

As we expect that each vehicle contains strongly non-IID data we create a custom data distribution among $K = 7$ learners as shown in Table E.2. In addition, the processing power for computing a certain amount of epochs per millisecond f_k for each learner, is distributed as: two standard vehicles ($f_k = 1$), two premium vehicles ($f_k = 1.3$), and two budget vehicles ($f_k = 0.7$); with the addition of one straggler that contains an older technology ($f_k = 0.15$). At each epoch the learner samples a single batch of $B = 16$ randomly selected values from the stored data (as per Table E.2). Like this, the training data changes constantly, to mimic the changing environment of the vehicular scenario. This makes this FL testing scenario unique in that the number of epochs computed also reflects the amount of data sampled from the environment.

In the described setting, the class-number 5 (6th class counting from zero) assumes

the role of a CO. In addition to the CO, class-number 3 is another non-CO class that is not too common and appears at only 3 learners. This is an over-exaggerated situation of having the CO data hidden at one node that is also a straggler. We expect this to be a realistic reflection of data in drone orchestrated FLs as nodes carry only a small amount of supervisory data for each class due to the fact that they stumble upon important objects randomly.

For detection, we implement a small convolutional neural network (CNN), common for the global and local models implemented in python tensorflow [42]. In more detail, the CNN has only one 3x3 layer of 64 channels using the rectifier linear unit (ReLU), that goes to a 2x2 polling layer. A dense, fully connected neural network (NN) layer of 64 ReLU activated neurons receives the polled outputs of the convolutional layer, which is then fully connected to a NN layer of 10 soft-max activated neurons, one for each of the 10 categories of the NIST dataset. The local optimizer at each learner is given by the FedProx calculation in Eq. (E.2), where the cost function $L_k()$ is a categorical cross-entropy loss function, and the learning rate performed well when fixed to $\gamma = 0.1$. The communication phase coefficient was considered in milliseconds and chosen as $\beta = 100$ considering our CNN model with a size of 2.5Mb that needs to be transmitted to all 7 learners, over a single $W = 80$MHz 802.11ax channel. Finally, in the reference frame of milliseconds, the cycle duration coefficient was set to $\alpha = 100$ in favor of allowing for higher flexibility when scaling the bandwidth allocation.

4.2 MNIST Testing

We proceed with the testing of all three approaches for five different values of the proximal importance hyperparameter $\mu \in \{0, 0.01, 0.1, 0.5\}$, as guided by the recommended values in [20]. μ values larger than 0.5 failed to produce productive results and only harmed the convergence outlook. The testing lasts for 200 rounds on the aforementioned CNN model. Aside the three shown FL implementations, we also implement a classical ML with only one learner that contains all the data. We do this to extract the performance ceiling of the NN approach, which is 98% for the validation accuracy and 0.0602 validation loss paired with training accuracy of 98.85% and training loss 0.0423.

In Fig. E.4 we can notice a limited impact of changing the μ parameter of FedProx, most likely due to the small amount of learners and not as significant straggler impact. This is expected given that [20] claim strong superiority over FedAvg in the cases of very large portions of stragglers. Interestingly, μ does not have a strong positive impact on the learning performance even in the case of MAX, and therefore, a system designer would most likely introduce a weak proximal term of $\mu = 0.01$. Additionally, using the ACT approach provides superior convergence, and in combination with $\mu = 0.01$ achieves the best overall accuracy. In addition to this, the ACT and $\mu = 0.01$ combination also

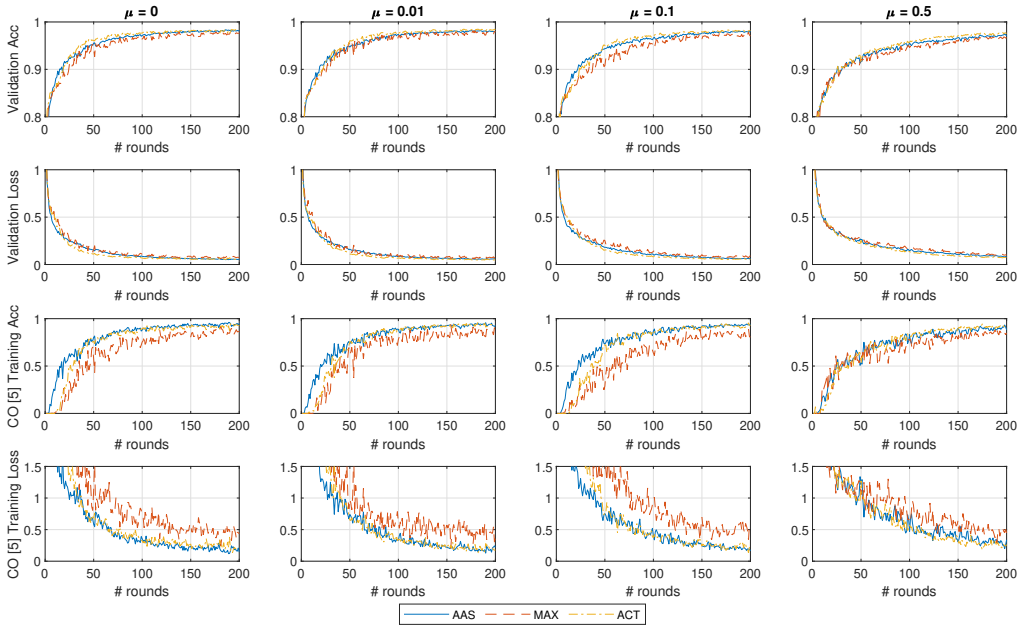


Fig. E.4: General and CO-specific accuracy and loss results obtained when testing all three methods in combination with FedProx using the MNIST dataset.

keeps up with the performance of AAS with regards to the CO class after the first several rounds of convergence.

To better investigate the behavior of the ACT approach we illustrate the evolution of the estimated contributions for learner $k = 1$ in Fig. E.5, where $G_{1,i}$ is based on the performance of the learner estimated from the previous learning round as in Eq. (E.19). The overall conclusion here is that we achieve CO learning without tailoring the solution to discern which class is the CO. This is possible as the calculation of $G_{k,i}$ is focused around the uniqueness of the dataset at each learner. Here we can notice that increasing the strength of the proximal parameter through setting higher μ values equalizes the contributions between all three methods, particularly in the first 40 rounds. Moreover, when $\mu = 0.5$ the contributions are stabilized and vary very little once the initial phase of 40 rounds.

Most notably, the accuracy of AAS suffers significantly when $\mu = 0.5$ which results in a performance that is equally matched to the MAX approach when detecting the CO. It is thus evident that a strong FedProx implementation harms total system accuracy, and above all, diminishes the impact of the using resource allocation. Finally, we conclude that the task of learning MNIST is too simplistic for our assumed scenario of traffic monitoring, and thus we continue with testing the FMNIST dataset in the following

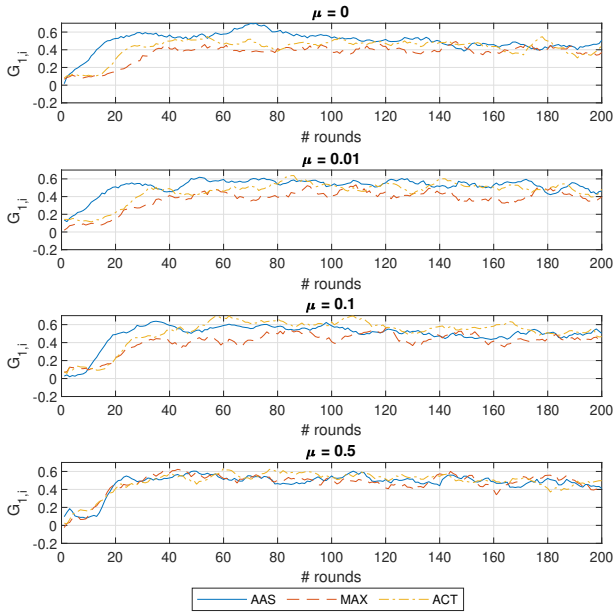


Fig. E.5: Contribution evolution for learner $k = 1$ in the case of using the MNIST dataset.

subsection.

4.3 FMNIST Testing

Since modeling common tasks of computer vision on MNIST is a very easy task we repeat the test on the FMNIST dataset. This dataset consists of 10 classes of fashion accessories in equal distribution as the MNIST dataset (a training set of 60 000 examples and a test set of 10 000 examples) and as in the case of MNIST consists of 28x28 grayscale images. The dataset classes are: (0) T-shirt/top, (1) Trouser, (2) Pullover, (3) Dress, (4) Coat, (5) Sandal, (6) Shirt, (7) Sneaker, (8) Bag, and (9) Ankle boot; where each item is taken from a fashion article posted on Zalando. Compared to the number MNIST, in FMNIST the intensity of each voxel plays a much bigger role and is scattered across larger parts of the image. We consider the FMNIST dataset as a computer vision task that sufficiently replicates the problem of detecting 10 different types of vehicles, in a much more simplistic context that is furthermore easily replicable.

In Fig. E.6, we show the learning performance in the same setting and $\mu \in \{0, 0.01, 0.1, 0.5\}$, across 200 rounds of training. It is most obvious that the overall accuracy has dropped quite a lot from the 98% in the MNIST case to 88% in the best

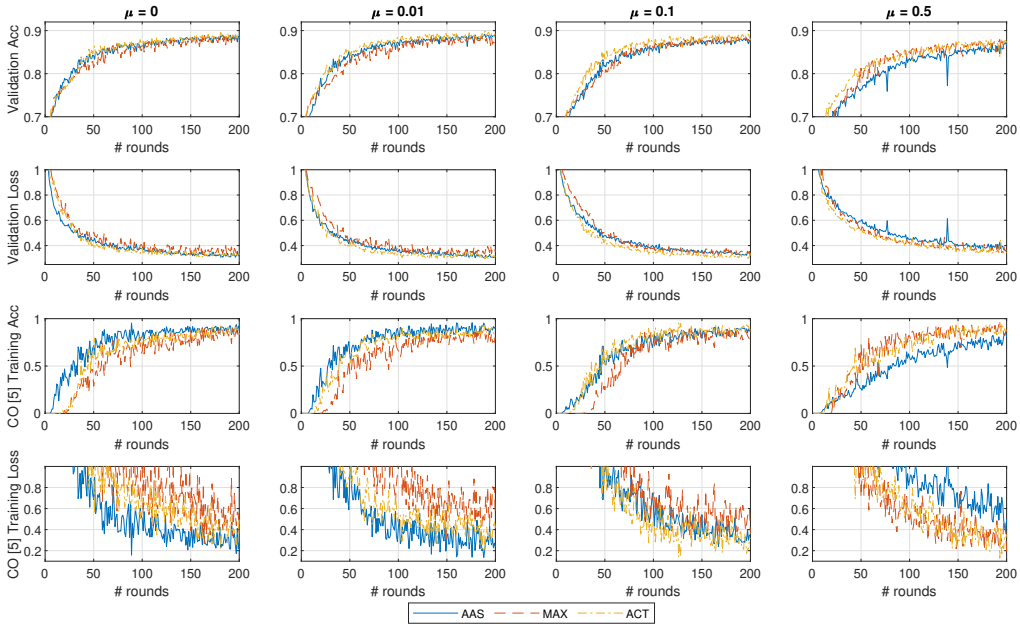


Fig. E.6: General and CO-specific accuracy and loss results obtained when testing all three methods in combination with FedProx using the FMNIST dataset.

case scenario of ACT with $\mu = 0.01$ for the FMNIST. Most notably the largest difference is that the increased difficulty of the learning problem introduces a lot more noise in the learning process, particularly for the CO class. Due to this, when using no FedProx ($\mu = 0$) AAS does a good job at accelerating the learning process in the first 20 rounds until it is overtaken by ACT. Even though the combination of ACT with $\mu = 0.01$ shows the best overall accuracy on the validation data, the accuracy of detecting the CO class with ACT never truly reaches the performance of AAS.

Finally, we conclude that even though $\mu = 0.1$ and $\mu = 0.5$ were eligible in the MNIST run, the overall increased complexity of FMNIST harms the accuracy outlook in both, but with the most severe impact on AAS. This experimental run therefore inspired us to investigate the issue of underfitting, and we proceed with testing FMNIST performance with a deeper model.

4.4 Deeper FMNIST Testing

In this testing scenario we expand the small convolutional neural network by adding another 3x3 layer of 64 channels using ReLU activators as a first layer. In Fig. E.7

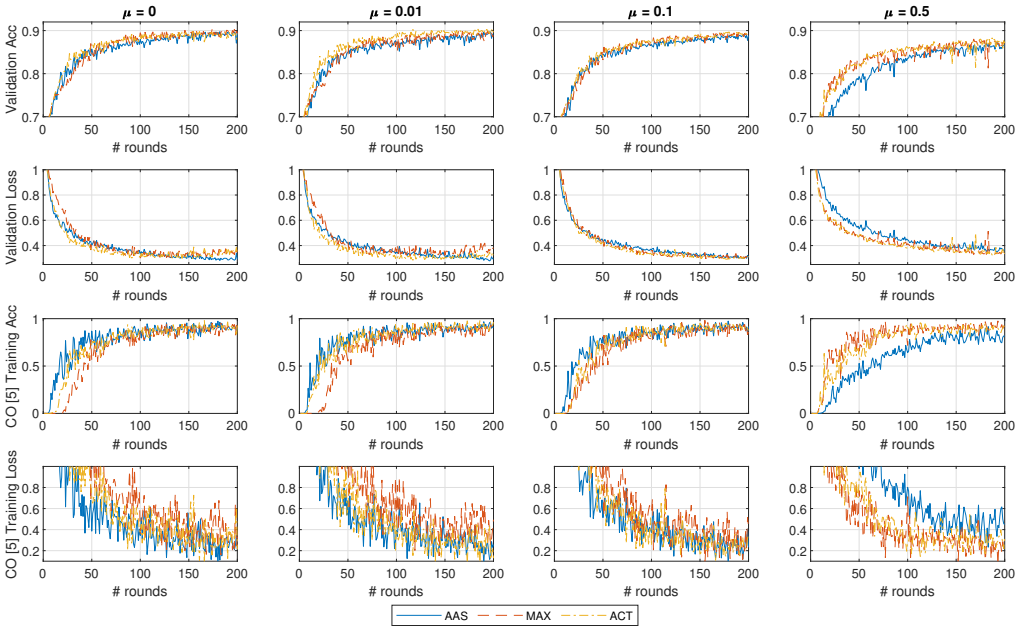


Fig. E.7: General and CO-specific accuracy and loss results obtained when testing, with an extra CNN layer, all three methods in combination with FedProx using the FMNIST dataset.

we show the outcomes of the testing, where the overall accuracy of the system has been improved to 90%. However, the larger model acted as an equalizer across all three approaches and in the case of $\mu = 0$ generally gave equal performance both in convergence time and overall accuracy. It is important to also look at the validation loss following the round $i = 150$ as it starts to diverge for both ACT and MAX approaches. This did not directly map into the accuracy of the detection, but nonetheless is a first sign of possible overfitting and eventual divergence.

With the deep model, this effect is diminished for the case of ACT with $\mu = 0.01$, and manages to reach the best convergence time along with overall accuracy from all tested implementations. This accuracy is also paired with improved detection of the CO that exactly matches the AAS approach. As such the ACT with $\mu = 0.01$ is both the best overall learning solution, but also the best CO detector.

It is also interesting to notice that the MAX approach does well with overall accuracy, particularly when compared to the inferior performance in the previous testing sets. Nonetheless, MAX is still inferior to both other approaches when it comes to detecting the CO class. Finally, we focus on the results on $\mu = 0.5$. When the proximal term has such a strong impact on the learning, all three approaches show inferior overall performance by 4-5 percentage points with regards to the best performing $\mu = 0.01$.

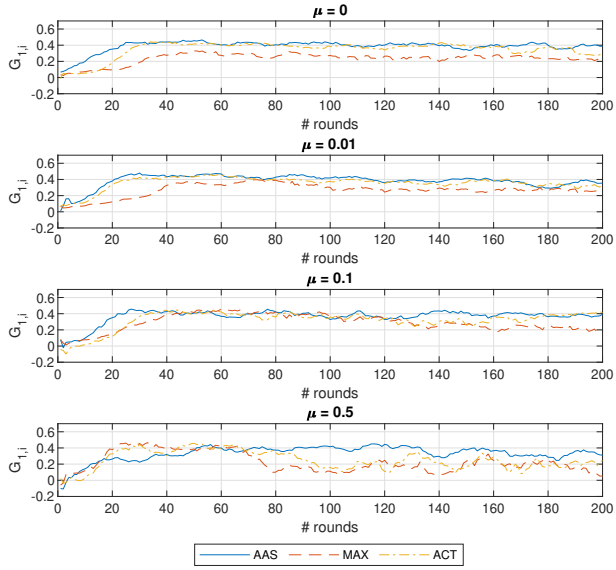


Fig. E.8: Contribution evolution for learner $k = 1$ in the case of using the FMNIST dataset with an extra CNN layer.

However, it is interesting to see that the impact is by far most severe on the AAS approach, even reducing the CO detection performance. Additionally, MAX gives the best result when it comes to learning the CO behavior for $\mu = 0.5$. Opposed to the behavior back in the MNIST testing, here AAS suffers from the increased complexity of the task, and in combination with a very strong proximal term reduces the overall learning of detection. This makes it is easy to conclude that a strong proximal term reduces the effect of resource allocation efforts.

We seek to discover the culprit for the inferiority of AAS in CO discovery when $\mu = 0.5$ by plotting the contributions of learner $k = 1$ in Fig. E.8. Looking at the contribution evolution in case $\mu = 0.5$ we extrapolate that AAS aims to keep the learner relevant while the reduced amount of learning across the whole network harms the potential contribution of all other nodes. This leads us to the final conclusion of this experiment which is that the ACT based approach is extremely versatile in providing good CO detection and accuracy even in the cases of $\mu = 0$, a properly assigned μ, and overly restricted FedProx implementation.

4.5 Testing Fleeting FMNIST

The final test with the experimental setup is constructed such that we introduce stress in the learning process by introducing temporary losses in the supervision process. This is done by introducing a likelihood that a learner k loses access to a detection class. This would be representative of a learner losing LOS of the object was able to supervise, and is therefore modelled as a two state markov model (such as the Gilbert Elliot [43]) that has a good and a bad state. Hence each supervisor has $p = 0.9$ chance to maintain supervision for that class (stay in the good state), and $1 - p = 0.1$ probability to lose supervision capability (and move to the bad state). If the vehicle loses supervision capabilities for that class, it has $r = 0.5$ probability to maintain that state (remain in the bad state) or $1 - r = 0.5$ probability to regain supervision of that class. The values for the state transitions in the Gilbert-Elliot model were chosen with the experimental setup in mind so that not too much data is lost with regards to the previous testing setups. These testing parameters were provisioned arbitrarily, because higher values would make the learning process very lengthy imposing unrealistic testing times for our experiment, but still provide a lot of stress to the learning system.

Hence, to compensate for the smaller dataset, we let the simulations run for 250 rounds, and focus only on $\mu \in \{0.01, 0.1\}$. The fleeting data is provided from the same seed and the Gilbert Elliot model starts from the good state for every possible detection combination. In Fig. E.9 we show the performance of all approaches on the aforementioned setup. Comparing this to the previous testing setup, we notice that the overall accuracy dropped by 1 percentage point for $\mu = 0.01$ and 2 percentage points when $\mu = 0.1$ due to the increased stress in the learning process. It is also apparent that both ACT and MAX show signs of overfitting – the diverging lines in the validation loss – which is improved when using $\mu = 0.1$, at the cost of reducing the overall system accuracy by an additional 1 percentage point.

Focusing on $\mu = 0.01$, all methods achieve nearly the same overall accuracy, since the learning of the computer vision task is bottlenecked by the presence of the data. However, AAS is superior in CO detection and it shows slightly inferior convergence time for overall accuracy (i.e. around the 50 round mark). In addition to this, AAS is the most data sensitive approach and experiences the largest overall accuracy dips in situations where many detection classes are in the bad state (such as around the 55th round and the 127th round). Finally, to better observe the noisy training data, we plot a 10-point moving average in Fig. E.10. Here we notice the in the common training scenario AAS and ACT perform rather equally when learning hidden information. However, in the presence of fleeting data, the ACT performance becomes very noisy and becomse slightly inferior than AAS with regards to CO learning performance. Nonetheless, as already mentioned, this CO learning performance of the AAS approach comes at a slight cost of general detection performance, in both fleeting and normal setting.

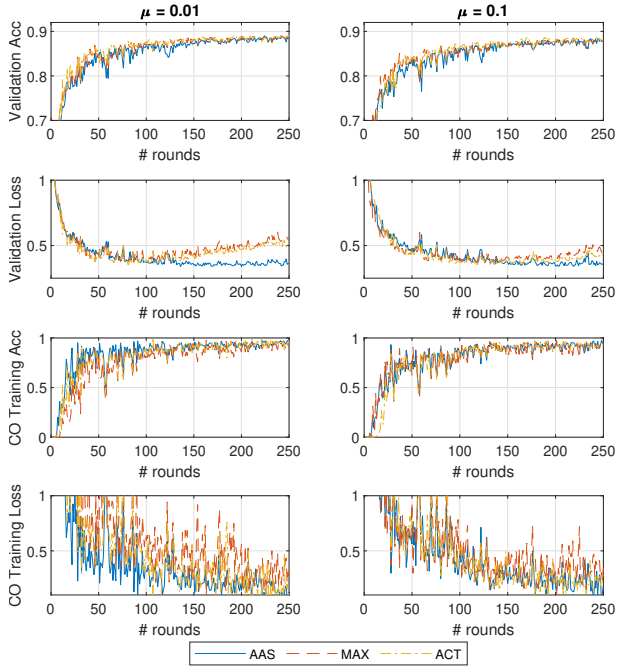


Fig. E.9: General and CO-specific accuracy and loss results obtained when testing, with an extra CNN layer, all three methods in combination with FedProx using the FMNIST in the case of fleeting data.

4.6 Key Takeaways

We condense several takeaways that were derived from all four experimental runs. The initial and most important conclusion is that the concepts of resource allocation and FedProx are at odds in the case of FL implementations. In more detail, the goal of FedProx is to reduce the impact of each learner individually while resource allocation methods strive to improve the overall performance by exploiting or compensating the heterogeneity of the system. Hence the impact of resource allocation methods is diminished when strengthening the role of the proximal term. Nonetheless, in the many tests a safe balance between both μ and resource allocation ensure good learning behavior. As such, we recommend that all future works consider perturbed gradient descent implementations, such as FedProx, when dealing with non-IID data in heterogeneous FL.

Additionally, in the initial testing of our setup we noticed that testing on MNIST is not sufficient to provide reasonable results for the implementations, due to how trivial the task of recognising digits is. Moreover, FL implementations, such as the proposed

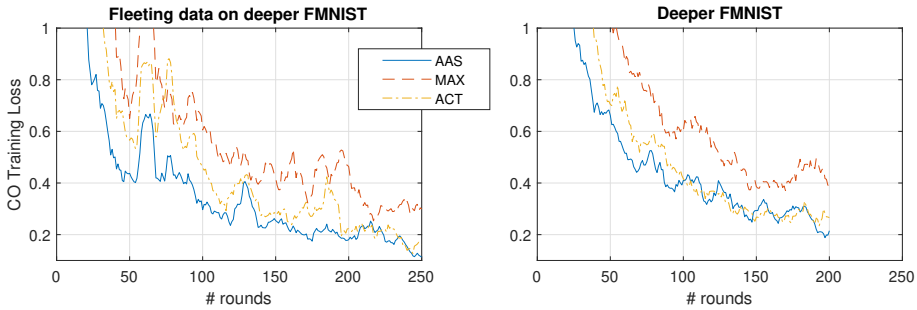


Fig. E.10: 10-point moving average of CO training loss for $\mu = 0.01$ of the fleeting data versus normal data sampling in the deeper FMNIST testcase.

drone implementation, are based in the distributed learning of complex tasks and require deeper NN models. In such cases, it was evident that increasing the total amount of computed epochs benefits the convergence time of the system with potentially harmful effects in CO detection accuracy. Moreover, deeper model implementations did not behave well under strong proximal terms.

As a consequence to this, learning hidden data can be addressed by equalizing the contributions by using AAS or by introducing strong proximal terms. However, the strong proximal terms have potential to slow down the convergence time for all nodes. Hence, the safest implementation to achieving the best combination of convergence time, overall accuracy and CO learning rate is using the ACT approach with a weak proximal term.

Finally, in a case where the data is fleeting, using a $\mu > 0$ was crucial to reach stable learning performance. In this setting, the low availability of data acted as a lower bound for all learning implementations, but most importantly harms the convergence time performance of AAS. This is understandable since AAS was the approach that cumulatively computed the least amount of epochs at each round. On the other hand, the ACT approach maintained superior performance to both static approaches by maintaining good CO detection performance and great convergence times.

Finally, we extrapolate that defining a proper μ is cardinal. However, the hyperparameter needs to be defined ahead of the deployment of the system. As such, since we would not have access to the training data, the feasibility of implementing AAS is uncertain especially for situations where the presence of data changes quickly. This gives another strong motivation for using reactive measures based on contributions and incentive calculations, such as ACT.

5 Conclusion

In this paper we investigated the learning process in a novel Federated Learning (FL) architecture, where a DTM acts as an orchestrator and traffic participants act as supervisors on its model. Such an implementation expects impairments on the learning process due to unbalanced and non-IID data scattered across heterogeneous learners that have variable computational equipment. We therefore test the ability of two static methods (AAS and MAX), and one incentive based reactive (ACT) resource allocation method to improve the speed of learning CO classes and maintaining good overall model accuracy. The validity of the methods was tested with an experimental FL implementation that uses the novel FedProx algorithm to learn from the MNIST and FMNIST datasets. The testing was conduced across combinations of different FedProx strength, CNN model depth, and fleeting data. From the testing we conclude that both reactive (ACT) resource allocation and FedProx are essential to securing model accuracy. In more detail, due to the inability to anticipate the distribution of the data across the learners, the use of ACT ensures proper operation of the FL implementation. In accord, the combination of properly set FedProx with an ACT implementation provided faster convergence times, better accuracy, but most importantly it matched the AAS method in learning to recognize the CO. Such behavior was consistent across most runs given the varying task complexity, model size, and data presence. The goal of future works would be to look into more advanced proactive approaches, especially for the presence of imperfect data supervision.

Conflict of Interest Statement

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Author Contributions

ID: investigation, writing; JJN and PP: writing, review, editing, resources, funding acquisition, supervision, and project administration.

Funding

The work was supported by the European Union's research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 812991 "PAINLESS" within the Horizon 2020 Program.

References

[1] SAE, "Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles," *SAE International,(J3016)*, 2016.

[2] T. A. S. Nielsen and S. Haustein, "On sceptics and enthusiasts: What are the expectations towards self-driving cars?" *Transport policy*, vol. 66, pp. 49–55, Aug. 2018.

[3] I. Yaqoob, L. U. Khan, S. A. Kazmi, M. Imran, N. Guizani, and C. S. Hong, "Autonomous driving cars in smart cities: Recent advances, requirements, and challenges," *IEEE Network*, vol. 34, no. 1, pp. 174–181, Aug. 2019.

[4] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated Learning: Strategies for Improving Communication Efficiency," *arXiv preprint arXiv:1610.05492*, Oct. 2016.

[5] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50–60, May 2020.

[6] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Trans. Intell. Syst. Technol.*, vol. 10, no. 2, Jan. 2019. [Online]. Available: https://doi-org.zorac.aub.aau.dk/10.1145/3298981

[7] W. Y. B. Lim, N. C. Luong, D. T. Hoang, Y. Jiao, Y. C. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," *IEEE Communications Surveys Tutorials*, vol. 22, no. 3, pp. 2031–2063, Apr. 2020.

[8] M. Aledhari, R. Razzak, R. M. Parizi, and F. Saeed, "Federated learning: A survey on enabling technologies, protocols, and applications," *IEEE Access*, vol. 8, pp. 140 699–140 725, July 2020.

[9] S. Niknam, H. S. Dhillon, and J. H. Reed, "Federated learning for wireless communications: Motivation, opportunities and challenges," 2020.

[10] Z. Yang, M. Chen, K.-K. Wong, H. V. Poor, and S. Cui, "Federated learning for 6g: Applications, challenges, and opportunities," *arXiv preprint arXiv:2101.01338*, 2021.

[11] S. Savazzi, M. Nicoli, M. Bennis, S. Kianoush, and L. Barbieri, "Opportunities of federated learning in connected, cooperative and automated industrial systems," *arXiv preprint arXiv:2101.03367*, 2021.

[12] L. U. Khan, S. R. Pandey, N. H. Tran, W. Saad, Z. Han, M. N. H. Nguyen, and C. S. Hong, "Federated learning for edge networks: Resource optimization and incentive mechanism," *IEEE Communications Magazine*, vol. 58, no. 10, pp. 88–93, Nov. 2020.

[13] T. Zeng, O. Semiari, M. Mozaffari, M. Chen, W. Saad, and M. Bennis, "Federated Learning in the Sky: Joint Power Allocation and Scheduling with UAV Swarms," in *Proc. of the IEEE International Conference on Communications (ICC), Next-Generation Networking and Internet Symposium*, Dublin, Ireland, June. , 2020, pp. 1–6.

[14] H. Zhang and L. Hanzo, "Federated learning assisted multi-uav networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 11, pp. 14 104–14 109, Sept. 2020.

[15] Y. Wang, Z. Su, N. Zhang, and A. Benslimane, "Learning in the air: Secure federated learning for uav-assisted crowdsensing," *IEEE Transactions on Network Science and Engineering*, pp. 1–1, Aug. 2020.

[16] I. Donevski, N. Babu, J. J. Nielsen, P. Popovski, and W. Saad, "Federated learning with a drone orchestrator: Path planning for minimized staleness," *IEEE Open Journal of the Communications Society*, Apr. 2021.

[17] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A joint learning and communications framework for federated learning over wireless networks," *IEEE Transactions on Wireless Communications*, Oct. 2020.

[18] M. M. Amiri and D. Gündüz, "Federated learning over wireless fading channels," *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, pp. 3546–3557, Feb. 2020.

[19] N. H. Tran, W. Bao, A. Zomaya, M. N. H. Nguyen, and C. S. Hong, "Federated learning over wireless networks: Optimization model design and analysis," in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, June 2019, pp. 1387–1395.

[20] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *arXiv preprint arXiv:1812.06127*, Dec. , 2018.

[21] V. Smith, C.-K. Chiang, M. Sanjabi, and A. S. Talwalkar, "Federated multi-task learning," in *in Proc. of Advances in neural information processing systems*, 2017, pp. 4424–4434.

[22] T. Li, M. Sanjabi, A. Beirami, and V. Smith, "Fair resource allocation in federated learning," in *in Proc. of International Conference on Learning Representations*, 2019.

[23] M. Mohri, G. Sivek, and A. T. Suresh, "Agnostic federated learning," *arXiv preprint arXiv:1902.00146*, Feb. 2019.

[24] M. Chen, Y. Liu, W. Shen, Y. Shen, P. Tang, and Q. Yang, "Mechanism design for multi-party machine learning," 2020.

[25] S. R. Pandey, N. H. Tran, M. Bennis, Y. K. Tun, A. Manzoor, and C. S. Hong, "A crowdsourcing framework for on-device federated learning," *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, pp. 3241–3256, Feb. 2020.

[26] J. Kang, Z. Xiong, D. Niyato, H. Yu, Y. Liang, and D. I. Kim, "Incentive design for efficient federated learning in mobile networks: A contract theory approach," in *in Proc. of 2019 IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS)*, Sept. 2019, pp. 1–5.

[27] T. Nishio, R. Shinkuma, and N. B. Mandayam, "Estimation of individual device contributions for incentivizing federated learning," *arXiv preprint arXiv:2009.09371*, 2020.

[28] J. Kang, Z. Xiong, D. Niyato, S. Xie, and J. Zhang, "Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10 700–10 714, Sept 2019.

[29] W. Shi, H. Zhou, J. Li, W. Xu, N. Zhang, and X. Shen, "Drone assisted vehicular networks: Architecture, challenges and opportunities," *IEEE Network*, vol. 32, no. 3, pp. 130–137, Jan. 2018.

[30] C. She, C. Liu, T. Q. Quek, C. Yang, and Y. Li, "Ultra-reliable and low-latency communications in unmanned aerial vehicle communication systems," *IEEE Transactions on Communications*, vol. 67, no. 5, pp. 3768–3781, Jan. 2019.

[31] P. Popovski, J. J. Nielsen, C. Stefanovic, E. De Carvalho, E. Strom, K. F. Trillingsgaard, A.-S. Bana, D. M. Kim, R. Kotaba, J. Park *et al.*, "Wireless access for ultra-reliable low-latency communication: Principles and building blocks," *Ieee Network*, vol. 32, no. 2, pp. 16–23, Apr. 2018.

[32] M. Mozaffari, W. Saad, M. Bennis, Y. Nam, and M. Debbah, "A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems," *IEEE Communications Surveys Tutorials*, vol. 21, no. 3, pp. 2334–2360, Mar. , 2019.

[33] T. Chavdarova, P. Baqué, S. Bouquet, A. Maksai, C. Jose, T. Bagautdinov, L. Lettry, P. Fua, L. Van Gool, and F. Fleuret, "Wildtrack: A multi-camera hd dataset for dense unscripted pedestrian detection," in *in Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5030–5039.

[34] N. Babu, K. Ntougias, C. B. Papadias, and P. Popovski, "Energy Efficient Altitude Optimization of an Aerial Access Point," in *Proc. of IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications*, London, Sep. , 2020, pp. 1–7.

[35] I. Donevski and J. J. Nielsen, "Dynamic Standalone Drone-Mounted Small Cells," in *in Proc. of European Conference on Networks and Communications (EuCNC)*, Dubrovnik, Croatia, Sep. , 2020, pp. 342–347.

[36] I. Donevski, J. J. Nielsen, and P. Popovski, "Standalone Deployment of a Dynamic Drone Cell for Wireless Connectivity of Two Services," in *Proc. of IEEE Wireless Communications and Networking Conference (WCNC)*, no. TBP, Nanjing, China , Apr. , 2021.

[37] A. Al-Hourani, S. Kandeepan, and A. Jamalipour, "Modeling Air-to-Ground Path Loss for Low Altitude Platforms in Urban Environments," in *Proc. of IEEE Global Communications Conference*, Austin, TX, Oct. , 2014, pp. 2898–2904.

[38] C. Xie, S. Koyejo, and I. Gupta, "Asynchronous federated optimization," *arXiv preprint arXiv:1903.03934*, Mar. , 2019.

[39] U. Mohammad, S. Sorour, and M. Hefeida, "Task allocation for asynchronous mobile edge learning with delay and energy constraints," *arXiv preprint arXiv:2012.00143*, 2020.

[40] Y. LeCun, L. D. Jackel, L. Bottou, C. Cortes, J. S. Denker, H. Drucker, I. Guyon, U. A. Muller, E. Sackinger, P. Simard *et al.*, "Learning algorithms for classification: A comparison on handwritten digit recognition," *Neural networks: the statistical mechanics perspective*, vol. 261, no. 276, p. 2, 1995.

[41] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017.

[42] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu,

and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[43] M. Boban, X. Gong, and W. Xu, "Modeling the evolution of line-of-sight blockage for v2v channels," in *In Proc. 2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, 2016, pp. 1–7.

Paper F

Federated learning with a drone orchestrator: Path planning for minimized staleness

Igor Donevski, Nithin Babu, Jimmy Jessen Nielsen, Petar Popovski, Walid Saad

The paper has been published in the
IEEE Open Journal of the Communications Society Vol. 2, pp. 1000–1014, 2021.

The layout has been revised.

Abstract

In this paper, we investigate the problem of scheduling transmissions for spatially scattered nodes that contribute to a collaborative federated learning (FL) algorithm via wireless links provided by a drone. In the considered system, the drone acts as an orchestrator, coordinating the transmissions and the learning schedule within a predefined deadline. The actual schedule is reflected in a planned path: as the drone traverses it, it controls the distance and thereby the data rate to each node. Hence, the model is structured such that the drone orchestrator uses the path (trajectory) as its only tool to achieve fairness in terms of learning staleness*, which reflects the learning time discrepancy among the nodes. Using the number of learning epochs performed at each learner as a performance indicator, we combine the average number of epochs computed and staleness into a balanced optimization criterion that is agnostic to the underlying FL implementation. We consider two methods for solving the complex trajectory planning optimization problem for static nodes: (1) successive convex programming (SCP) and (2) deep reinforcement learning (RL). Considering the proposed criterion, both methods are compared in three specific scenarios with few nodes. The results show that drone-orchestrated FL outperforms an immobile deployment by providing improvements in the range of 57% to 87.7%. Additionally, RL-guided trajectories are generally superior to SCP provided ones for complex node arrangements.*

1 Introduction

Initially meant for military uses, then followed by a boom in commercial entertainment usage, the flying drones or unmanned aerial vehicles (UAVs) have a growing importance in the world of communications. The use of drones for wireless communication purposes has also received a surge of attention [1, 2] due to their excellent coverage to outdoor users. In particular, due to their flexibility, low altitude platform (LAP) drones are useful to act as on-demand drone small cells (DSCs) for wireless communication support. DSCs have the potential to continuously relocate while providing service to spatially scattered nodes that are unable to establish a high bandwidth ground-to-ground communication. Adhering to the surge of interest in enabling connected intelligence [3, 4], one can envision the use of such aerial platforms as an effective means to implement *collaborative federated learning (FL)*, where few geographically scattered nodes collaborate to improve a common machine learning (ML) model. Such an FL use case requires periodic and high bandwidth communications to take place according to a given schedule. The planning of this schedule is directly dependent on the DSC's position, and it imposes various challenges. Moreover, assuming that all learners in an FL contain non-IID data that is useful to the overall model causes asynchrony between the learning input (epochs computed) of each node. Therefore, we consider the drone in the role of

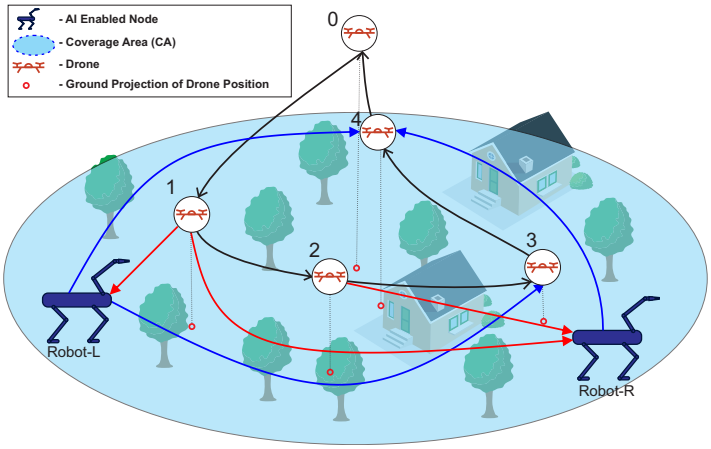


Fig. F.1: Toy illustration showcasing ML mode passing to the learners (red arrows) in the downloading (DL) phase, that then return the updated models in the uploading (UL) phase to achieve FL schedule as in Fig. F.2. The drone performs this only by adjusting its trajectory, that in this example starts and finishes at the center of the CA.

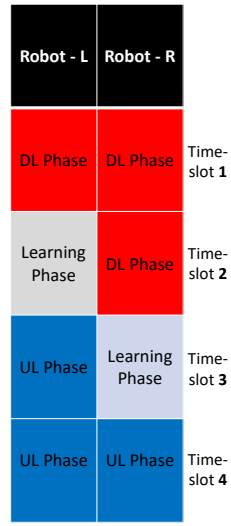


Fig. F.2: FL learners perform one iteration of learning thanks to balanced DL and UL phases.

an *orchestrator*, whose trajectory is the available degree of freedom that can be adjusted to minimize the maximum *staleness* (difference between most ML epochs and the least ML epochs computed at a learner in the FL).

1.1 State of the Art

The effects of dynamic DSCs that move in favor of users' locations has been investigated in [5–7], where the focus is on the superior spectral efficiency and latency achieved by DSCs in various scenarios. In our previous work [8], we demonstrated the impact of accounting for the dynamic movements of standalone DSC, equipped with a tilting directional antenna. Moreover, the work in [9] focused on the energy efficiency for DSC deployment, while the authors in [10] and [11] studied the problem of placement optimization of a single cell and interference-limited multi DSC deployments, respectively.

However, the consideration of dynamic drones calls for efficient trajectory planning. The work in [12] focused on finding the drone trajectory that achieves minimal UAV energy expenditure while serving multiple nodes with a rotary-wing UAV. In [13], the authors investigated energy efficiency by scheduling the sleep timers of ground wireless nodes as well as the UAV's trajectory. Purpose-first three-dimensional trajectory design

can be achieved as done in [14], whose goal was to maximize the minimum average data collection rate from all nodes for a stochastic channel model. Finally, considering the complexity of the issue, there is a strong incentive of solving trajectory optimization problems with the use of reinforcement learning (RL), as done in [15–17].

Drone cloudlet implementations receive a growing attention for edge computation purposes and offering portable processing services. In this setting, the works in [18] and [19] investigated the design of a drone trajectory, along with the problem of communications and computational resource allocation in favor of lowering the energy consumption of an Internet of Things (IoT) network. Combining this with the recent works on efficient offloading of the learning for RL, [20], sparked a new demand for drone-aided intensive edge computations. In a common centralized ML implementation, the drone would act as a sink for all the collected data which is then processed, as in the cloudlet design [18]. Additionally, FL implementations in drones have been a topic of significant interest in the literature. These prior works mostly consider the drones as learners [21–24]. However, even with current advances in energy efficient FL [25] and low power computation systems [1], we consider the concept of drone-mounted ML-computation hardware as heavy and energy inefficient, thus reducing the flight time of the UAV.

1.2 Drone Orchestrator for Reducing Staleness

The progress of robot implementations for laborious tasks in remote locations motivates investigating a setting of connected intelligence. In example, logging robots [26] that proceed with their main task of woodcutting can enhance their detection performance of critical flora and fauna through a collaborative learning process. In Fig. F.1, we show two distant ground robot-nodes L and R that are not energy restricted and are equipped with powerful ML computation equipment, and due to their spatial arrangement, require the communications support of a mobile drone. In this setting, both the learning and the sensing are distributed to the robot nodes that perform iterative improvements on a common ML model [27] with non-IID data [28] collected from their own sensors. As such, the drone assumes the role of an FL orchestrator that receives the model updates and aggregates them. This is a computational task that does not require powerful processors [27]. Such a setup sees potential practical implementations such as orchestrating automated agriculture, forestry, personalized healthcare [26], and border control [29] operations. Finally, this architecture can apply to a plethora of ML implementations, such as multi-perspective computer vision, multi-agent utility optimization, or semi-supervised parameter estimation. If needed, these implementations can also exploit the ability of FL to conceal sensitive information collected by the nodes.

In Fig. F.1 and Fig. F.2 we illustrate how each robot-node goes trough three phases: downloading (DL) model (red), learning, and uploading (UL) model improve-

[1] https://www.dji.com/dk/manifold-2

ments (blue). Transmission times to (DL) and from (UL) each node occupy useful time periods that would be preferably allocated for the learning phase at each robot. In addition, the learning at each node is impacted by its processing capability; higher processing capability at a node, with regards to other learners in the network, makes the data collected by its sensors more dominant when constructing the common model. Hence, the objective is to control the channel to each node through the drone-orchestrator's trajectory with the goal to aid the slow learning nodes (low processing capability) by modifying the DL and UL transmission times in order to minimize the work/learning discrepancies between nodes, called *staleness*.

Staleness is a cardinal metric for our setup since all nodes are assumed to possess useful data and, therefore, stragglers cannot be dropped. This creates asynchrony between the amount of learning each robot does. We model this asynchrony by the largest difference of epochs computed among the learners, which has been shown to be key for the performance of the next generation of asynchronous FL [30–32] The maximum staleness comes as a consequence of the asynchrony of such an FL implementation which is an issue that we want to tackle by implementing path planning in the duration of a single FL round. Solving the issue of staleness by only controlling the drone's trajectory requires new approaches as the transmissions occur with variable duration and only at the head and the tail of a pre-planned trajectory. This is in contrast to most works that are concerned with trajectory optimization problems whose goal is to maximize/minimize metrics that are often a direct representation of the aggregate rate instead of the fairness.

1.3 Main Contributions and Organization

The main contribution of this paper is to develop, to the best of our knowledge, the first framework that designs a drone trajectory for serving FL networks with the purpose of customizing per node metrics across longer time periods, thus addressing the problem of *staleness* for deployments of scattered nodes. Addressing staleness gives a unique optimization criterion for the trajectory problem since it tackles shortening the transmission periods while equalizing work across many learners. Such a challenge makes our problem significantly different from prior works on trajectory optimization because of the combination of non-linear resource (channel quality across a trajectory) for solving a combinatorial optimization problem. It is apparent that minimizing staleness is not only limited for FL uses, but it also applies to drone trajectory design for lowering working-time discrepancies for IoT fields, data freshness for drones data-sink, and various edge computation scenarios where imbalanced work may harm the outlook of the implementation and the efficiency of the underlying service.

Section 2 describes the considered system, introduces the wireless communications traffic model of the FL implementation, and proposes a novel metric that contributes towards a balance between the total amount of epochs computed and *staleness*. We

further compare the performance of two trajectory optimization approaches, successive convex programming and reinforcement learning, both detailed in Sections 3, and 4 respectively. The SCP requires that we approximate several metrics in order to get to a solvable form in a computationally efficient manner. The RL approach [33] makes no such demands and has the potential for future use in a stochastic and unpredictable environment. Moreover, a key novelty here is the fact that we created a more distance efficient hexagonal trajectory map and introduced a secondary RL experience buffer with only good memories that balances convergence and exploration. In Section 5 we compare and analyze both approaches. Finally, conclusions are drawn in Section 6.

2 System Model

We consider a geographical coverage area (CA) defined by a circular range with radius $D_{\max}$, containing a set of K spatially distributed static nodes with each scattered node labeled by $k \in \mathcal{K} = \{1, 2, .., K\}$. A single drone-orchestrator travels across this CA and has to complete its route within a predefined deadline T referred to as global cycle clock. The global cycle clock is defined by the FL implementation and it is crucial to the correct operation of an underlying task. Note that in both FL algorithms FedAvg [27] and FedProx [28] it is critical that not too much local work (very long T) takes place as it impairs the aggregate learning. On the other hand, too little local work (short T) means the transmission overhead will become dominant [27]. We further define N discrete timeslot intervals $i \in \mathcal{N} = \{1, 2, .., N\}$ each with a duration of Δ seconds as $T = N\Delta$. The size of Δ is determined based on the drone speed and CA radius as discussed in Section 4 and Section 5. The drone trajectory $\mathbf{P}^{\mathrm{d}} = \{\mathbf{p}_i^{\mathrm{d}}\} \in \mathbb{R}^{N\times 2}$, where each $\mathbf{p}_i^{\mathrm{d}}$ represents a way-point for each timeslot i as the i-th row of $\mathbf{P}^{\mathrm{d}}$ is decribed by the horizontal coordinates $\mathbf{p}_i^{\mathrm{d}} = (x_i^{\mathrm{d}}, y_i^{\mathrm{d}})$, while its altitude is always H. The drone can fly horizontally, limited by a maximum speed of $v_{\max}$, and it is equipped with a directional antenna that tilts to ensure coverage over the CA as in [34], so that basic control signaling is always supported. This does not imply satisfactory FL model transfer rate for distant users, making it necessary to move the drone closer to ensure faster model exchange. The K nodes are found on the ground (zero height) at positions $\mathbf{p}^k = (x^k, y^k)\ \forall k \in \mathcal{K}$, resulting in a drone-to-node k horizontal distance of:

$$d_{i,k}(\mathbf{p}_i^{\mathrm{d}}) = \sqrt{(x_i^{\mathrm{d}} - x^k)^2 + (y_i^{\mathrm{d}} - y^k)^2}, \tag{F.1}$$

where $k \in \mathcal{K}$. As such, the value of $d_{i,k}$ is subject to the anticipated movement of the drone at time i, which further impacts the data rate $R_{i,k}(\mathbf{p}_i^{\mathrm{d}})$ for each node. We do not adjust drone's height during the trajectory, as it is incompatible with the use of a directional antenna in a sense that it would affect the size of the CA (e.g., see [1, 2], and [8]). Moreover, adjusting the height can raise potential liability concerns with respect to collisions, as well as drastically increase the optimization complexity. We

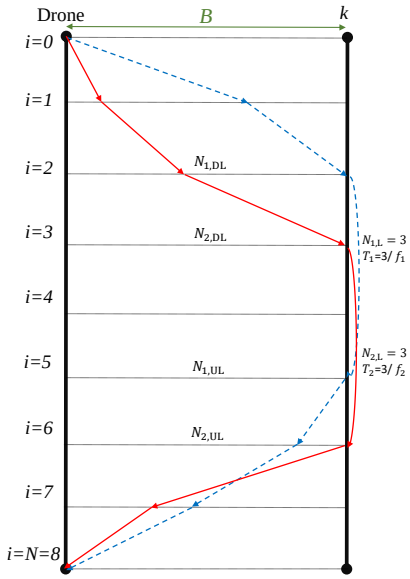


Fig. F.3: An example of the FL process with two learners; $k = 1$ blue-dashed line and $k = 2$ red-solid line. Here $N_{1,\mathrm{L}} = N_{2,\mathrm{L}} = 3$, and if $f_1 = f_2$, the system has zero *staleness* since both nodes have an equal amount of epochs.

also consider localization precision defined by a radius of r meters around the allocated drone position $\mathbf{p}_i^{\mathrm{d}}$.

2.1 Federated Learning Traffic Model

At the start of the FL cycle, each node starts with the DL phase that lasts $N_{k,\mathrm{DL}}(\mathbf{P}^{\mathrm{d}})$ timeslots, implicitly given as:

$$\sum_{i=0}^{N_{k,\mathrm{DL}}(\mathbf{P}^{\mathrm{d}})} R_{i,k}(\mathbf{p}_i^{\mathrm{d}}) \cdot \Delta = B \;\; \forall k \in \mathcal{K}, \tag{F.2}$$

where $R_{i,k}(\mathbf{p}_i^{\mathrm{d}})$ is the instantaneous data rate for timeslot i of node k. Each node concludes the FL cycle with the UL phase of duration $N_{k,\mathrm{UL}}(\mathbf{P}^{\mathrm{d}})$ implicitly given by:

$$\sum_{i=N-N_{k,\mathrm{UL}}(\mathbf{P}^{\mathrm{d}})}^{N} R_{i,k}(\mathbf{p}_i^{\mathrm{d}}) \cdot \Delta = B \;\; \forall k \in \mathcal{K}. \tag{F.3}$$

The leftover time in between both transmission phases is where the learning occurs for each learner. Since all phases need to be completed within N timeslots, the discrete

learning period $N_{k,\mathrm{L}}(\mathbf{P}^{\mathrm{d}})$ during the drone's trajectory lasts:

$$N_{k,\mathrm{L}}(\mathbf{P}^{\mathrm{d}}) = N - N_{k,\mathrm{DL}}(\mathbf{P}^{\mathrm{d}}) - N_{k,\mathrm{UL}}(\mathbf{P}^{\mathrm{d}}), \ \ \forall k \in \mathcal{K}. \tag{F.4}$$

Fig. F.3 illustrates the phases for $K = 2$ where the ML process is based on a model of size B Mb which, for simplicity, is assumed to be identical for both UL and DL. The illustration portrays a two-dimensional system that has a width of B Mbs and height of the number of discrete timeslots N. Each one of the dashed-blue ($k = 1$) and solid-red ($k = 2$) lines represent the FL stage. Meanwhile, the slope of the lines capture $R_{i,k}(\mathbf{p}_i^{\mathrm{d}})$ that changes across all timeslots because the drone moves horizontally and impacts the wireless channel.

However, ML is an iterative process where each learning pass of the sensed data is called an epoch T_k. Back in Fig. F.3, the allocated processing time for each node is three slots, and even though we mitigate the asynchrony introduced by the spatial arrangement, the computed epochs T_k depend on the processing capabilities of node k. To account for such learners, referred to as *stragglers* in the FL community, we scale the computational capability with a scalar value f_k. The value f_k represents the amount of epochs processed per second and is a collective measure of the processor cores and speed, ML accelerator (such as repurposed rasterisation cores) and/or the environment sampling rate. The number of epochs spent by device k is therefore a function of the flown trajectory $\mathbf{P}^{\mathrm{d}}$ and is calculated as:

$$T_k(\mathbf{P}^{\mathrm{d}}) = N_{k,\mathrm{L}}(\mathbf{P}^{\mathrm{d}}) \cdot \Delta \cdot f_k. \tag{F.5}$$

In classical ML, the more epochs computed and the more data is provided, the higher the expected model accuracy. FL is a cyclic process whereby at the end of the cycle there are K models with different weights $\boldsymbol{w}_{\mathrm{k}}$ that are received by the drone orchestrator and are aggregated to the common model [27] [28]. The orchestrator then initiates a new learning cycle and returns the aggregated model weights $\boldsymbol{w}_{\mathrm{d}}$ to all participating learners/nodes.

2.2 Staleness in a No-Drop Federated Learning

Since in an FL the data is non-IID distributed among learners, we model the performance of the FL as a no-drop FL where learning occurs in an asynchronous fashion and all participants are trustworthy. Like this, discrepancies in T_k between the different learners i.e. $|T_k(\mathbf{P}^{\mathrm{d}}) - T_l(\mathbf{P}^{\mathrm{d}})|; \forall k \neq l; k, l \in \mathcal{K}$, are referred to as *staleness* [31]. This creates an asynchronous FL where staleness undermines the total learning done $\sum_{k=1}^{K} T_k$ and therefore slows down FL convergence and lowers overall system accuracy in the training phase [30–32]. Moreover, both [28] that tackles learning without dropping straggles, and the asynchronous optimization [31] work, show that even when the local optimizer is designed for some asynchronous amount of work the maximum

staleness impacts the performance of the FL. Hence, we aim to improve learning performance when aggregating the collective model by minimizing the largest epoch number difference between any two learners:

$$s(\mathbf{P}^{\mathrm{d}}) = \max(|T_k(\mathbf{P}^{\mathrm{d}}) - T_l(\mathbf{P}^{\mathrm{d}})|);\ \forall k \neq l;\ k, l \in \mathcal{K}, \tag{F.6}$$

where T_k is relaxed to $T_k \in \mathbb{R}^+$, for the purpose of generalizing the analysis. To avoid fully neglecting good learners, we introduce the mean of the total number of epochs performed as a stabilizing factor. We can now define the primary optimization criterion for our drone trajectory as an average-anchored staleness (AAS):

$$\max_{\mathbf{P}^{\mathrm{d}}} \frac{1}{K} \sum_{k=1}^{K} T_k(\mathbf{P}^{\mathrm{d}}) - s(\mathbf{P}^{\mathrm{d}}), \tag{F.7}$$

where the system constraints of speed, initial location, and deadline are described in Section 3 for the SCP approach and Section 4 for the RL approac. As mentioned, different local FL optimizers can tolerate some asynchronous amount of work. Such tolerance s_{tol} can be accounted for in (F.7) by converting the second term $s(\mathbf{P}^{\mathrm{d}})$ to $\max(s(\mathbf{P}^{\mathrm{d}}), s_{\mathrm{tol}})$. Although the implementation in [32] would moderately tolerate maximum staleness of 4, we proceed the work by tackling the most challenging scenario where the trajectory would need to be optimized if no tolerance was allowed $s_{\mathrm{tol}} = 0$.

A more simplified look at (F.7) is that if node positioning is stochastic as in a point process, the goal of our optimization problem would map to reducing the maximum deviation in learning performed. This gives a good general overview that is data-agnostic [35], without the need to assume the impact of data at some particular learner and solely on spatial and computational performance. Therefore, AAS provides trajectories that serve an equally balanced amount of learning and *staleness*, which can be further enhanced with additional resource allocation techniques [36] combined with FL incentive calculations [37]. Finally, with (F.7) we bring asynchronous FL as close as reasonable by the disposable resources to a classical synchronous FL (where all learners compute the same amount of epochs in a cycle), without dropping any stragglers.

2.3 Propagation Environment

A node can belong to one of two propagation groups, nodes that have direct line-of-sight (LoS) or no-LoS (NLoS). As such, the path loss experienced for node k at time i is $\ell_{i,k}$ and becomes a sum of the free space path loss (FSPL) and the additional large-scale shadowing coefficient for each one of the propagation groups. We note that since we are concerned with lengthy transmission timescales of several seconds we use the mean fading coefficients for each propagation group, namely η_{LoS} and η_{NLoS}. This provides a well-generalized approach opposed to working with random variables for the fading calculations of the normally distributed excessive path loss, that is introduced due to

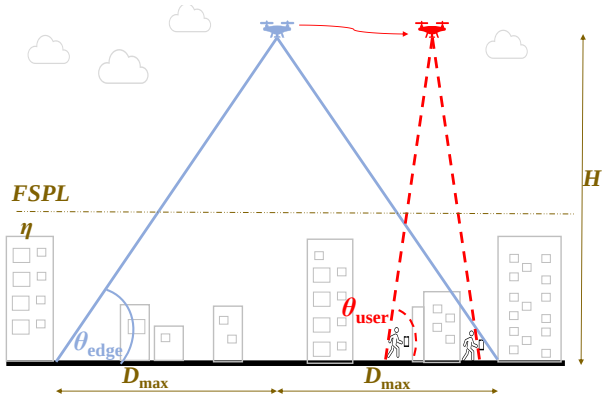


Fig. F.4: Reference LAP placement (solid blue) and a drone orchestrator in trajectory (dashed red) [8].

the large features of the topology [38]. Taking into account a directional antenna with directivity defined by G_t, the path loss experienced for each propagation group becomes:

$$\ell_{i,k,\mathrm{LoS}}(\mathbf{p}_i^{\mathrm{d}}) = -10\log(G_t) + 20\log(\sqrt{d_{i,k}^2(\mathbf{p}_i^{\mathrm{d}}) + H^2}) + C + \eta_{\mathrm{LoS}}, \tag{F.8}$$

and,

$$\ell_{i,k,\mathrm{NLoS}}(\mathbf{p}_i^{\mathrm{d}}) = -10\log(G_t) + 20\log(\sqrt{d_{i,k}^2(\mathbf{p}_i^{\mathrm{d}}) + H^2}) + C + \eta_{\mathrm{NLoS}}, \tag{F.9}$$

where log is a shortened version of the common logarithm $\log_{10}$ and the term C is a substitute for the carrier frequency f_c constant in FSPL $C = 20\log(\frac{f_c 4\pi}{c})$. Hence, a node's probability to belong to either group is directly dependent on the probability for a LoS to happen, $\mathrm{P}_{\mathrm{LoS}}(\mathbf{p}_i^{\mathrm{d}})$. To represent the probability we use the s-curve model defined by [39]:

$$P_{\mathrm{LoS}}(\mathbf{p}_i^{\mathrm{d}}) = \frac{1}{1 + a\exp(-b(\theta_{\mathrm{user}}(\mathbf{p}_i^{\mathrm{d}}) - a))}, \tag{F.10}$$

where a and b are constants dependent on the topological setting, and the elevation angle at user side $\theta_{\mathrm{user}}(\mathbf{p}_i^{\mathrm{d}})$, expressed in degrees $0 \leq \theta_{\mathrm{user}}(\mathbf{p}_i^{\mathrm{d}}) \leq 90$, is a function of the drone's horizontal position since the height H is fixed and $\theta_{\mathrm{user}}(\mathbf{p}_i^{\mathrm{d}}) = \arctan(\frac{H}{d_{i,k}(\mathbf{p}_i^{\mathrm{d}})})$, as illustrated in Fig. F.4.

Given the existence of only two propagation groups, the probability $P_{\mathrm{NLoS}}(\mathbf{p}_i^{\mathrm{d}}) = 1 - P_{\mathrm{LoS}}(\mathbf{p}_i^{\mathrm{d}})$ which finally allows us to calculate the final path loss expectation $L_{i,k}(\mathbf{p}_i^{\mathrm{d}})$ for node k as derived from (F.8), (F.9), and (F.10) and given in linear terms as a function

of the drone's position through $\ell_{i,k,\mathrm{LoS}}(\mathbf{p}_i^{\mathrm{d}})$ and $\ell_{i,k,\mathrm{NLoS}}(\mathbf{p}_i^{\mathrm{d}})$:

$$10\log[L_{i,k}(\mathbf{p}_i^{\mathrm{d}})] = \ell_{i,k\mathrm{LoS}}(\mathbf{p}_i^{\mathrm{d}}) \cdot P_{\mathrm{LoS}}(\mathbf{p}_i^{\mathrm{d}}) \tag{F.11}$$

$$+ \ell_{i,k\mathrm{NLoS}}(\mathbf{p}_i^{\mathrm{d}}) \cdot P_{\mathrm{NLoS}}(\mathbf{p}_i^{\mathrm{d}}) \tag{F.12}$$

$$= P_{\mathrm{LoS}}(\mathbf{p}_i^{\mathrm{d}})(\eta_{\mathrm{LoS}} - \eta_{\mathrm{NLoS}}) + \ell_{\mathrm{NLoS}}(\mathbf{p}_i^{\mathrm{d}}). \tag{F.13}$$

2.4 Data Rate

We consider a tilting antenna that keeps the whole CA in a communications coverage range and can provide a non-zero rate at any time. Such a setup reduces interference to users outside the CA [8] as well. The gain of the antenna G_t is given by its effectiveness E_r to fit an ideal conical beamwidth $G_t = E_r 10\log(G_I)$, where the ideal conical antenna has gain:

$$G_I = \frac{2}{1 - \sin\left(\theta_{\mathrm{edge}} \frac{\pi}{180}\right)}, \tag{F.14}$$

where $\theta_{\mathrm{edge}} = \arctan(\frac{H}{D_{\max}})$ is the elevation angle at the cell edge, when the drone is positioned in the middle of the CA and applies no antenna tilt, as shown on Fig. F.4. With this we define the final expected path loss expression, as a function of the location of the drone, and fully expanded [8]:

$$\begin{aligned} 10\log(L_{i,k}(\mathbf{p}_i^{\mathrm{d}})) = & \frac{\eta_{\mathrm{LoS}} - \eta_{\mathrm{NLoS}}}{1 + a\exp\left\{-b[\arctan\left(\frac{H}{d_{i,k}(\mathbf{p}_i^{\mathrm{d}})}\right) - a]\right\}} \\ & + 20\log\left(\sqrt{d_{i,k}^2(\mathbf{p}_i^{\mathrm{d}}) + H^2}\right) \\ & - E_r 10\log\left[\frac{2}{1 - \sin\left(\theta_{\mathrm{edge}} \frac{\pi}{180}\right)}\right] + C + \eta_{\mathrm{NLoS}}. \end{aligned} \tag{F.15}$$

We consider an orthogonal multiple access scheme (e.g., using frequency division multiple access (FDMA)), thus, we consider a noise-limited system with no interference for both DL and UL phases. Moreover, to emphasize the importance of the drone position in its trajectory we preallocate a frequency spectrum W that remains constant for each node. Therefore, at each timeslot the achievable rate becomes:

$$R_{i,k}(\mathbf{p}_i^{\mathrm{d}}) = W\log_2\left[1 + \frac{P_t}{W N_0 L_{i,k}(\mathbf{p}_i^{\mathrm{d}})}\right], \tag{F.16}$$

where P_t is the transmission power that is assumed to be identical at both node and drone side, while N_0 is the noise spectral density linearly scaling the noise with the channel bandwidth W.

3 Problem Analysis and Convex Approximation for Trajectory Optimization

In this section, we determine the optimal trajectory of the drone orchestrator in a way to minimize the discrepancy between the T_k values while maximizing each individual T_k value for a given mission completion time using the SCP technique. We use this method with the overarching goal of devising an algorithm that will provide a solution in deterministic polynomial time for any scenario with arbitrary arrangement of nodes. Due to the several non-convex parameters involved, this is non-trivial and requires a combination of several analytical techniques.

As defined in Section 2, the value of the length of a timeslot $\Delta = \frac{T}{N}$ is selected such that any position reachable within that timeslot does not impose significant changes in the rate performance; i.e. can be considered to be approximately unchanged. Hence by first order Taylor approximation, at some time instant t, the discrete position of the drone at the time slot $t + \Delta$ can be approximated as:

$$\mathbf{p}^{\mathrm{d}}_{i+1} = \mathbf{p}^{\mathrm{d}}_i + \mathbf{v}^{\mathrm{d}}_i \Delta + \frac{1}{2}\mathbf{a}^{\mathrm{d}}_i \Delta^2, \ \ \forall i \in \mathcal{N}. \tag{F.17}$$

Similarly, the velocity vector for the time slot $t + \Delta$ can be approximated as:

$$\mathbf{v}^{\mathrm{d}}_{i+1} = \mathbf{v}^{\mathrm{d}}_i + \mathbf{a}^{\mathrm{d}}_i \Delta, \ \ \forall i \in \mathcal{N}. \tag{F.18}$$

The use of time discretization reduces the number of variables to $2 \cdot N$, resulting in the following *staleness* minimization problem:

$$(\mathrm{P1}) : \underset{\mathbf{P}^{\mathrm{d}}}{\text{maximize}} \ \ \frac{1}{K}\sum_{k=1}^{K} T_k(\mathbf{P}^{\mathrm{d}}) - s(\mathbf{P}^{\mathrm{d}}), \tag{F.19}$$

$$\text{s.t.} \ \ |T_k(\mathbf{P}^{\mathrm{d}}) - T_l(\mathbf{P}^{\mathrm{d}})| \leq s(\mathbf{P}^{\mathrm{d}}), \ \ k, l \in \mathcal{K}, \tag{F.20}$$

$$\frac{2BT}{\Delta \sum_{i=1}^{N} R_{i,k}(\mathbf{p}^{\mathrm{d}}_i)} + \frac{T_k(\mathbf{P}^{\mathrm{d}})}{f_k} \leq T, \ \ \forall k \in \mathcal{K}, \tag{F.21}$$

$$\left\|\mathbf{p}^{\mathrm{d}}_{i+1} - \mathbf{p}^{\mathrm{d}}_i\right\| \leq \min\left\{2r, \Delta v_{\max}\right\}, \tag{F.22}$$

$$T_k(\mathbf{P}^{\mathrm{d}}) \geq 1, \ \ \forall k \in \mathcal{K}, \tag{F.23}$$

$$\mathbf{p}^{\mathrm{d}}_1 = \mathbf{p}_I, \tag{F.24}$$

$$\mathbf{v}^{\mathrm{d}}_1 = \mathbf{v}_I, \tag{F.25}$$

$$\mathbf{v}^{\mathrm{d}}_N = \mathbf{v}_F, \tag{F.26}$$

$$\left\|\mathbf{v}^{\mathrm{d}}_i\right\| \geq v_{\min}, \ \ \forall i, \tag{F.27}$$

$$(\mathrm{F.17}), (\mathrm{F.18}). \tag{F.28}$$

Maximizing the objective function (F.19) is equivalent to minimizing the maximum difference between the T_k values of the nodes in the CA. (F.21) is the mission completion

time constraint, where the first term accounts for the time required for transmitting both uplink and downlink the B bits of data. In the aforementioned term, we approximate the data rate during the UL and DL phases as the average rate through the full time T. This avoids having to convert the problem into a mixed-integer one. Constraint (F.22) represents the localization precision limitation by which the maximum distance between consecutive positions of the drone is limited to $2 \cdot r$ as given back in Section 2; (F.24) is the initial drone position constraint; while (F.23) guarantees at least one iteration of the local ML model with the given data subset for every node.

It is evident that the average rate of (F.21) depends on $\mathbf{p}_i^{\mathrm{d}}$ through $L_{i,k}(\mathbf{p}_i^{\mathrm{d}})$ and therefore the LoS probability. The complex expression of $P_{\mathrm{LoS}}(\mathbf{p}_i^{\mathrm{d}})$ as reported in (F.10) makes solving (P1) using convex methods outside our computational capabilities. To circumvent the impact of the LoS probability on function convexity, we use a homogeneous approximation for the LoS probability [12]. Indeed, since $P_{\mathrm{LoS}}(\mathbf{p}_i^{\mathrm{d}})$ is an increasing function of the elevation angle, we consider the LoS probability of all the nodes equal to the LoS probability of the edge user device [11], i.e. $P_{\mathrm{LoS}}(\mathbf{p}_i^{\mathrm{d}}) \approx P_{\mathrm{LoS}}(\mathbf{p}_{\mathrm{edge}}^{\mathrm{d}}) \forall i, k : d_{i,k} \leq D_{\max}$. Hence the corresponding rate value is the lower bound of $R_{i,k}(\mathbf{p}_i^{\mathrm{d}})$; which in a LoS dominated region is equal to the actual $R_{i,k}(\mathbf{p}_i^{\mathrm{d}})$ value. Therefore the lower bound of the average achievable rate of node k when the drone is at time i becomes:

$$\overline{R}_{i,k}(\mathbf{p}_i^{\mathrm{d}}) = W \log_2 \left[1 + \frac{\gamma_o}{\left(H^2 + d_{i,k}^2 \right) \overline{L}} \right], \tag{F.29}$$

where $\overline{L} = 10^{\left\{ \left[P_{\mathrm{LoS}}(\mathbf{p}_{\mathrm{edge}}^{\mathrm{d}})(\eta_{\mathrm{LoS}} - \eta_{\mathrm{NLoS}}) + \eta_{\mathrm{NLoS}} \right] / 10 \right\}}$ and $\gamma_o = \dfrac{P_t c^2 G_I^{E_r}}{W N_o (4\pi f)^2}$ are obtained by substituting (F.9) in (F.11). This rate-equivalent approximation is useful in providing solutions to the convex approximation approach but is not necessary in Section 4 where we approach the same problem using reinforcement learning.

The objective function and the constraints (F.20), (F.22)-(F.26), (F.17), (F.18) are convex functions of the position and the velocity variable. However, the mission completion time constraint (F.21) is non-convex because of the data rate expression in (F.29) and the minimum velocity constraint is also a non-convex function of the velocity variable. Hence (F.19) cannot be solved directly by using a convex optimization technique. We will thus address this challenge by using the sequential convex programming technique.

3.1 Sequential Convex Programming for Trajectory Optimization

The SCP approach allows us to represent a non-convex optimization problem as a sequence of convex optimization problems, and, then, solve them iteratively until the

solution converges [40]. The candidate solution obtained through the SCP is guaranteed to satisfy the Karush-Kuhn-Tucker (KKT) conditions of the actual non-convex problem. Hence the solution obtained through the SCP technique cannot be considered as the global optimum of the problem, but, instead, it is a local optimum. However, the rate of convergence of the SCP algorithm is linear in complexity thereby making it suitable for solving real-time drone positioning problems [40].

To solve (F.19) using the SCP technique, we introduce an auxiliary variable $\lambda_{i,k}(\mathbf{p}_i^{\mathrm{d}})$ to tackle the non-convex constraint (F.21); where $\lambda_{i,k}(\mathbf{p}_i^{\mathrm{d}})$ is the first order Taylor approximation of $\overline{R}_{i,k}(\mathbf{p}_i^{\mathrm{d}})$ expressed as:

$$\lambda_{i,k}(\mathbf{p}_i^{\mathrm{d}}) \leq \log_2\left(1+\frac{\gamma}{H^2+\left\|\mathbf{p}_i^{\mathrm{d},l}-\mathbf{p}^k\right\|^2}\right) - \alpha_{k,i}^{\mathrm{d},l}\left(\left\|\mathbf{p}_i^{\mathrm{d}}-\mathbf{p}^k\right\|^2-\left\|\mathbf{p}_i^{\mathrm{d},l}-\mathbf{p}^k\right\|^2\right), \tag{F.30}$$

where $\alpha_{k,i}^{\mathrm{d},l}=\dfrac{\gamma\log_2 e}{(H^2+\gamma+\left\|\mathbf{p}_i^{\mathrm{d},l}-\mathbf{p}^k\right\|^2)(H^2+\left\|\mathbf{p}_i^{\mathrm{d},l}-\mathbf{p}^k\right\|^2)}$; $\left\{\mathbf{p}_i^{\mathrm{d},l}\right\}$ is the set of drone positions obtained from the l^{th} iteration; $\gamma=\gamma_o/\overline{L}$. The right-hand side of (F.30) is a concave function of $\mathbf{p}_i^{\mathrm{d}}$; hence (F.30) is a convex constraint of the variables $\mathbf{p}_i^{\mathrm{d}}$ and $\lambda_{i,k}(\mathbf{p}_i^{\mathrm{d}})$. Similarly the non-convex minimum velocity constraint can be equivalently represented as;

$$\beta \geq v_{\min}, \tag{F.31}$$

$$\left\|\mathbf{v}_i^{\mathrm{d},l}\right\|^2+2\left(\mathbf{v}_i^{\mathrm{d}}-\mathbf{v}_i^{\mathrm{d},l}\right)\left(\mathbf{v}_i^{\mathrm{d},l}\right)^T\geq\beta^2, \tag{F.32}$$

with (F.30), (F.31), and (F.32), the optimization problem (F.19), can be equivalently written as;

$$(\mathrm{P1.1}):\underset{\mathbf{P}^{\mathrm{d}}}{\text{maximize}}\ \ \frac{1}{K}\sum_{k=1}^{K}T_k(\mathbf{P}^{\mathrm{d}})-s(\mathbf{P}^{\mathrm{d}}), \tag{F.33}$$

$$\text{s.t.}\ \ \frac{2BT}{\Delta\sum_{i=1}^{N}\lambda_{i,k}(\mathbf{P}^{\mathrm{d}})}+\frac{T_k(\mathbf{P}^{\mathrm{d}})}{f_k}\leq T,\ \ k\in\mathcal{K}, \tag{F.34}$$

$$(\mathrm{F.17}),(\mathrm{F.18}),(\mathrm{F.20}),(\mathrm{F.22})-(\mathrm{F.26}),(\mathrm{F.31}),(\mathrm{F.32}). \tag{F.35}$$

Finally, the objective function and the constraints of (F.33) are convex and the problem can be iteratively solved as reported in Algorithm 1. In our case, every iteration makes use of available convex optimization tool box by MATLAB called CVX [41]. Finally, we note that the resulting trajectory may be a local maximum that is influenced by the initialization of the approximated trajectory positions. Nonetheless, this is sufficient given that SCP should provide reliably similar performances across different testing scenarios and executes in polynomial time.

Algorithm 1: SCP for Trajectory Optimization

Input: $l = 0; \{\mathbf{p}_i^{\mathrm{d},l}\}, \{\mathbf{v}_i^{\mathrm{d},l}\}\ \forall i \in \mathcal{N}$
repeat
Solve (P1.1) using available convex optimization tool box to obtain the optimal solution: $\{\mathbf{p}_{\mathrm{o},i}^{\mathrm{d}}\}, \{\mathbf{v}_{\mathrm{o},i}^{\mathrm{d}}\}$
$l = l + 1$;
$\mathbf{p}_i^{\mathrm{d},l} = \mathbf{p}_{\mathrm{o},i}^{\mathrm{d}}; \mathbf{v}_i^{\mathrm{d},l} = \mathbf{v}_{\mathrm{o},i}^{\mathrm{d}} \quad \forall i \in \mathcal{N}$
until The fractional increase in the objective function of (P1.1) is less than a threshold δ
Output: Optimal drone positions: $\{\mathbf{p}_i^{\mathrm{d},l}\}$

4 Reinforcement Learning for Trajectory Optimization

In Section 3, we analyzed (F.7) by converting it to a convex form and solving it iteratively as in SCP. Due to the shortcomings of SCP, we are interested in evaluating the performance and suitability of RL [33] particularly without making approximations on the original scenario proposed in Section 2. In addition, RL has a potential to work with more difficult and realistic models that include stochastic rates, node positions, and variations in the drone speed, all of which can be suitable for future real-world variations of the problem. Given this motivation, we will now investigate how to design an RL that solves the trajectory problem as per the defined model in Section 2. In general, RL is a learning approach that is used for finding the optimal way of executing a task by letting an entity, named ***agent***, take ***actions*** that affect its ***state*** within the acting ***environment*** [33]. The agent improves over time by incorporating the ***rewards*** it had received for its appropriate performance in all ***episodes*** within that same environment.

4.1 Environment Remodelling

We discretize the continuous environment by splitting the horizontal space into an *odd-r* horizontal layout hexagonal lattice, such as illustrated in Fig. F.5. Beyond reducing the continuous space, the grid becomes a truthful representation of the trajectory limitations due to drone positioning. Setting r as the inner radius of each hexagon, shown with yellow at Fig. F.5, does not harm the precision of the planned trajectory and offers superior packing. This also requires the position of each static node k defined by the center of the hexagon it is in. Using the nearest neighbouring center distance $2 \cdot r$

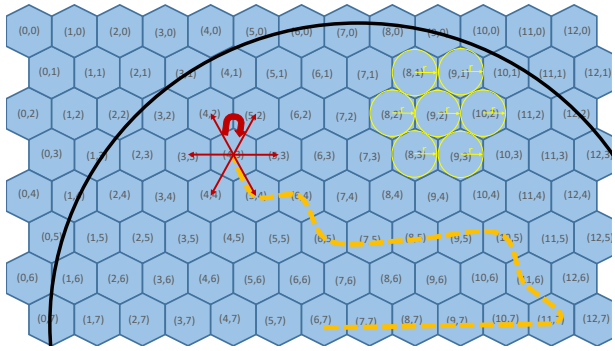


Fig. F.5: A cropped illustration of the *odd-r* hexagonal lattice (blue) fitting the CA's border (black solid line), with few examples of the precision circles with inner radius r (yellow). The drone has flown a trajectory (dashed orange line) and is faced with deciding among the 7 actions (solid red arrows).

division of CA's diameter $2 \cdot D_{\max}$, results in a two dimensional $M \times M$ lattice of size:

$$M = \frac{D_{\max}}{r}. \tag{F.36}$$

We define as a single episode the completion of the deadline T during which all nodes need to have their updates sent back to the drone and is discretized by $\Delta = T/N$ for all N timeslots. The drone starts at point $p_0^{\mathbf{d}}$ and finishes at point $p_N^{\mathbf{d}}$ by travelling from point i to point $i+1$ with a maximum speed limitation of $\frac{p_{i+1}^{\mathbf{d}} - p_i^{\mathbf{d}}}{\Delta} \leq v_{\max}^{\mathbf{d}}$. Since the drone speed is limited, we allow for full trajectory resolution of the hexagonal lattice by setting:

$$\Delta = \frac{2 \cdot r}{v_{\max}^{\mathrm{d}}}, \tag{F.37}$$

resulting in a total number of timeslots as:

$$N = \frac{T \cdot v_{\max}^{\mathrm{d}}}{2 \cdot r}. \tag{F.38}$$

In this way, at every time-step, the drone has the choice to move to each of the six neighbouring hexagons or not move at all, illustrated with red at Fig. F.5, totalling to an action space of 7. Since the trajectory is expressed by all previous drone movements the problem of trajectory planning becomes a Markov Decision Process (MDP) that under special conditions has a size of an N-tuple with base 7 totalling to 7^N states.

4.2 Drone Trajectory as an MDP

To make the trajectory problem solvable by RL, the main purpose of the reformulation until now was to make it representable by an MDP. To this end, we use a standard

MDP representation as a 4-tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R})$ with sets: state space $\mathcal{S}$, action space $\mathcal{A}$, probability of transition $\mathcal{P}$, and a state-action reward map $\mathcal{S} \times \mathcal{A} -> \mathcal{R}$.

- $\mathcal{S}$ - Each state in the set is defined by the drone's (x, y) coordinates in the drone trajectory taken, depicted with orange back in Fig. F.5, the leftover timeslots that lack coordinates are padded totalling to $2 \cdot N$ states. We additionally specify each state with the number of leftover timeslots $N - i$, and (x^k, y^k), f_k, and $R_{i,k}$ for each node k. Thus we have a state described by $F = 1 + (2 \cdot N) + (4 \cdot K)$ features. Since the state space for the MDP scales with the number of timeslots N, the complexity of trajectory optimization becomes more challenging to the RL agent from two additional perspectives. A larger N directly increases the size of input features of the network F while it also extends the time for which the agent needs to finish a whole episode, providing less experience for the same time spent learning.

- $\mathcal{A}$ - The action space is defined by all possible movement directions on the sides of the hexagon plus the action of remaining in the same place formatted into a 7-tuple.

- $\mathcal{P}$ - Since the defined MDP is deterministic, no randomness is included in the set and all transitions follow the agent's decisions. Therefore, the next state is a direct consequence of the action that the agent takes, which is the maximal value among the seven outputs in the tuple.

- $\mathcal{R}$ - Since the performance of the drone can only be known once it completes the trajectory, defining continuous rewards for the agent will lead to a suboptimal performance due to the imbalance between the progressive rewards and the optimization criterion. Therefore, the reward set $\mathcal{R}$ is 0 except for the end of the episode where the one-time reward is as calculated from (F.7). Through this single reward the RL agent needs to learn to utilize its movements for, distributing the rates during DL, moving during the learning period, and distributing the rates during UL.

4.3 Deep Q-Learning

Due to the size of MDP, we create an RL agent as a feed-forward neural network (NN), with F input neurons, Y hidden states each with the same number of neurons Z, all using rectified linear (ReLU) activation functions, and an output neuron count of 7. Compared to other state of the art Q-function approximators, our selected NNs provide reasonable accuracy with lower convergence time when compared to e.g. recurrent NNs (RNNs). When receiving the current state, described with F features as input, the NN agent outputs its evaluation for all seven actions that can be taken. However, the use of NNs in RL tasks may fail to converge especially in problems with complex optimal policy

in great state space [42], such as ours. Therefore, we rely on deep RL, using double Q-learning and experience replay methods derived from [43] to bring the problem as close as possible to traditional supervised learning.

Experience replay requires that we store past episodes in a replay buffer. An experience e is defined as a tuple of five elements that occurred during action step t as in $e_t = (s_t, a_t, r_t, s_{t+1}, d_t)$ where: s_t was the starting state of the agent, a_t is the action that the agent took, r_t is the reward that the agent received, s_{t+1} the state at which it arrived, and $d_t = \{0, 1\}$ is an indicator for a terminating action that has 1 if the action finishes the episode or 0 if it leads to another state. This allows the use of mini batches of size β from the stored experiences. However, considering the vast amount of possible trajectories that the drone can try there is still a need to reduce the many unproductive trajectories while exploring the state space. Hence, we also include a secondary experience buffer that only stores the ten best performing episodes. Both buffers are used for training each episode as it was observed that such approach accelerates the convergence while maintaining the exploration.

Finally, for the double-Q-learning RL algorithm, we need to keep two separate agents with the same properties but with different weight values $w_{\boldsymbol{P}}$ and $w_{\boldsymbol{T}}$. As such they will output a different Q-action function when given the same state. One is used to choose the actions, called a primary model $Q_{\mathrm{P}}(s_t, a_t)$, while the other model evaluates the action during the training, called a target model $Q_{\mathrm{T}}(s_t, a_t)$. Therefore training occurs when taking a batch of experiences e_t from the buffer that is used to update the model as:

$$Q_{\mathrm{P}}^{\mathrm{new}} = (1 - \alpha)Q_{\mathrm{P}} + \alpha[r_t + (1 - d_t)\gamma \max Q_{\mathrm{T}}(s_{t+1}, a)], \tag{F.39}$$

where $\max Q_{\mathrm{T}}(s_{t+1}, a)$ is the action chosen as per the agent, α is the learning rate which was an input to the Adam optimizer [44], and γ is a discount factor that reduces the impact of long term rewards. We implement this with soft updates where instead of waiting several episodes to replace the target model with the primary. The target model receives continuous updates discounted by value τ as in $w_{\mathrm{T}} = w_{\mathrm{T}} \cdot (1 - \tau) + w_{\mathrm{P}} \cdot \tau$. Finally, to input new instances in the experience memory, the state space was sampled using the ϵ-greedy strategy. Here, with probability ϵ we take an action uniformly at random from the action space A, while with probability $1 - \epsilon$ we act greedily in favor of the agent's decision. After each episode, we lower the ϵ value by multiplying it with a decay coefficient ψ as in $\epsilon^{\mathrm{new}} = \epsilon \cdot \psi$.

4.4 Training an RL Agent for Trajectory Optimization

In our implementation the drone is trained in an offline manner. In other words, the drone will compute the full trajectory before taking any action in the simulated environment, which is modeled as an MDP. Since we use off-policy learning with sampling, this is easily adaptable to online training, where the ML model would be continuously trainable even during its operation. This is done with scalability in mind as we anticipate

Table F.1: Training parameters for RL trajectory optimization

Label	Definition	Value
γ	Q-learning discount factor	0.9999
α	Learning Rate	0.0001
τ	Soft copy coefficient for double Q-learning	0.005
Y	Number of hidden layers	3
Z	Number of neurons per hidden layer	512
ϵ	Greedy epsilon coefficient	0.5
ψ	Decay factor for epsilon	0.99
β	Number of randomly chosen experiences per batch	256

that integrated FL implementations require more fine tuned reward systems.

During the training process, RL needed roughly 50-100 episodes of replaying the same scenario solely to understand the situation and converge to stable flying routes that do not incorporate random erratic movements. Once such stability was achieved, the RL agent managed to learn to fly towards slow learners and away from learners that received too much attention. After an extensive experimentation, the parameters given in Table F.1 resulted in a good learning progress. The results provided in the Section 5 were achieved within 400 episodes of training, which took roughly $80 - 120$ minutes of training on a GPU accelerated implementation of the script written in Python.

5 Simulation Results and Analysis

The purpose of our simulation is to best evaluate the behaviour of our proposed solutions to the AAS minimization problem that arises in asynchronous FL networks. Moreover, we aim to both validate our framework and decide on an optimization approach that handles the problem based on the values introduced in Sec. 2. Since the performance of the provided trajectories strongly depends on the arrangement of the nodes, both approaches were put against a few handpicked scenarios. We chose this testing approach as aggregation of many randomized runs is unlikely to provide useful results due to the uniqueness of each scenario. Additionally, as our goal is to inspect the usefulness of using trajectory when maximizing AAS, we test on deployments with few nodes. In such scenarios it is easier to sample a challenging scenario for the optimization problems and execute a performance assessment that is directly observable.

In Table F.2 we list the parameters used for testing both solutions. The parameters were inspired by object detection models as in IMAGEAI's YOLOv3 [2], being served by

[2]https://imageai.readthedocs.io/en/latest/detection/

Table F.2: Testing environment settings [8, 38, 45]

Label	Definition	Value
f_c	Channel Carrier Frequency	5.8 Ghz
W	Channel Bandwidth for each k	20 Mhz
N_0	Noise Spectral Power	-174 dBm/Hz
H	Drone's flying altitude	100 m
$v_{\max}$	Maximum achievable drone speed	10 m/s
Δ	Time Discretization Interval	0.2 s
P_t	Transmission Power	23 dBm
r	Localization Precision Radius	1 m
B	Machine Learning Model Size	2000 Mb
$D_{\max}$	Radius of Coverage Area	260 m
T	Federated Learning Cycle Deadline	30 s
E_r	Antenna Effectiveness	0.6
a	P(LoS) Constant for Suburban topology	4.88
b	P(LoS) Constant for Suburban topology	0.43
η_{LoS}	Added Large Scale for LoS group	0.2
η_{NLoS}	Added Large Scale for NLoS group	24

a rotary-wing UAV [3] in a suburban environment. We judge the performance of the algorithms by how well they maximize the AAS metric from (F.7) that is directly dependent on the average learning done by all nodes subtracted by the achieved *staleness s*. As such, the drone trajectory optimization becomes a balancing act of anticipating moving towards and away from specific learners and pre-calculating the possible *staleness* for that learning cycle. The results shown in this section are calculated with a simulation of the non-approximated scenario in Section 2, common for both SCP and RL.

For testing, we assume that the drone should start at the center of the CA, $p_0^d = [0, 0]$. This is done in accord with many works that use a recharging or a battery-swap station [46] at the center of the CA in order to offer uninterrupted, seamless, and standalone service. In our simulated scenarios, we consider the case in which the drone has finished a battery swap from a ground station in the center of the CA, elevated itself to a height H, and initiated its service while it is at a very unbalanced position with regards to the node arrangement.

In this way we test the trajectory optimization approaches in a very challenging environment, where the drone trajectory starts in an unbalanced and therefore unfavorable position with regards to the ground FL network. These simulations cover the first cycle of the drone flight, after the end of which, changes in the area may occur as

[3] https://www.dji.com/dk/matrice100/info#specs

Table F.3: Coordinates and Computational Capability for each node k in all three tested scenarios.

	Straight $K=2$			**Hidden** $K=3$			**Forced** $K=3$		
	X_k	Y_k	f_k	X_k	Y_k	f_k	X_k	Y_k	f_k
k=1	60	129	1	10	43	1	0	0	1
k=2	260	0	0.7	29	11	1	1	183	0.7
k=3	N/A	N/A	N/A	169	183	0.7	40	43	0.7

some learners drop due to lack of data or new learners appearing in other areas of the CA. Therefore, in each cycle the drone will perform swings across the field of learners, until the time comes for the drone to go back to the center, recharge, and get back to orchestrating the FL network. We tested our approach and evaluated its performance

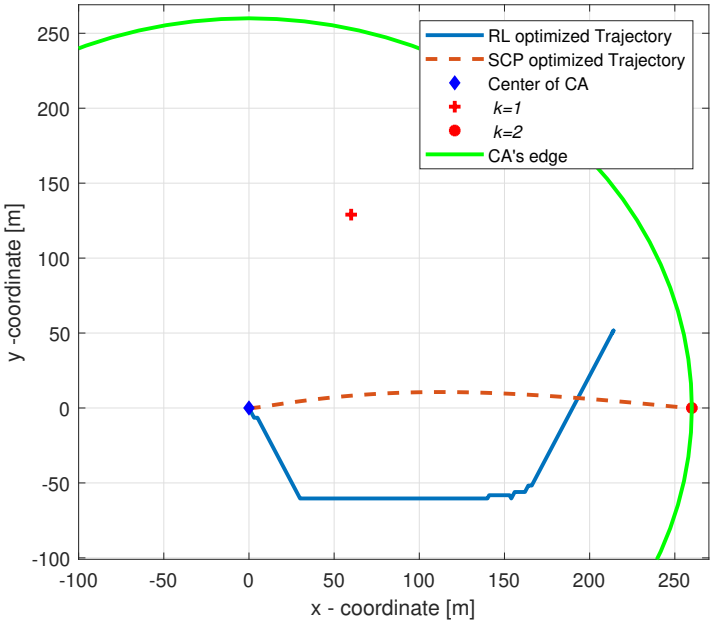


Fig. F.6: Staleness minimization solutions for **Straight** where: RL yields an AAS of 7 with s=3.7; SCP yields AAS=7.32 with s=3.

in Table F.3. These three scenarios are representative of special case arrangements in which the drone needs to take quick action to best improve the FL's performance for that specific FL cycle. The computational capability f_k for each device in Table F.3 is also taken in reference of the task of object detection, where we expect that one epoch

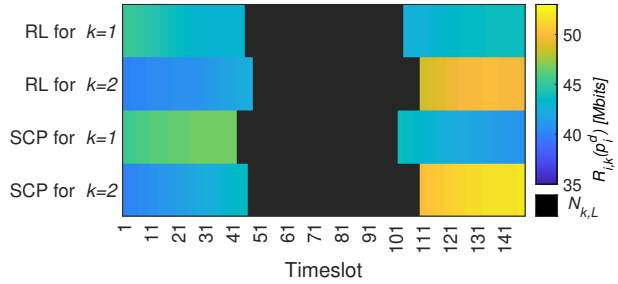


Fig. F.7: A datarate heatmap showcasing DL (left) - Learning (middle) - UL (right) schedule for both RL and SCP approaches for each device in **Straight**.

should last roughly one second on a computationally powerful node.

5.1 Straight Trajectory

The first arrangement, and named **Straight**, considers only two users $K = 2$, where the drone has a clear path to the slowest learner located in the rightmost corner. If no trajectory is considered, and the service provider is static at the starting location it would achieve an AAS of 3.90 with $s = 5.74$. The SCP solution provides a direct flight towards the slowest learner and achieving an AAS of 7.32 and a *staleness* $s = 3$, outperforming the static implementation by 87.7%. On the other hand, the RL algorithm takes a complicated trajectory, as shown on Fig. F.6, and achieves worse results an AAS of 7 with $s = 3.7$, outperforming the static implementation by 79.48%.

Recall that, although no stochastic parameters were involved, the ϵ-greedy approach was used in the RL training for the purpose of exploring the state space. This is significant since it leads the drone to position itself in undesirable locations, at times. In such cases, the agent eventually receives a lower reward even though the trajectory change was caused by a random epsilon event. Thus, RL produces a skewed trajectory in which the drone dramatically evades the good learner $k = 2$ in a behaviour that is most likely due to overstating the importance of *staleness* and the positioning of $k = 2$. RL is outperformed in this simple scenario mostly due to the lack of overfitting mitigation for the Q-learning agent. In future works, we intend to investigate an improved agent with a combination of RNNs and dropout as in [47].

In Fig. F.7, we show the chronological progression of the data rates for the ML model transmissions for each link in the Straight scenario. The schedule is clearly visible, starting with the DL phase (colored bars on the left), followed by a period of learning (central black bars), and finalize with the UL phase (colored bars on the right). Therefore, the width of each bar manifests the duration of the phase for that node. The graph enables us to further analyze the implication of the trajectory, and we notice that the major difference occurs in the DL phase where the RL provided trajectory manages

to decrease the rate towards $k = 1$ enough to finish later than $k = 1$ with the SCP trajectory. Since this also comes at a cost to the rate for the straggler node $k = 2$, this measure does not sufficiently improve the staleness value in the RL case so as to outperform the SCP provided solution even though their straggler performance is nearly equal.

5.2 One Hidden Node

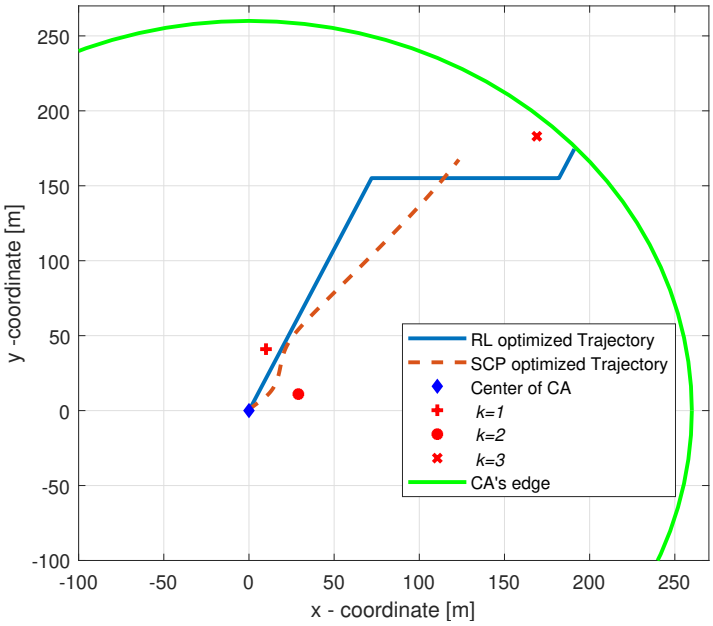


Fig. F.8: Staleness minimization solutions for **Hidden** where: RL yields an AAS of 7.8 with s=3.7; SCP yields an AAS of 7.1 with s=4.72.

The second arrangement, shown in Fig. F.8 and named **Hidden**, has three learners $K = 3$ present in the CA. Here the slowest learner is hidden far away from the center with two well performing nodes in the way of the direct trajectory connecting it. If static at the starting location, the orchestrator would achieve an AAS of 4.52 with $s = 7.46$. Both algorithms show prowess to discover the policy of breaching the barrier of fast learners to get to the slowest, hence providing an adequate solution. However, the RL algorithm does this much more efficiently with an AAS of 7.8 with $s = 3.7$, outperforming the static implementation by 72.56%. RL outperforms its SCP counterpart that achieved an AAS of 7.1 with $s = 4.72$, outperforming the static implementation by 57.08%. In both SCP and RL implementations, the drone is trying to leave the barrier of good learners

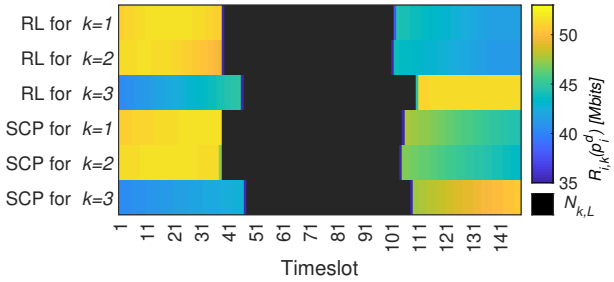


Fig. F.9: A datarate heatmap showcasing DL (left) - Learning (middle) - UL (right) schedule for both RL and SCP approaches for each device in **Hidden**.

as fast as possible and move towards the neglected user to match the rate it gave to the fast learners in the DL phase. In this testing scenario, using unapproximated rate goes massively in the benefit of discovering good positions with high precision for the RL approach.

Additionally, in Fig. F.9, we show the evolution of the transmission schedule for the **Hidden** scenario and highlight the better performance of the RL implementation particularly when dealing with the slow learner $k = 3$. Here, the RL guided drone travels very close to $k = 3$ during the final UL phase and, thus, it achieves near maximum rates and transfers ≈ 52 Mbits each timeslot.

An important note is that during RL training, it was observed that the RL agent would sometimes try a policy of flying away from all nodes in an effort to reduce the performance of nodes $k = 1$ and $k = 2$. This signifies the superiority of using our AAS metric as opposed to solely relying on s as an optimization metric since AAS "anchors" the drone from moving away from all learners. If we only relied on *staleness* as an independent optimization metric it would reflect heavily in the total amount of computation work done by the FL nodes, therefore slow down FL convergence.

5.3 Forced Departure

Next, in Fig. F.10, we consider a third scenario named **Forced** that has one well performing node directly in the center of the CA and two slow learners scattered at two positions away from the center. This scenario forces the drone orchestrator to quickly move out from its starting position to the closest and fastest learner, to serve a more balanced role away and towards the two other nodes. Here a static drone fixed at the starting location would achieve an AAS of 4.22 with $s = 6.62$.

This scenario showcases how the proposed RL algorithm allows the drone to find a high-performing trajectory that yields an AAS of 7.2 with $s = 3.48$, thus significantly outperforming the static implementation by 70.61%. In contrast, here, the SCP algorithm achieved an AAS of 6.74 with $s = 4.02$, outperforming the static implementation

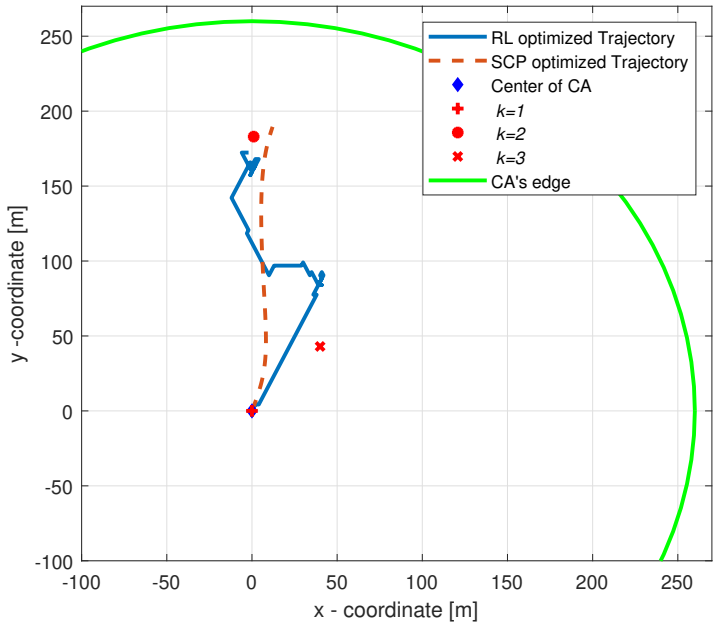


Fig. F.10: Staleness minimization solutions for **Forced** where: RL yields an AAS of 7.2 with s=3.48; SCP yields an AAS of 6.74 with s=4.02.

by only 59.71%. As the drone shuffles around after reaching some satisfactory distance from the good node, it is apparent that RL-guided trajectories could incorporate many erratic and unnecessary movements. This behavior of RL is clearly visible in this scenario due to the relative closeness of all three nodes next to the starting location.

Finally, in Fig. F.11 we can observe that the performance of both RL and SCP is very similar in treating the problem of maximizing AAS. Nonetheless, it is noticeable that RL does well at compensating the most disadvantaged node $k = 2$, particularly in the UL phase.

5.4 Key Takeaways

In a nutshell, the RL implementation is bound to be superior in comparison to the SCP approach due to the granularity of each action it can produce by its progressive decisions. Additionally, as RL needs no approximations for the environmental parameters, it is capable of discovering true optimums, if given the time and well done exploration, overfitting avoidance, and good hyper-parameter adjustment. However, to advance the RL implementation in a realistic environment, it is also beneficial to perform online learning. This would come with great energy requirements as the drone would have to

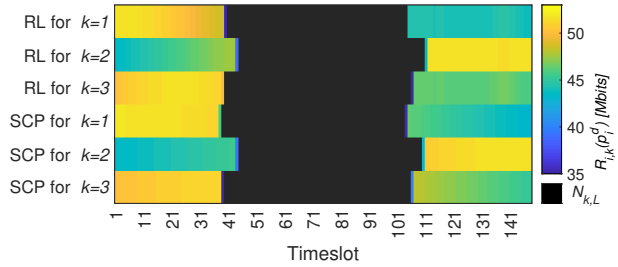


Fig. F.11: A datarate heatmap showcasing DL (left) - Learning (middle) - UL (right) schedule for both RL and SCP approaches for each device in **Forced**.

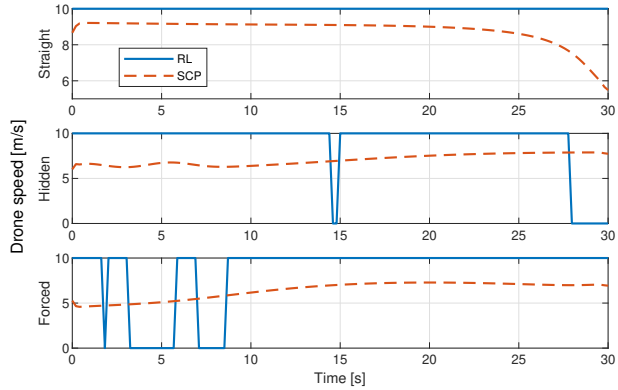


Fig. F.12: The speed distribution of the drone across time for both RL and SCP guided trajectories, for all three scenarios.

dedicate a lot of its processing capabilities to improve the implementation (although some of this could be alleviated through the offline training phase). As such, simplified approximations such as the SCP approach may be more feasible to implement in first deployments of drone orchestrators. We are optimistic that later designs of RL implementations that are guided by the SCP algorithm will provide superior performance. This combination with offline and online RL training, should be adept in finding near optimal trajectories in every stochastic, chaotic and energy limited implementations.

An important takeaway from all three scenarios is that SCP generally reacts a lot slower to the distance discrepancies mainly due to the approximations needed for converting the problem into a convex one. However, the RL approach yields many unnecessary movements that may have negative effects on the drone battery life and therefore its flight duration. For example, Fig. F.12 shows that RL provided trajectories keep the drone at full speed for the majority of the time. This, combined with our previous result analysis for Fig. F.10, we can conclude that often times this is unnecessary for the AAS

performance. The common way to address this in RL is to assign negative rewards for drone's movements. This will need to be carefully designed in order to not impair the performance of the FL network, given the long-term utility of the application.

6 Conclusion

In this paper, we have considered a drone equipped with a wireless interface in the role of an orchestrator in an FL implementation where it coordinates the transmission and learning by only adjusting its flying trajectory, and therefore, the horizontal distance to each node. Considering the total amount of learning performed across all nodes and the learning discrepancies between them as an optimization criterion, two trajectory optimization approaches were compared: deep RL and SCP. From our analysis of their potential in maximizing the combined performance metric, we have concluded that RL has shown its suitability and general superiority in solving the task of trajectory optimization for lowering *staleness* in FL networks. Nonetheless, both solutions for the drone-orchestrated FL concept outperform a static implementation with improvements in the range of 57% to 87.7%. All in all, SCP approaches are simpler and reliably provide decent performance in all scenarios without the need of hardware accelerated computing. As such, both approaches are important in transitioning towards superior FL implementations for scattered networks. However, RL implementations are necessary when encountering realistic wireless channel, weather and user mobility conditions. This inspires future works where an RL agent goes trough the process of pre-training with SCP approximated trajectories, and is then transferred for learning in the real world environments.

Acknowledgement

The work was supported by the European Union's research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 812991 "PAINLESS" within the Horizon 2020 Program. The research was also supported by the U.S. National Science Foundation under Grant CNS-1836802.

References

[1] A. Fotouhi, H. Qiang, M. Ding, M. Hassan, L. G. Giordano, A. Garcia-Rodriguez, and J. Yuan, "Survey on UAV Cellular Communications: Practical Aspects, Standardization Advancements, Regulation, and Security Challenges," *IEEE Communications Surveys Tutorials*, vol. 21, no. 4, pp. 3417–3442, Mar. , 2019.

[2] M. Mozaffari, W. Saad, M. Bennis, Y. Nam, and M. Debbah, "A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems," *IEEE Communications Surveys Tutorials*, vol. 21, no. 3, pp. 2334–2360, Mar. , 2019.

[3] W. Saad, M. Bennis, and M. Chen, "A vision of 6g wireless systems: Applications, trends, technologies, and open research problems," *IEEE network*, vol. 34, no. 3, pp. 134–142, Oct. , 2019.

[4] Z. Yang, M. Chen, K.-K. Wong, H. V. Poor, and S. Cui, "Federated learning for 6g: Applications, challenges, and opportunities," *arXiv preprint arXiv:2101.01338*.

[5] A. Fotouhi, M. Ding, and M. Hassan, "Dynamic Base Station Repositioning to Improve Performance of Drone Small Cells," in *Proc. of IEEE Globecom Workshops (GC Wkshps)*, Washington DC, Dec. , 2016, pp. 1–6.

[6] B. Galkin, J. Kibiłda, and L. A. DaSilva, "A Stochastic Model for UAV Networks Positioned Above Demand Hotspots in Urban Environments," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 7, pp. 6985–6996, Jul. 2019.

[7] M. Mozaffari, A. Taleb Zadeh Kasgari, W. Saad, M. Bennis, and M. Debbah, "Beyond 5G With UAVs: Foundations of a 3D Wireless Cellular Network," *IEEE Transactions on Wireless Communications*, vol. 18, no. 1, pp. 357–372, Jan. 2019.

[8] I. Donevski and J. J. Nielsen, "Dynamic Standalone Drone-Mounted Small Cells," in *Proc. of European Conference on Networks and Communications (EuCNC)*, Dubrovnik, Croatia, Sep. , 2020, pp. 342–347.

[9] M. Alzenad, A. El-Keyi, F. Lagum, and H. Yanikomeroglu, "3-D Placement of an Unmanned Aerial Vehicle Base Station (UAV-BS) for Energy-Efficient Maximal Coverage," *IEEE Wireless Communications Letters*, vol. 6, no. 4, pp. 434–437, Aug. , 2017.

[10] N. Babu, K. Ntougias, C. B. Papadias, and P. Popovski, "Energy Efficient Altitude Optimization of an Aerial Access Point," in *Proc. of IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications*, London, Sep. , 2020, pp. 1–7.

[11] N. Babu, C. B. Papadias, and P. Popovski, "Energy-Efficient 3-D Deployment of Aerial Access Points in a UAV Communication System," *IEEE Communications Letters*, vol. 24, no. 12, pp. 2883–2887, Aug. , 2020.

[12] Y. Zeng, J. Xu, and R. Zhang, "Energy Minimization for Wireless Communication With Rotary-Wing UAV," *IEEE Transactions on Wireless Communications*, vol. 18, no. 4, pp. 2329–2345, Apr. , 2019.

[13] C. Zhan, Y. Zeng, and R. Zhang, "Energy-Efficient Data Collection in UAV Enabled Wireless Sensor Network," *IEEE Wireless Communications Letters*, vol. 7, no. 3, pp. 328–331, Jun. , 2018.

[14] C. You and R. Zhang, "3D Trajectory Optimization in Rician Fading for UAV-Enabled Data Harvesting," *IEEE Transactions on Wireless Communications*, vol. 18, no. 6, pp. 3192–3207, Jun. , 2019.

[15] U. Challita, W. Saad, and C. Bettstetter, "Interference Management for Cellular-Connected UAVs: A Deep Reinforcement Learning Approach," *IEEE Transactions on Wireless Communications*, vol. 18, no. 4, pp. 2125–2140, Apr. , 2019.

[16] Y. Chen, X. Lin, T. Khan, and M. Mozaffari, "Efficient Drone Mobility Support Using Reinforcement Learning," in *Proc. of IEEE wireless communications and networking conference (WCNC)*, Seoul, Korea (South), May, 2020, pp. 1–6.

[17] H. Bayerlein, P. De Kerret, and D. Gesbert, "Trajectory optimization for autonomous flying base station via reinforcement learning," in *Proc. of IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, Kalamata, Jun. , 2018, pp. 1–5.

[18] S. Jeong, O. Simeone, and J. Kang, "Mobile Edge Computing via a UAV-Mounted Cloudlet: Optimization of Bit Allocation and Path Planning," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 3, pp. 2049–2063, Mar. , 2018.

[19] M. Li, N. Cheng, J. Gao, Y. Wang, L. Zhao, and X. Shen, "Energy-Efficient UAV-Assisted Mobile Edge Computing: Resource Allocation and Trajectory Optimization," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 3, pp. 3424–3438, March. , 2020.

[20] C. Pradhan, A. Li, C. She, Y. Li, and B. Vucetic, "Computation Offloading for IoT in C-RAN: Optimization and Deep Learning," *IEEE Transactions on Communications*, July. , 2020.

[21] W. Y. B. Lim, J. Huang, Z. Xiong, J. Kang, D. Niyato, X.-S. Hua, C. Leung, and C. Miao, "Towards federated learning in uav-enabled internet of vehicles: A multi-dimensional contract-matching approach," *arXiv preprint arXiv:2004.03877*, 2020.

[22] J. S. Ng, W. Y. B. Lim, H.-N. Dai, Z. Xiong, J. Huang, D. Niyato, X.-S. Hua, C. Leung, and C. Miao, "Joint auction-coalition formation framework for communication-efficient federated learning in uav-enabled internet of vehicles," *IEEE Transactions on Intelligent Transportation Systems*, Dec. , 2020.

[23] H. Zhang and L. Hanzo, "Federated learning assisted multi-uav networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 11, pp. 14 104–14 109, Sep. , 2020.

[24] T. Zeng, O. Semiari, M. Mozaffari, M. Chen, W. Saad, and M. Bennis, "Federated Learning in the Sky: Joint Power Allocation and Scheduling with UAV Swarms," in *Proc. of the IEEE International Conference on Communications (ICC), Next-Generation Networking and Internet Symposium*, Dublin, Ireland, June. , 2020, pp. 1–6.

[25] Z. Yang, M. Chen, W. Saad, C. S. Hong, and M. Shikh-Bahaei, "Energy efficient federated learning over wireless communication networks," *IEEE Transactions on Wireless Communications*, Nov. , 2020.

[26] M. Bergerman, J. Billingsley, J. Reid, and E. van Henten, "Robotics in agriculture and forestry," in *Springer handbook of robotics*. Springer, 2016, pp. 1463–1492.

[27] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated Learning: Strategies for Improving Communication Efficiency," *arXiv preprint arXiv:1610.05492*, Oct. 2016.

[28] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *arXiv preprint arXiv:1812.06127*, Dec. , 2018.

[29] S. J. Kim and G. J. Lim, "Drone-aided border surveillance with an electrification line battery charging system," *Journal of Intelligent & Robotic Systems*, vol. 92, no. 3-4, pp. 657–670, Dec. , 2018.

[30] W. Zhang, S. Gupta, X. Lian, and J. Liu, "Staleness-aware async-sgd for distributed deep learning," *arXiv preprint arXiv:1511.05950*, Nov. , 2015.

[31] C. Xie, S. Koyejo, and I. Gupta, "Asynchronous federated optimization," *arXiv preprint arXiv:1903.03934*, Mar. , 2019.

[32] U. Mohammad and S. Sorour, "Adaptive task allocation for asynchronous federated mobile edge learning," *arXiv preprint arXiv:1905.01656*, May. , 2019.

[33] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, Oct. , 2018.

[34] I. Donevski, J. J. Nielsen, and P. Popovski, "Standalone Deployment of a Dynamic Drone Cell for Wireless Connectivity of Two Services," in *Proc. of IEEE Wireless Communications and Networking Conference (WCNC)*, no. TBP, Nanjing, China , Apr. , 2021.

[35] M. Mohri, G. Sivek, and A. T. Suresh, "Agnostic federated learning," *arXiv preprint arXiv:1902.00146*, Feb. 2019.

[36] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A Joint Learning and Communications Framework for Federated Learning over Wireless Networks," *IEEE Transactions on Wireless Communications*, Oct. , 2020.

[37] L. U. Khan, S. R. Pandey, N. H. Tran, W. Saad, Z. Han, M. N. Nguyen, and C. S. Hong, "Federated Learning for Edge Networks: Resource Optimization and Incentive Mechanism," *IEEE Communications Magazine*, vol. 58, no. 10, pp. 88–93, Oct. , 2020.

[38] A. Al-Hourani, S. Kandeepan, and A. Jamalipour, "Modeling Air-to-Ground Path Loss for Low Altitude Platforms in Urban Environments," in *Proc. of IEEE Global Communications Conference*, Austin, TX, Oct. , 2014, pp. 2898–2904.

[39] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP Altitude for Maximum Coverage," *IEEE Wireless Communications Letters*, vol. 3, no. 6, pp. 569–572, Dec. , 2014.

[40] T. D. Quoc and M. Diehl, "Sequential convex programming methods for solving nonlinear optimization problems with dc constraints," *arXiv preprint arXiv:1107.5841*, Jul. , 2011.

[41] M. Grant, S. Boyd, and Y. Ye, "CVX: Matlab software for disciplined convex programming," 2009.

[42] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *nature*, vol. 518, no. 7540, pp. 529–533, Feb. , 2015.

[43] H. Van Hasselt, A. Guez, and D. Silver, "Deep Reinforcement Learning with Double Q-learning," *arXiv preprint arXiv:1509.06461*, Sep. , 2015.

[44] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, Dec. , 2014.

[45] C. She, C. Liu, T. Q. Quek, C. Yang, and Y. Li, "Ultra-Reliable and Low-Latency Communications in Unmanned Aerial Vehicle Communication Systems," *IEEE Transactions on Communications*, vol. 67, no. 5, pp. 3768–3781, May. , 2019.

[46] K. Fujii, K. Higuchi, and J. Rekimoto, "Endless Flyer: A Continuous Flying Drone with Automatic Battery Replacement," in *Proc. of IEEE 10th International Conference on Ubiquitous Intelligence and Computing*, Vietri sul Mere, Dec., 2013, pp. 216–223.

[47] I. Donevski, G. Vallero, and M. A. Marsan, "Neural Networks for Cellular Base Station Switching," in *Proc. of IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops*, Paris, France, Apr. , 2019, pp. 738–743.

Paper G

Fairness Based Energy-Efficient 3D Path Planning of a Portable Access Point: A Deep Reinforcement Learning Approach

Nithin Babu, Igor Donevski, Alvaro Valcarce, Petar Popovski, Jimmy Jessen Nielsen, Constantinos B. Papadias

The paper has been submitted in the
IEEE Open Journal of the Communications Society, 2022.

The layout has been revised.

Abstract

In this work, we optimize the 3D trajectory of an unmanned aerial vehicle (UAV)-based portable access point (PAP) that provides wireless services to a set of ground nodes (GNs). Moreover, as per the Peukert effect, we consider pragmatic non-linear battery discharge for UAV's battery. Thus, we formulate the problem in a novel manner that represents the maximization of a fairness-based energy efficiency metric and is named fair energy efficiency (FEE). The FEE metric defines a system that lays importance on both the per-user service fairness and the PAP's energy efficiency. The formulated problem takes the form of a non-convex problem with non-tractable constraints. To obtain a solution we represent the problem as a Markov Decision Process (MDP) with continuous state and action spaces. Considering the complexity of the solution space, we use the twin delayed deep deterministic policy gradient (TD3) actor-critic deep reinforcement learning (DRL) framework to learn a policy that maximizes the FEE of the system. We perform two types of RL training to exhibit the effectiveness of our approach: the first (offline) approach keeps the positions of the GNs the same throughout the training phase; the second approach generalizes the learned policy to any arrangement of GNs by changing the positions of GNs after each training episode. Numerical evaluations show that neglecting the Peukert effect overestimates the air-time of the PAP and can be addressed by optimally selecting the PAP's flying speed. Moreover, the user fairness, energy efficiency, and hence the FEE value of the system can be improved by efficiently moving the PAP above the GNs. As such, we notice massive FEE improvements over baseline scenarios of up to 88.31%, 272.34%, and 318.13% for suburban, urban, and dense urban environments, respectively.

1 Introduction

To provide seamless network connectivity, it is expected that future radio access networks implement much denser deployments of small cells, that imply very high deployment costs. A more cost-efficient solution for serving a set of ground nodes (GNs) is to use an unmanned aerial vehicle (UAV) that carries a radio access node, hereafter referred to as a portable access point (PAP) [1].

The third generation partnership project (3GPP) item [2] proposes the architecture and Quality-of-Service (QoS) requirements for such a system. The ability to have a controllable maneuver and the presence of line-of-sight (LoS) dominant air to ground channels [6] make it appropriate for applications such as data collection from wireless sensor networks (WSNs), enhancing the cellular coverage, remote sensing, emergency deployments, and so on [1]. The main drawback of the PAP system is its limited air-time which is a function of the capacity of the onboard battery unit and its power consumption profile. The air-time of a PAP is defined as the duration it remains aloft.

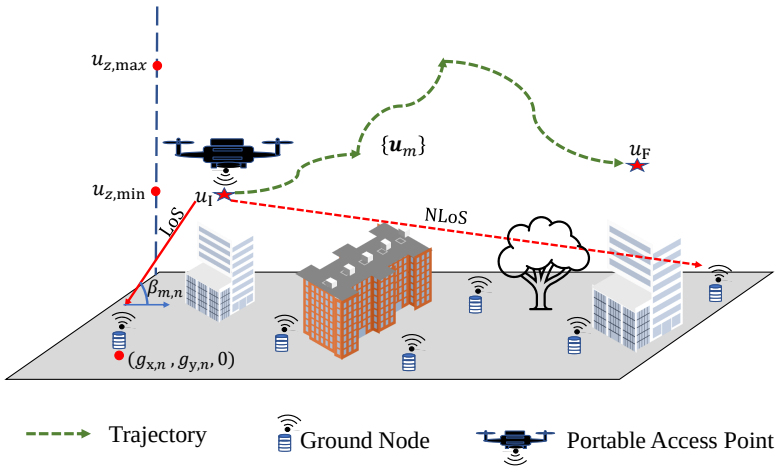


Fig. G.1: Trajectory determination scenario.

The power consumed by a PAP varies with its mode of flying; for instance, a PAP consumes the maximum amount of power when it climbs vertically up, whereas the power consumption can be the least during a horizontal flight at an certain non-zero velocity [3]. Hence, the air-time of a PAP can be increased by suitably selecting its flying mode and velocity. Moreover, the available capacity of a PAP battery unit is a non-linear function of the power-draw profile of the PAP [4]. Additionally, the air to ground channel LoS probability and the path loss between a PAP and a GN are proportional functions of the elevation angle and 3D distance between them, respectively [6]. Consequently, the trajectory of a PAP can be used as a tool to increase its air-time and improve the channel to a GN. Hence, in this work, we design a 3D trajectory for a PAP that maximizes the number of bits transmitted per Joule of energy consumed while guaranteeing a fair service to the GNs measured in terms of fair energy efficiency (FEE) of the system.

1.1 Related Works

The works in [6]- [29] consider UAV placement optimization and trajectory design problems with main objectives as maximizing coverage area, throughput, air-time, energy efficiency, and minimizing mission time, power consumption, e.t.c. The authors of [6]-[10] consider the 3D placement of UAV(s) to maximize the coverage area. In [6], the authors propose a probabilistic LoS-non-LoS (NLoS) air to ground channel model and use it to find the optimal hovering altitude of a stationary UAV that maximizes the coverage area. [7] and [8] find the optimal altitude that maximizes the coverage area

of a multi- and single-UAV system using circle packing theory. [9] uses a combination of exhaustive search and maximal weighted area algorithm to propose an optimal UAV placement method that maximizes the number of users covered, whereas [10] considers the placement optimization of a dynamic standalone drone equipped with a steerable antenna. The work in [11] proposes a power-efficient deployment of multiple UAVs which are used as aerial base stations to collect data from ground Internet of Things (IoT) devices, whereas [12] and [13] consider minimizing the total transmit power of a drone base station by considering a downlink communication scenario.

The authors of [14]- [18] consider the average throughput of a UAV-based aerial communication system as the performance metric. [14] and [15] maximize the minimum average throughput by considering an uplink communication between a set of GNs and a UAV, whereas [16], [17], [18], and [19] consider a downlink communication scenario. In [20]- [24], the authors consider the aerial vehicle's energy consumption while proposing an energy-efficient UAV(s) deployment policy. In [20], the authors propose a tractable power consumption model for a single-rotor rotary-wing UAV and use it to design a 2D trajectory that consumes the least amount of energy. In [24], we extend the model to a multi-rotor UAV and propose a 2D trajectory for a PAP that maximizes the number of bits transmitted per Joule of energy consumed while following a fly-hover-communicate protocol to serve the users. The algorithm given in [21] designs an energy-efficient 2D trajectory for a fixed-wing UAV, while in [22], we determine a set of energy-efficient hovering points using circle packing theory. The works [25]- [27] use a deep reinforcement learning (DRL) technique to design a UAV(s) placement policy that guarantees fair service to the users. [25] uses the UAV trajectory as a tool to achieve fairness in terms of learning staleness, which reflects the learning time discrepancy among the users. The proposed policies of [26] and [27] achieve fairness in terms of coverage and throughput, respectively. Comprehensive lists of works that consider placement optimization of a UAV-based system are available in [28] and [29].

1.2 Main Contributions and Paper Organization

The works in [6]- [17] propose UAV trajectory design algorithms that either maximize communication-related parameters such as the coverage area and sum or average throughput or minimize the transmit power. The works mainly design a 2D trajectory or represent the 3D optimal UAV(s) positioning problems as two subproblems that optimize the vertical and horizontal positioning of the UAV(s) recurrently. Even though the problem formulations to maximize the energy-efficiency in [20]- [24] consider the aerial vehicle's power consumption, the solutions are again 2D flight trajectories. Please note that for an energy-limited system such as a PAP, maximizing the number of bits transmitted per Joule of energy consumed while ensuring user fairness is paramount. Maximizing the throughput for a given energy budget is different from maximizing the energy efficiency since each movement of the UAV should maximize the throughput and

minimizes the energy consumption simultaneously. Moreover, a UAV consumes different power during its axial climb and forward flight modes. Neglecting this, as in [18], [19], and [26], falsely overestimates the air-time of a UAV resulting in the initiation of the early-landing procedure before completing the planned trajectory. Furthermore, [18] and [26] propose trajectory planning and resource allocation schemes for high-mobility users in which the trajectory parameters and the resources are allocated to guarantee high instantaneous throughput fairness between all users. Even though the proposed fairness metric is ideal for analyzing the performance of the considered scenario, it might be sub-optimal for an IoT application such as data collection from an IoT network. For such applications, long-term fairness metrics are more suitable. For instance, consider a scenario in which the PAP is deployed to deliver a file of a given size to all the users by the end of the trajectory. In this case, the service fairness could be measured at the end of the trajectory; if all the users are delivered with an equal amount of bits on an average by the end of the trajectory, the fairness between the users will be high.

Suppose the PAP flies near to a user in a given time instant. In that case, it is more efficient to allocate more resources to that user since the communication channel to the user, as well as the throughput, will improve. However, to guarantee a high long-term user fairness, the later segments of the trajectory should be closer to the remaining users. Finally, none of the above works consider the Peukert effect seen in Li-ion batteries that are typically used in UAVs. Neglecting the Peukert effect overestimates the air-time of the PAP, resulting in initiating the early-landing procedure before completing the planned trajectory. In practise, the PAP will be flying at different velocities resulting in different power consumption; hence the remaining air-time of the PAP varies after each action as a non-linear function of the power consumption. This affects the system's energy efficiency since the number of trajectory segments varies as a non-linear function of the power consumption profile. In essence, the 3D trajectory design of a PAP that maximizes the fairness-based energy efficiency while factoring in the UAV power consumption and the Peukert effect has, to the best of our knowledge, not yet been considered in the literature. The main contributions of this work are summarized as follows:

- We propose a method to model the non-linear Peukert effect of the PAP battery using the data points from a data sheet;
- We additionally propose an algorithm to estimate the air-time of a PAP by considering the Peukert effect and the PAP power consumption profile;
- We introduce a user fairness-based energy efficiency metric called the fair energy efficiency that considers user fairness, sum throughput, and the PAP propulsion power consumption;
- Finally, we implement a twin delayed deep deterministic policy gradient (TD3)-based 3D path planning algorithm to design a 3D trajectory for the PAP that

maximizes the FEE value of the system.

This paper is structured as follows: Section 2 explains the system setup, propagation environment, the 3D power consumption model of the PAP, and the FEE metric. In the section, we also detail the Peukert effect of the PAP battery and propose an algorithm to estimate the air-time of the PAP. Section 3 includes the problem formulation to maximize the FEE of the system and the solving methodology. Section 4 presents the main findings through numerical evaluations, and elaborates the significance of the results. Finally, Section 5 summarizes the main findings of this work. All the quantities are in SI units unless otherwise specified.

2 System Model

In this work, we consider a PAP deployed to serve a set of N GNs. Each GN $n \in \mathcal{N} = \{1, 2, 3, .., N\}$ is located at $\mathbf{g}_{\mathrm{n}} = [\mathbf{g}_{\mathrm{h},n}, 0]$ of Cartesian space (x, y, z) with $\mathbf{g}_{\mathrm{h},n} = [g_{\mathrm{x},n}, g_{\mathrm{y},n}]$, as shown in Fig. G.1. The PAP flies along a 3D path to serve the set of GNs. Both the PAP and the GNs are assumed to be equipped with omni-directional antennas.

2.1 PAP Trajectory Model

The optimal flying path of the PAP is obtained by dividing the total air time T into M time segments of length δ_t each such that $T = M\delta_t$ [20]. The value of δ_t is chosen so that within each segment the PAP can be assumed to fly with a constant velocity, and the change in path loss values between the PAP and each GN is insignificant, i.e., $\delta_t v_{\max} \leq \Delta$ where $v_{\max}$ is the maximum speed of the UAV and Δ is the maximum change in distance below which the path loss values between the PAP and each GN remain stationary. Consequently, the path of the PAP can be represented using $M + 1$ points, whose locations are denoted as $\mathbf{u}_m = [\mathbf{u}_{\mathrm{h},m}, u_{\mathrm{z},m}]$, $m \in \mathcal{M} = \{1, 2, 3, .., M + 1\}$ where $\mathbf{u}_{\mathrm{h},m} = [u_{\mathrm{x},m}, u_{\mathrm{y},m}]$ is the projection of the PAP location on the horizontal plane. The length of each segment and the maximum PAP velocity are constrained as,

$$\left\|\mathbf{u}_{m+1} - \mathbf{u}_m\right\| = \delta_t v_m \leq \delta_t v_{\max} \leq \Delta \ \forall m \in \mathcal{M}^{'}, \tag{G.1}$$

where $\mathcal{M}^{'} = \{1, 2, ..M\}$. In a particular segment, the PAP follows a time-division multiple access (TDMA) scheme to serve the GNs: let $T_{m,n}$ be the time allocated to the n^{th} GN while the PAP is flying in the m^{th} path segment with a speed of v_m m/s, then,

$$\Sigma_{n=1}^{N} T_{m,n} = \delta_t \quad \forall m \in \mathcal{M}^{'}. \tag{G.2}$$

2.2 Propagation Environment

The communication channel between the PAP and a GN at a given time can be either LoS or NLoS depending on the relative position of the GN with respect to the PAP and the blockage profile of the environment. The LoS and NLoS path loss values can be expressed as [6]- [11],

$$L_{m,n}^{\text{los}} = 20\log d_{m,n}^{\text{3D}} + 20\log f_c + 20\log\left(\frac{4\pi}{c}\right) + \eta^{\text{los}}, \tag{G.3}$$

$$L_{m,n}^{\text{nlos}} = 20\log d_{m,n}^{\text{3D}} + 20\log f_c + 20\log\left(\frac{4\pi}{c}\right) + \eta^{\text{nlos}}, \tag{G.4}$$

with $d_{m,n}^{\text{3D}} = \sqrt{{d_{m,n}^{\text{2D}}}^2 + u_{\text{z},m}^2}$ and $d_{m,n}^{\text{2D}} = \|\mathbf{u}_{\text{h},m} - \mathbf{g}_{\text{h},n}\|$. f_c and c are the carrier frequency and the velocity of light, respectively. The corresponding probability of existence of a LoS link between the PAP and the n^{th} GN while the PAP is in the m^{th} path segment can be expressed as,

$$P_{m,n}^{\text{los}} = \frac{1}{1 + a\exp\left[-b(\beta_{m,n} - a)\right]}, \tag{G.5}$$

with $\beta_{m,n} = \text{arctand}\left(\frac{u_{\text{z},m}}{d_{m,n}^{\text{2D}}}\right)$; a and b are the environment dependent parameters; η^{los} and η^{nlos} are the mean values of the respective additional path loss values due to long-term channel variations. For a given elevation angle, this additional path loss has a Gaussian distribution [5], and we use its mean value in this work [6]- [8]. The mean value depends on the building profile of the region and it is noticed that the change in the additional path loss within a particular propagation group (LoS/NLoS) is insignificant compared to the change in path loss value from one group to the other [5], [6]. This allows us to model the path loss with a constant gap between the two propagation groups as given in (G.3) and (G.4). Hence, the expected spectral efficiency to the n^{th} GN is given by,

$$\overline{R}_{m,n} = P_{m,n}^{\text{los}} R_{m,n}^{\text{los}} + (1 - P_{m,n}^{\text{los}}) R_{m,n}^{\text{nlos}}, \tag{G.6}$$

where $R_{m,n}^{x} = \log_2\left(1 + \frac{P_t}{\sigma^2 10^{L_{m,n}^{x}/10}}\right)$ $\forall x \in \{\text{los}, \text{nlos}\}$; P_t and σ^2 are the respective transmitted signal and noise power values.

2.3 UAV power consumption model

In this section, we provide the general expressions to calculate the total power consumed by the PAP during a considered time slot. The definitions and values[1] of

[1] https://dl.djicdn.com/downloads/m100/M100_User_Manual_EN.pdf

Table G.1: UAV's physical properties [24].

Label	Definition	Value
W	Weight of the UAV in Newton	24.5 N
N_{R}	Number of rotors	4
v_m	UAV's horizontal flying velocity	-
v_{tip}	Tip speed of the rotor	102 m/s
A_{f}	Fuselage area	0.038 m^2
$\rho(u_{\mathrm{z},m})$	Air density at $u_{\mathrm{z},m}$	-
C_D	Drag Co-efficient	0.9
A_r	Rotor disc area	0.06 m^2
Δ_p	Profile drag coefficient	0.002
s	Rotor solidity	0.05

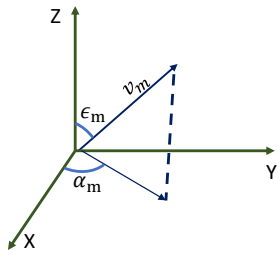


Fig. G.2: Velocity Vector.

all the variables used in this section are given in Table G.1. In the m^{th} time slot, the PAP moves from $\mathbf{u}_m$ to $\mathbf{u}_{m+1}$ in δ_t seconds. Then, as shown in Fig. G.2, the PAP velocity vector in the spherical coordinates system can be represented as $\mathbf{v}_m = (v_m, \alpha_m, \epsilon_m)$, in which $v_m = \|\mathbf{u}_{m+1} - \mathbf{u}_m\|/\delta_t$ is the speed of the PAP at which it travels from $\mathbf{u}_m$ to $\mathbf{u}_{m+1}$, $\alpha_m = \arctan\left[(u_{\mathrm{y},m+1} - u_{\mathrm{y},m})/(u_{\mathrm{x},m+1} - u_{\mathrm{x},m})\right]$ and $\epsilon_m = \arctan\left[\|\mathbf{u}_{\mathrm{h},m+1} - \mathbf{u}_{\mathrm{h},m}\|/(u_{\mathrm{z},m+1} - u_{\mathrm{z},m})\right]$ are the azimuth and elevation angles of $\mathbf{u}_{m+1}$ with respect to the axes located at $\mathbf{u}_m$. In each time slot, the PAP is in one of the following flight conditions:

Forward flight ($v_m \neq 0$, $\epsilon_m \neq 0$)

The forward flight condition contains the following PAP flying modes: 1) the PAP moves along a plane that is parallel to the horizontal plane ($v_m \neq 0$, $\epsilon_m = 90°$) commonly called as level forward flight; 2) forward (inclined) ascent or descent mode in which the PAP moves in the 3D space thereby changing all the 3 coordinates of its position ($v_m \neq 0$,

$\epsilon_m \notin \{90°, 0°\}$). The amount of power required to maintain this flight condition can be determined using [24], [3],

$$P_{\text{uav}}^{\text{fwd}}(\mathbf{v}_m) = \underbrace{N_{\text{R}} P_{\text{b}} \left(1 + \frac{3v_m^2}{v_{\text{tip}}^2}\right)}_{P_{\text{blade}}} + \underbrace{\frac{1}{2} C_D A_{\text{f}} \rho(u_{\text{z},m}) v_m^3}_{P_{\text{fuselage}}} + \underbrace{W \left[\sqrt{\left(\sqrt{\frac{W^2}{4N_{\text{R}}^2 \rho^2(u_{\text{z},m}) A_{\text{r}}^2} + \frac{v_m^4}{4}} - \frac{v_m^2}{2}\right)} + \cos\epsilon_m\right]}_{P_{\text{induce}}} \tag{G.7}$$

where $P_{\text{b}} = \frac{\Delta_p}{8} \rho(u_{\text{z},m}) s A_{\text{r}} v_{\text{tip}}^3$ and $\rho(u_{\text{z},m}) = (1 - 2.2558.10^{-5} u_{\text{z},m})^{4.2577}$. P_{blade} and P_{fuselage} are the powers required to overcome the profile drag forces of the rotor blades and the fuselage of the aerial vehicle that oppose its forward movement, respectively, while P_{induce} represents the induced power from the rotation of rotors.

Hover ($v_m = 0$)

In this mode, the PAP is static and its position is the same as that in the previous time slot. From [24], the hovering power consumption of a PAP is estimated using,

$$P_{\text{uav}}^{\text{hov}} = N_{\text{R}} P_{\text{b}} + \frac{W^{3/2}}{\sqrt{4N_{\text{R}} \rho(u_{\text{z},m}) A_{\text{r}}}}. \tag{G.8}$$

Axial climb or descent ($v_m \neq 0$, $\epsilon_m = 0$)

Here, the PAP moves along the +/− z-direction. Using (12.35) of [3], the power required by the PAP to climb vertically ($\epsilon_m = 0$) is expressed as,

$$P_{\text{uav}}^{\text{vert}}(v_m) = \frac{W}{2} \left(v_m + \sqrt{v_m^2 + \frac{2W}{N_{\text{R}} \rho(u_{\text{z},m}) A_{\text{r}}}}\right) + N_{\text{R}} P_{\text{b}}. \tag{G.9}$$

Hence, the total power consumed by the PAP while it flies along the m^{th} path segment is calculated as,

$$P_{\text{uav}}(\mathbf{v}_m) = \begin{cases} P_{\text{uav}}^{\text{fwd}}(\mathbf{v}_m) & \text{if } v_m \neq 0 \,\&\, \epsilon_m \neq 0, \\ P_{\text{uav}}^{\text{hov}} & \text{if } v_m = 0, \\ P_{\text{uav}}^{\text{vert}}(v_m) & \text{if } v_m \neq 0 \,\&\, \epsilon_m = 0. \end{cases} \tag{G.10}$$

2.4 Fair Energy Efficiency

The fair energy efficiency (bits/Joule) of the system is expressed as,

$$\mathrm{FEE}\left(\mathcal{V}_M\right) = \frac{\mathrm{FI}\left(\mathcal{V}_M\right) \sum_{m=1}^{M} \sum_{n=1}^{N} D_{m,n}\left(\mathbf{v}_m\right)}{\sum_{m=1}^{M} \delta_t P_{\mathrm{uav}}(\mathbf{v}_m)}, \tag{G.11}$$

where $D_{m,n}\left(\mathbf{v}_m\right) = BT_{m,n}\overline{R}_{m,n}$ is the number of bits transmitted to the n^{th} GN while the PAP is in the m^{th} segment, and

$$\mathrm{FI}\left(\mathcal{V}_M\right) = \frac{\left[\sum_{n=1}^{N} \overline{\mathrm{D}}_n\left(\mathcal{V}_M\right)\right]^2}{N \sum_{n=1}^{N} \overline{\mathrm{D}}_n^2\left(\mathcal{V}_M\right)}, \tag{G.12}$$

is the fairness index with $\overline{\mathrm{D}}_n\left(\mathcal{V}_M\right) = \sum_{m=1}^{M} \mathrm{D}_{m,n}\left(\mathbf{v}_m\right)/M$ giving the average number of bits transmitted to the n^{th} GN by the end of the trajectory. $\mathrm{FI}\left(\mathcal{V}_M\right) = 1$ means the PAP sends equal number of data bits to the GNs when it completes the trajectory. B is the total available bandwidth; $\mathcal{V}_M = \{\mathbf{v}_m, \forall m \in \mathcal{M}^{'}\}$. Considering the energy efficiency metric alone could allow the PAP to fly above a sub-set of GNs to maximize the energy efficiency by increasing the sum rate. Furthermore, the fairness index can be maximized either by maximizing the average number of bits transmitted to each GN or by minimizing it. The FEE metric defined in (G.11) is a weighted energy efficiency metric, where the weight is the fairness index. This forces the PAP to follow a 3D trajectory that maximizes energy efficiency and per-user fairness.

2.5 The Peukert Effect

A usual approach to estimate the maximum air-time of a PAP is to find the ratio of the initial onboard energy to the sum of instantaneous power consumption values [20]- [24]. This calculation has the fundamental assumption that the available discharge time of the PAP battery remains the same irrespective of the power-draw profile of the PAP. But, in practice, the battery discharge rate affects its available discharge time as shown in Fig. G.3, called the Peukert effect [4].

Let c_o be the rated capacity of a cell of the PAP battery unit in ampere-hours (Ah) and t_o be the rated discharge time in hours (h). This means, if the PAP draws 1A of current from the cell, the cell will be completely discharged after t_o hours. However, in practice, the current drawn by the PAP changes with time as a function of the power required and the terminal voltage of the battery:

$$P_{\mathrm{uav}}(\mathbf{v}_m) = i_m^{\mathrm{b}} \cdot n^{\mathrm{b}} \cdot V_m^{\mathrm{b}} \quad \forall m \in \mathcal{M}^{'}, \tag{G.13}$$

where n^{b} is the number of battery cells connected in series to form the battery unit of the PAP with V_m^{b}, the terminal voltage of a battery cell at the beginning of the m^{th}

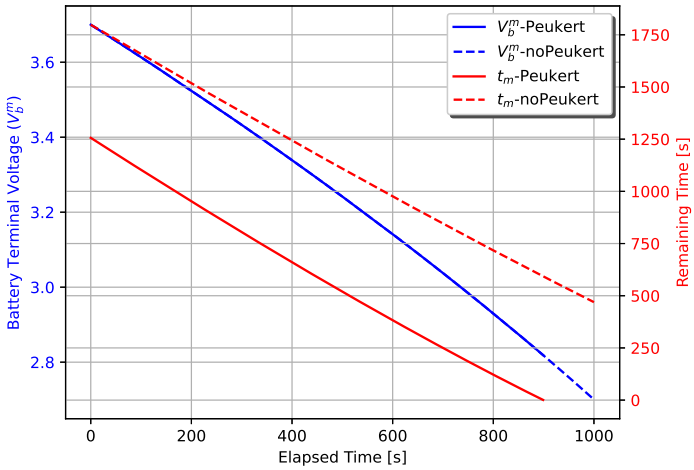


Fig. G.3: Air-time with and without considering the Peukert effect for constant 200W power-draw until the battery discharges completely.

time slot; also, $V_1^{\mathrm{b}} = V_o$ is the nominal voltage of the battery. Hence, the current drawn by the PAP during the m^{th} slot is $i_m^{\mathrm{b}} = P_{\mathrm{uav}}(\mathbf{v}_m)/(n^{\mathrm{b}} \cdot V_m^{\mathrm{b}})$. After the m^{th} slot, the battery terminal voltage drops according to,

$$V_{m+1}^{\mathrm{b}} = V_m^{\mathrm{b}} - s_m^{\mathrm{b}}(i_m^{\mathrm{b}}) \cdot i_m^{\mathrm{b}} \delta_t \ \forall m \in \mathcal{M}^{'}, \tag{G.14}$$

where $s_m^{\mathrm{b}}(i_m^{\mathrm{b}})$ is the rate of change of terminal voltage per Ah that changes as a function of i_m^{b}. In addition to the drop in the battery terminal voltage, the remaining discharge time also changes after each time slot according to,

$$t_{m+1}^{\mathrm{b}} = t_m^{\mathrm{b}} \left(\frac{c_m^{\mathrm{b}} - i_m^{\mathrm{b}} \delta_t}{i_{m+1}^{\mathrm{b}} t_m^{\mathrm{b}}} \right)^{p^{\mathrm{b}}} \quad \forall m \in \mathcal{M}^{'}, \tag{G.15}$$

with

$$t_1^{\mathrm{b}} = t_o \left(\frac{c_o}{i_1^{\mathrm{b}} t_o} \right)^{p^{\mathrm{b}}}; \tag{G.16}$$

$c_m^{\mathrm{b}} = t_m^{\mathrm{b}} i_m^{\mathrm{b}}$ with $c_1^{\mathrm{b}} = c_o$; $p^{\mathrm{b}} > 1$ is the Peukert coefficient that depends on the type of the battery used; i_{m+1}^{b} is determined by substituting the voltage determined using (G.14) in (G.13) to guarantee a power output of $P_{\mathrm{uav}}(\mathbf{v}_{m+1})$. The PAP should reach the destination either before the value of the terminal voltage reaches V_{cutoff}: $V_{M+1}^{\mathrm{b}} \geq V_{\mathrm{cutoff}}$ or $t_{M+1}^{\mathrm{b}} \geq 0$.

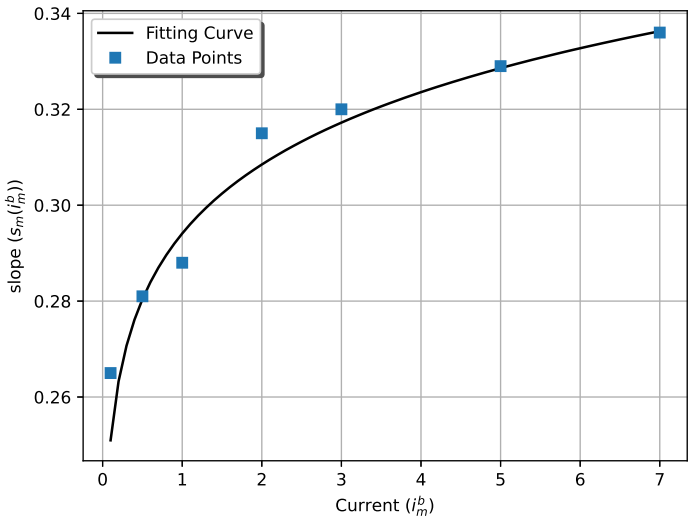


Fig. G.4: Variation of battery discharge slope for different discharge current.

The slope, $s_m^{\mathrm{b}}(i_m^{\mathrm{b}})$ depends on the type of the battery used in the PAP. For a Li-ion battery with $c_o = 4.5$ Ah, $t_o = 3$ h, $i_o = 1$ A, $V_{\mathrm{cutoff}} = 2.5$ V, and $V_o = 3.7$ V, we perform a curve fitting over the variation of $s_m^{\mathrm{b}}(i_m^{\mathrm{b}})$ with regards to i_m^{b} using the data points from the data sheet [30] as shown in Fig. G.4:

$$s_m^{\mathrm{b}}(i_m^{\mathrm{b}}) = f_1^{\mathrm{b}} \cdot {i_m^{\mathrm{b}}}^{f_2^{\mathrm{b}}}, \tag{G.17}$$

where $f_1^{\mathrm{b}} = 0.2941$, and $f_2^{\mathrm{b}} = 0.06888$.

The Peukert effect is better explained in Fig. G.3. The figure shows the voltage drop and the remaining discharge time of a typical Li-ion battery (commonly used in UAVs) during discharge when the PAP draws a power of 200 W continuously. As shown in the figure, a battery is useful until the terminal voltage or the remaining discharge time becomes lower than the corresponding threshold values (2.9 V and 30 Seconds, respectively), whichever happens first. The air-time of the PAP is defined as the time elapsed from the beginning till the battery is useful. As seen in the figure, neglecting the Peukert effect overestimates the air-time of the PAP. Thus a trajectory planned considering the availability of air-time determined without considering the Peukert effect will initiate the early-landing procedure before completing the trajectory. This affects the system's service fairness, sum rate, and energy efficiency. The PAP air-time for a given power profile considering the Peukert effect can be estimated using Algorithm 1.

Algorithm 2: PAP Air-Time Estimation

Initialize $m = 1$ $P_{\text{uav}}(\mathbf{v}_m)$, t_o, V_o, c_o, V_{cutoff}, δ_t;
while *1* **do**
  if *($m == 1$)* **then**
    $V_m^{\text{b}} = V_o$; find i_m^{b} from (G.13); find t_m^{b} using (G.16);
  if *($t_m^{\text{b}} \leq 0$)* **then**
    break;
  find $s_m^{\text{b}}(i_m^{\text{b}})$ using (G.17);
  find V_{m+1}^{b} using (G.14);
  if *($V_{m+1}^{\text{b}} < V_{cutoff}$)* **then**
    break;
  $m = m + 1$; update $P_{\text{uav}}(\mathbf{v}_m)$;
  find i_m^{b} from (G.13); t_m^{b} using (G.15);
Output: Air-Time: $T_{\text{uav}} = m\delta_t$.

3 Trajectory Optimization Using DRL Method

In this section, we formulate the problem and use the deep reinforcement learning technique to design an optimal trajectory for the PAP that maximizes the FEE of the system.

3.1 Problem Formulation

The FEE of the considered system can be increased by suitably designing the 3D trajectory of the PAP. The corresponding problem can be formulated as,

$$\begin{aligned}
&(\text{P1}): \underset{\mathcal{V}_M}{\text{maximize}} \quad \text{FEE}\left(\mathcal{V}_M\right), \\
&\text{s.t.} \quad V_{M+1}^{\text{b}} \geq V_{\text{cutoff}}; t_{M+1}^{\text{b}} \geq 0, && \text{(G.18)} \\
&\qquad u_{\text{x},m+1} = u_{\text{x},m} + \underbrace{\delta_t v_m \sin\epsilon_m \cos\alpha_m}_{\Delta_m^{\text{x}}(\mathbf{v}_m)} \; \forall m \in \mathcal{M}^{'}, && \text{(G.19)}
\end{aligned}$$

$$u_{\mathrm{y},m+1} = u_{\mathrm{y},m} + \underbrace{\delta_t v_m \sin\epsilon_m \sin\alpha_m}_{\Delta_m^{\mathrm{y}}(\mathbf{v}_m)} \forall m \in \mathcal{M}^{'}, \tag{G.20}$$

$$u_{\mathrm{z},m+1} = u_{\mathrm{z},m} + \underbrace{\delta_t v_m \cos\epsilon_m}_{\Delta_m^{\mathrm{z}}(\mathbf{v}_m)} \forall m \in \mathcal{M}^{'}, \tag{G.21}$$

$$u_{\mathrm{z,min}} \leq u_{\mathrm{z},m} \leq u_{\mathrm{z,max}} \quad \forall m \in \mathcal{M}, \tag{G.22}$$

$$\Sigma_{n=1}^{N} T_{m,n} = \delta_t \quad \forall m \in \mathcal{M}^{'}, \tag{G.23}$$

$$\mathbf{u}_{M+1} = \mathbf{u}_{\mathrm{F}}; \mathbf{u}_1 = \mathbf{u}_{\mathrm{I}}, \tag{G.24}$$

$$T_{m,n} \geq 0 \quad \forall m \in \mathcal{M}^{'}, n \in \mathcal{N}. \tag{G.25}$$

The objective function of (P1) maximizes the FEE; (G.18) ensures that the PAP will not run out of onboard available battery capacity at any point of the trajectory. The x, y, and z coordinates of the PAP position are changed according to (G.19) – (G.21), respectively. The flying region of the PAP is limited in the z-direction using (G.22) with $u_{\mathrm{z,min}}$ and $u_{\mathrm{z,max}}$ as the respective minimum and maximum permitted flying altitudes. (G.23) is the TDMA scheduling constraint. (G.24) constrains the initial and final positions of the PAP to be $\mathbf{u}_{\mathrm{I}}$ and $\mathbf{u}_{\mathrm{F}}$, respectively. (P1) is a non-convex optimization problem with a large number of optimization variables restricting the use of conventional convex optimization methods such as sequential convex programming [31]. Consequently, (P1) is equivalently represented as a Markov Decision Process (MDP) with continuous state and action spaces, and a DRL-based algorithm is proposed to design a 3D trajectory for the PAP that maximizes the FEE of the system.

The PAP is considered as an agent of the DRL framework; the framework takes the state observed by the PAP, s_m, and outputs an action, a_m. The agent receives a reward r_m after taking the action a_m that moves it from state s_m to state s_{m+1}. The whole trajectory of the PAP is considered as an episode of the DRL framework; an episode ends (i.e., $m = M+1$) if it runs out of the onboard battery capacity. It should be noted that the value of M is not constant here, and it varies according to the profile of the PAP power consumption. Additionally, to model the FEE solely as a function of the PAP trajectory, we schedule the data transmission to each GN for a time that is proportional to the respective expected spectral efficiency (i.e. $T_{m,n} = \overline{R}_{m,n} / \sum_{n=1}^{N} \overline{R}_{m,n}$, $\forall m \in \mathcal{M}^{'}, n \in \mathcal{N}$).

3.2 PAP Trajectory as an MDP

Since the next state and action of the PAP depend only on the present state of the PAP, we use a standard MDP representation as a 4-tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R})$ with sets: state space $\mathcal{S}$, action space $\mathcal{A}$, probability of transition $\mathcal{P}$, and a state-action reward map $\mathcal{S} \times \mathcal{A} -> \mathcal{R}$.

State Space, $\mathcal{S} = \{s_m\}$

The state of the PAP consists of 3D-coordinates of the PAP location written relative to the destination, the PAP's battery terminal voltage, the total energy consumed, coordinates of the GNs written relative to the horizontal projection of the PAP position, and the number of bits transmitted to each GN until the end of the m^{th} time slot:

$$s_m = \{\{(\mathbf{u}_m - \mathbf{u}_{\text{F}}), V_m^{\text{b}}, E_m, \{(\mathbf{g}_{\text{h},n} - \mathbf{u}_{\text{h},m})\}, \{D_{m,n}^{\text{sum}}\}\}, \tag{G.26}$$

where $D_{m,n}^{\text{sum}} = \sum_{j=1}^{m} D_{j,n}(\mathbf{v}_j, T_{jn})$ is the total number of bits transmitted to the n^{th} GN till the end of the m^{th} time slot; $E_m = \sum_{j=1}^{m} \delta_t P_{\text{uav}}(\mathbf{v}_j)$ is the total energy consumed until the end of the m^{th} time slot. Hence, s_m has $5 + 3 \cdot N$ dimensions.

Action Space $\mathcal{A} = \{a_m\}$

Since all the state dimensions are functions of the 3D movement of the PAP, the action a_m taken by the PAP is velocity-steered and can be expressed as a vector of dimension 3: $a_m = \{c_m^{\text{x}}, c_m^{\text{y}}, c_m^{\text{z}}\} \in [-1, 1]$ such that the components of the velocity vector are given by,

$$v_m = \sqrt{{c_m^{\text{x}}}^2 + {c_m^{\text{y}}}^2 + {c_m^{\text{z}}}^2} \cdot \frac{v_{\max}}{3}, \tag{G.27}$$

$$\epsilon_m = \arctan\left(\frac{\sqrt{{c_m^{\text{x}}}^2 + {c_m^{\text{y}}}^2}}{c_m^{\text{z}}}\right), \tag{G.28}$$

$$\alpha_m = \arctan\left(\frac{c_m^{\text{y}}}{c_m^{\text{x}}}\right). \tag{G.29}$$

Moreover, if the action takes the PAP out of the altitude boundaries, the z-coordinate of the next state is readjusted to the corresponding boundary value.

Reward Space $\mathcal{R} = \{r_m\}$

The reward function determines how fast the PAP finds the optimal trajectory. Here, the primary objectives are to maximize the FEE and let the PAP reach the specified destination before the battery becomes obsolete by satisfying all the constraints of (P1). To efficiently map the above objectives, we leverage the reward shaping technique [32]. Hence, the reward r_m is expressed as,

$$r_m = f_m + p_m, \tag{G.30}$$

where,

$$p_m = \begin{cases} \text{FEE}(\mathcal{V}_{m+1}) & \text{if FEE improves,} \\ 0 & \text{otherwise/if m=M+1,} \end{cases} \tag{G.31}$$

is the position reward that encourages the PAP to move in a direction that improves the FEE of the system, and

$$f_m = \begin{cases} \kappa_{\text{f}} \cdot \text{FEE}(\mathcal{V}_M) & \text{if} \quad m = M+1, \\ 0 & \text{else,} \end{cases} \tag{G.32}$$

is the terminal reward. The value of κ_{f} should be selected in a way that ensures the sum of the position rewards is always less than or equal to the terminal reward. κ_{f} is needed to balance the position and terminal rewards. Otherwise, the position reward would dominate over the terminal reward.

Since the defined MDP is deterministic, no randomness is considered and all transitions follow the agent's decisions [25]. Therefore, the next state is a direct consequence of the current action of the agent.

Episode Termination

The FEE of the system increases when the PAP spends the maximum time over the air to serve the GNs. Hence, an episode is terminated when the remaining air-time (T_{uav}) of the PAP is equal to the minimum time required by the PAP to reach the destination from the current position with a speed of $v_{\max}$: $T^{\min}_{\text{uav},m} = \|\mathbf{u}_m - \mathbf{u}_{\text{F}}\|/v_{\max}$. The remaining air-time of the PAP can be estimated using Algorithm 1. Consequently, if an action takes the PAP to $\mathbf{u}_{m+1}$ and if $T_{\text{uav}} < T^{\min}_{\text{uav},m+1}$, the action is discarded and the PAP moves to the destination from $\mathbf{u}_m$ with a speed of $v_{\max}$ m/s. Accordingly, the GNs are served for a maximum amount of time while ensuring a safe landing of the PAP at the destination.

The safety check explained above satisfies (G.18) by ensuring sufficient energy available at the PAP to fly back to the destination after each action. The 3D coordinates of the PAP after taking an action are determined by substituting (G.27)-(G.29) in (G.19)-(G.21), satisfying the PAP movement constraints. The altitude constraint (G.22) is satisfied by limiting the action space if such an action violates the constraint. The proposed heuristic time allocation in which the data transmission to each GN is scheduled for a time proportional to the respective expected spectral efficiency satisfies the TDMA constraints (G.23) and (G.25). Finally, all the episodes start and end at $\mathbf{u}_{\text{I}}$ and $\mathbf{u}_{\text{F}}$, respectively satisfying constraint (G.24).

3.3 TD3-Based PAP 3D Path Design

Here, the objective is to find the optimal policy π that takes the current state of the PAP (agent) and gives an action that maximizes the expected return: $R_m = \sum_{i=m}^{M} \gamma^{i-m} r_m$,

where γ is a discount factor determining the priority of short-term rewards. The action value of a state, $Q_\pi(s_m, a_m)$, gives the expected return for starting in state s_m, taking action a_m, and then acting according to the policy π forever after. The optimal action-value function is given by the Bellman equation as,

$$Q^*(s_m, a_m) = \left[r_m + \gamma \max_{a_{m+1}} Q^*(s_{m+1}, a_{m+1}) \right]. \tag{G.33}$$

In DRL, $Q^*(s_m, a_m)$ is approximated by a neural network $Q_\phi(s_m, a_m)$ with parameters

Algorithm 3: Energy Efficient 3D Path Planning

Initialize the locations of GNs;
Initialize critic networks Q_{ϕ_1}, Q_{ϕ_2}, and actor network μ_θ with random parameters ϕ_1, ϕ_2, θ;
Initialize target networks $\phi_{1,\text{tgt}} \leftarrow \phi_1$, $\phi_{2,\text{tgt}} \leftarrow \phi_2$, $\theta_{\text{tgt}} \leftarrow \theta$
Initialize replay buffer $\mathcal{H}$;
for *each episode* **do**
  Initialize the location of PAP to $\mathbf{u}_I$, $m = 1$, $V_m^b = V_o$, $t_m = t_o$; $d = 1$;
  Formulate the state of the PAP s_m;
  while *the episode is not over* **do**
    The agent takes an action with exploration noise: $a_m = \mu_\theta(s_m) + \epsilon$, observe the reward r_m and new state s_{m+1};
    store (s_m, a_m, r_m, s_{m+1}) in the replay buffer;
    if *replay buffer is sufficient* **then**
      sample mini batch of $|\mathcal{H}^{'}|$ transitions from $\mathcal{H}$;
      compute target actions using (G.41);
      compute target using (G.38);
      update critic networks by one step gradient descent using,
      $\nabla_{\phi_j} \frac{1}{|\mathcal{H}'|} \sum_{e_i \in \mathcal{H}'} \left(Q_{\phi_j}(s_i, \mu_\theta(s_i)) - y_i \right)^2$ for $j \in \{1, 2\}$;
      if *it is time to update* **then**
        update policy network by one step gradient ascent using,
        $\frac{\sum_{e_i \in \mathcal{H}'} \nabla_{\mu_\theta(s_i)} Q_{\phi_1}(s_i, \mu_\theta(s_i)) \nabla_\theta \mu_\theta(s_i)}{|\mathcal{H}'|}$;
        update target networks using, $\theta_{\text{tgt}} = \tau\theta_{\text{tgt}} + (1 - \tau\theta)$;
        $\phi_{j,\text{tgt}} = \tau\phi_{j,\text{tgt}} + (1 - \tau\phi_j)$ for $j \in \{1, 2\}$.
    $m = m + 1$;

ϕ. Then the closeness of $Q_\phi(s_m, a_m)$ to $Q^*(s_m, a_m)$ is judged by evaluating the mean-squared Bellman error (MSBE) function:

$$L(\phi, \mathcal{H}) = \underset{\{e_i\}\sim\mathcal{H}}{\mathrm{E}} \left[\left(Q_\phi(s_i, a_i) - y_i \right)^2 \right], \tag{G.34}$$

where

$$y_i = r_i + \gamma(1-d) \max_{a_{i+1}} Q_\phi(s_{i+1}, a_{i+1}), \tag{G.35}$$

is the target value and $d = 1$ represents the terminal state. The expectation in (G.34) is taken over a mini batch of experiences, $\{e_i\} = \{(s_i, a_i, r_i, s_{i+1}, d)\} = \mathcal{H}'$ sampled from the experience replay buffer, $\mathcal{H}$. The parameter ϕ is updated to minimize the MSBE. Since the considered action space is continuous, the evaluation of the MSBE is not trivial because of the $\max_{a_{i+1}} Q_\phi(s_{i+1}, a_{i+1})$ term in the target value where the maximization has to be done over a continuous action space. To tackle this, we use an actor-critic framework-based twin delayed deep deterministic policy gradient (TD3) algorithm [33]. An actor-critic framework uses an actor network that takes the state s_m as input and outputs the action a_m, whereas the Q-value of the taken action a_m at state s_m is estimated by the critic network. At the end of the training, the actor network represents the optimal policy, π. Hence, (G.34) and (G.35) can be rewritten as,

$$L(\phi, \mathcal{H}) = \underset{\{e_i\}\sim\mathcal{H}}{\mathrm{E}} \left[\left(Q_\phi(s_i, \mu_\theta(s_i)) - y_i \right)^2 \right], \tag{G.36}$$

$$y_i = r_i + \gamma(1-d) Q_\phi(s_{i+1}, \mu_\theta(s_{i+1})), \tag{G.37}$$

where μ_θ is the actor network with parameters θ and Q_ϕ is the critic network with parameters ϕ. From (G.36) and (G.37), the target y_i depends on the same parameters we are trying to train: ϕ and θ which makes the MSBE minimization unstable. The solution is to use target networks that have sets of parameters which come close to ϕ and θ, but with a time delay. The parameters of the target network are denoted as ϕ_{tgt} and θ_{tgt}, respectively. In order to avoid the overestimation problem of the deep deterministic policy gradient (DDPG) algorithm [34], the TD3 algorithm proposed in [33] uses:

Clipped Double-Q Learning

in which two critic networks are used instead of one, and uses the smaller of the two Q-values to form the targets in the MSBE functions:

$$y_i = r_i + \gamma(1-d) \min_{j=1,2} Q_{\phi_{j,\mathrm{tgt}}}(s_{i+1}, \mu_{\theta_{\mathrm{tgt}}}(s_{i+1})), \tag{G.38}$$

where $Q_{\phi_{j,\text{tgt}}}$ for $j \in \{1, 2\}$ are the corresponding target critic networks. Both networks are then trained to minimize this target:

$$L(\phi_1, \mathcal{H}) = \underset{\{e_i\}\sim\mathcal{H}}{\text{E}}\left[Q_{\phi_1}(s_i, \mu_\theta(s_i)) - y_i\right]^2, \tag{G.39}$$

$$L(\phi_2, \mathcal{H}) = \underset{\{e_i\}\sim\mathcal{H}}{\text{E}}\left[Q_{\phi_2}(s_i, \mu_\theta(s_i)) - y_i\right]^2, \tag{G.40}$$

that avoids the overestimation problem;

Delayed Policy Updates

through which the TD3 updates the policy (μ_θ) and target networks less frequently than the critic networks (once every K critic networks update);

Target Policy Smoothing

which adds a clipped noise on each dimension of the action produced by the target policy network. After adding the clipped noise, the target action is then clipped to lie in the valid action range: $[a_{\min}, a_{\max}]$,

$$\mu_{\theta_{\text{tgt}}}(s_{i+1}) = \text{clip}\left(\mu_{\theta_{\text{tgt}}}(s_{i+1}) + \text{clip}(\epsilon, -c, c), a_{\min}, a_{\max}\right), \tag{G.41}$$

where $\epsilon \sim \mathcal{N}(0, \sigma)$ and $\text{clip}(x, a, b) = \max(\min(x, b), a)$. This avoids the problem of developing an incorrect sharp peak for some actions by the Q-function approximator. The steps to design a fair energy-efficient 3D trajectory for the PAP using the TD3 framework are given in Algorithm 2.

4 Numerical Evaluation

In this section, we present our main findings obtained through numerical evaluations. The evaluations consider a square area of 1000×1000 meters with 16 GNs. The values of all the environment-related parameters are listed in Table G.2 [25]. To the best of our knowledge, the same overall setting has not been considered in the literature yet, hence we are comparing our results with the following two baseline scenarios:

1. Baseline 1: the first maneuver executed by the PAP is a diagonal climb from $\mathbf{u}_\text{I}$ up to the center of the region with coordinates $(500, 500, 100)$; hovers there until it only has sufficient energy to reach its destination; and flies reclined to the end of its trajectory $\mathbf{u}_\text{F}$;

Table G.2: Testing environment settings

Label	Definition	Value
f_c	Channel carrier frequency	5.8 GHz
c	Velocity of light	$3 \cdot 10^8$ m/s
B	Channel bandwidth for each GN	40 MHz
N_0	Noise spectral power	-174 dBm/Hz
$u_{z,\min}$	PAP's minimum flying altitude	20 m
$u_{z,\max}$	PAP's maximum flying altitude	100 m
$v_{\max}$	Maximum achievable PAP speed	24 m/s
δ_t	Time discretization Interval	1 s
P_t	Transmission Power	23 dBm

Table G.3: Network Architecture

Network(s)	Layer	Depth	Activation
Critic	Input Layer	56	–
Critic	Hidden Layer 1	256	ReLu
Critic	Hidden Layer 2	512	ReLu
Critic	Hidden Layer 3	512	ReLu
Critic	Output Layer	1	ReLu
Actor	Input Layer	53	–
Actor	Hidden Layer 1	256	ReLu
Actor	Hidden Layer 2	512	ReLu
Actor	Hidden Layer 3	512	ReLu
Actor	Output Layer	3	TanH

2. Baseline 2: the first maneuver executed by the PAP is a diagonal climb from $\mathbf{u}_\mathrm{I}$ to (200,200,100). It then continues through the shortest path between the locations of the GNs until it only has sufficient energy to reach its destination. We determine the shortest path using the well known travelling salesman algorithm.

The simulations are done considering 3 different environment scenarios namely, sub-urban, urban and dense urban with $(a, b, \eta^{\mathrm{LoS}}, \eta^{\mathrm{NLoS}})$ parameters $(4.88, 0.43, 0.2, 24)$, $(9.61, 0.16, 1.2, 23)$, and $(12.08, 0.11, 1.8, 26)$, respectively [6].

The architecture of actor and critic networks used in the simulations are listed in Table G.3. After an extensive experimentation, the values of various hyper parameters associated with the networks that give the maximum FEE value after training the networks for 1000 episodes are listed in Table G.4. Fig. G.6 plots the PAP power consumption and air-time as a function of the speed. The vertical flying power consumption

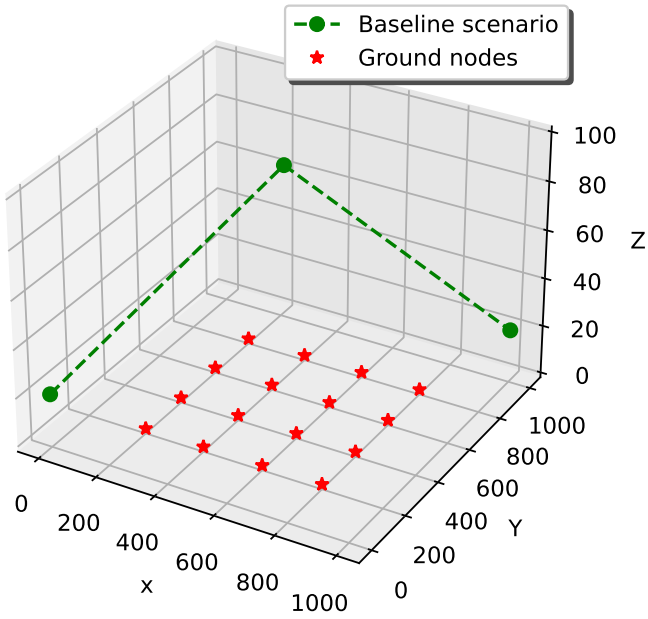


Fig. G.5: Baseline1 scenario to compare the performance; the PAP starts at (0,0,20), flies to the center of the geographical region (500,500,100), hovers there until the battery capacity reaches the threshold value, flies back to the destination (1000,1000,20).

increases with speed since the PAP requires more power to overcome the downward drag force. When the PAP is flying horizontally, the power consumption initially decreases and then increases after 11 m/s: because the magnitude of power required to overcome the rotor-induced drag force decreases with the PAP velocity; in the low-speed regime, it dominates the power consumed to overcome the fuselage and rotor profile drag forces. Correspondingly, the maximum PAP air-time, using Algorithm 1, is obtained as 1616 seconds (s) when the PAP is flying at a speed of 11 m/s. The figure also shows the importance of considering the Peukert effect during the trajectory planning of the PAP; neglecting the Peukert effect overestimates the air-time of the PAP that could force the PAP to initiate landing procedure before completing the planned trajectory.

Finally, we perform two types of RL training to showcase the effectiveness of our approach in different scenarios. The first (offline) approach assumes the same fixed user positions for both the training and testing part. This analysis is done to evaluate the capability of the actor network to solve problem where all users are uniformly spaced along the x and y directions, as shown in Fig. G.5. This fixed arrangement has very sparse GNs which makes it difficult for the FI problem. The second approach considers

Table G.4: Network Parameters.

Label	Definition	Value
α	Actor learning rate	10^{-4}
β	Critic learning rate	10^{-3}
$\lvert\mathcal{H}^{'}\rvert$	Batch size	64
$\lvert\mathcal{H}\rvert$	Replay buffer size	2×10^5
K	Network update interval	2
τ	Soft update factor	0.001
γ	Discount factor	0.99
κ_{f}	-	1000

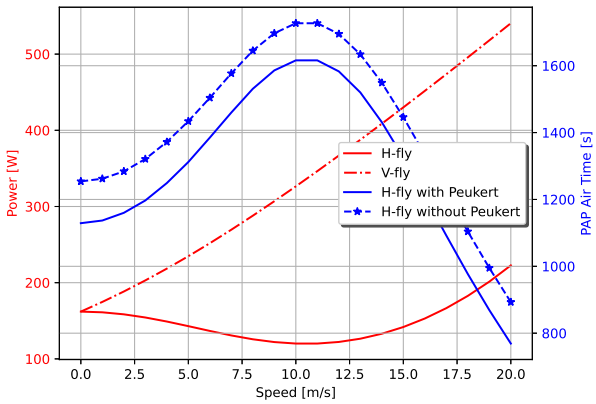
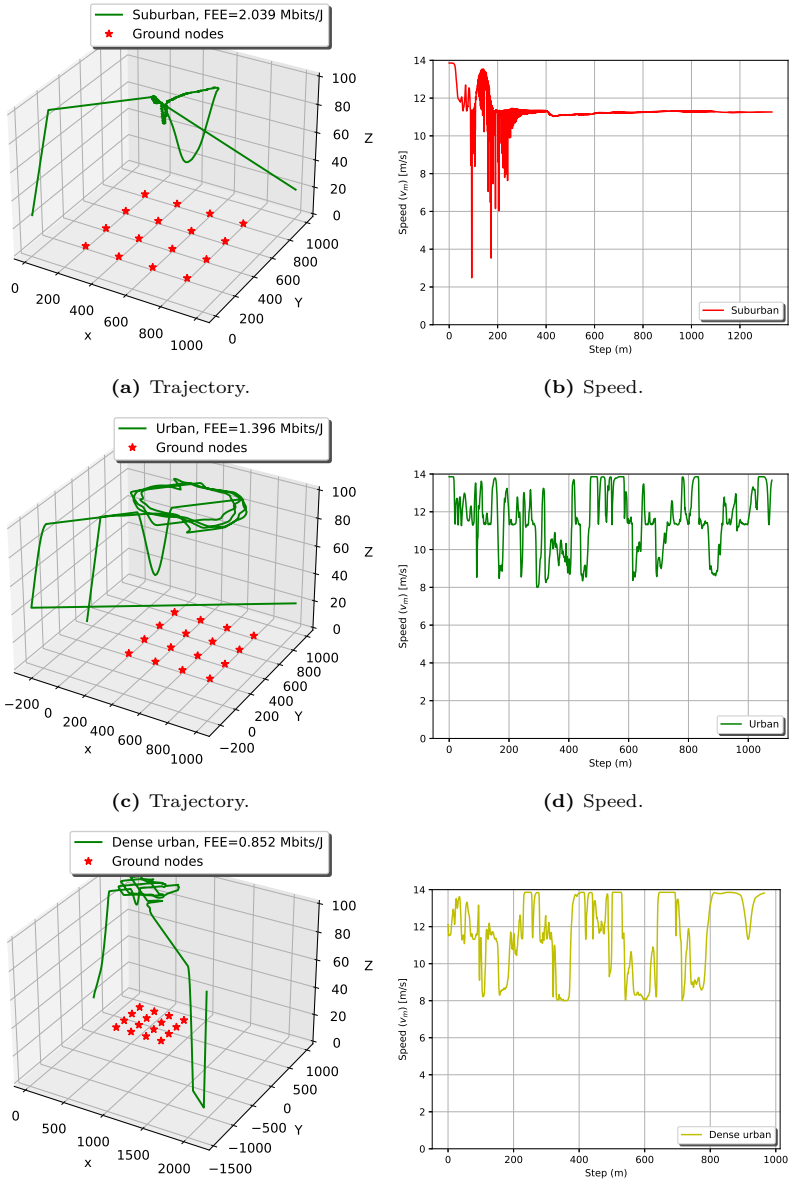


Fig. G.6: Variation of the PAP power consumption and air-time (endurance) as a function of the flying velocity. H-fly represents a level forward flight at the maximum height ($v_m \neq 0$, $\epsilon_m = 90°$); v-fly represents axial climb or descent ($v_m \neq 0$, $\epsilon_m = 0$).

random positions, such as in a point process, where the x and y coordinates of each GN are uniformly distributed for each training episode. In this approach the testing is done on a set of random GN arrangements that the agent has not used for training before, which also includes the fixed uniform positions. This is a slightly easier problem when solving fairness problems such as FI due to the likelihood of users to cluster and/or disperse due to the entropy of the system.

4.1 Fixed Uniform User Positions (Offline RL)

The resulting trajectories following the training process of the actor network for the first arrangement are given in Fig. G.7. We notice that in the case of suburban deployments, the TD3 DRL method behaves very similarly to the stop-and-hover baseline deployment.

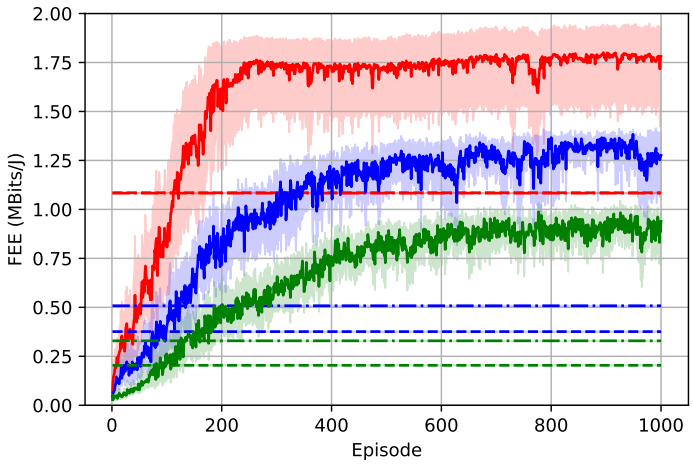


(a) Trajectory. **(b)** Speed.

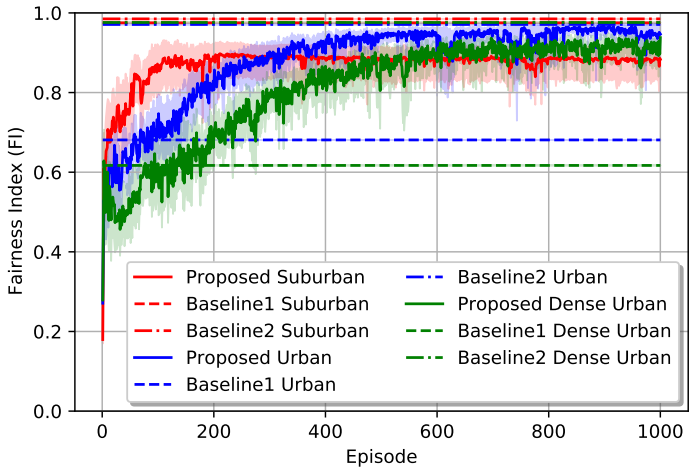
(c) Trajectory. **(d)** Speed.

(e) Trajectory. **(f)** Speed.

Fig. G.7: Sample PAP trajectories obtained using the trained actor network.

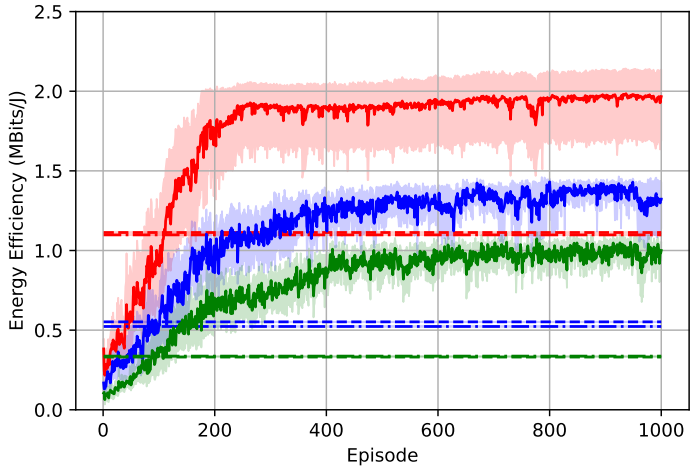


(a) Fair energy efficiency.

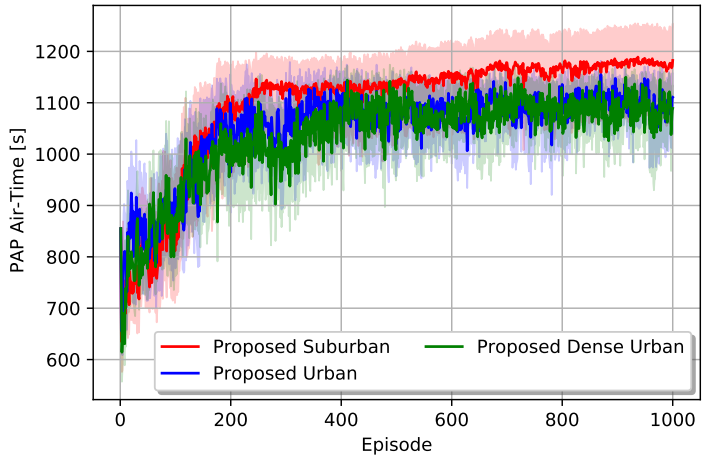


(b) Fairness index.

Fig. G.8: FEE, FI



(a) Energy-efficiency.



(b) PAP Airtime.

Fig. G.9: EE, and PAP air-time improvements as the training progresses. The training procedure is repeated for 16 different random seeds. The shadow regions around the plots show a 95% confidence interval across randomized repetitions.

Table G.5: Improvement with respect to the baselines.

		Proposed	Baseline1	Improvement	Baseline2	Improvement
Suburban	FEE	2.039	1.085	87.93 %	1.083	88.27%
	FI	0.896	0.975	-8.1 %	0.985	-9.03%
	EE	2.276	1.113	104.49%	1.098	107.28%
Urban	FEE	1.4	0.376	272.34%	0.508	175.59%
	FI	0.968	0.681	42.14 %	0.971	-0.31%
	EE	0.876	0.552	58.69 %	0.523	67.49%
Dense urban	FEE	0.853	0.204	318.13 %	0.329	159.27%
	FI	0.902	0.617	46.19 %	0.976	-7.58%
	EE	0.946	0.331	185.8 %	0.337	180.71%

This is due to the very high likelihood of having a LoS with all users and gives less relevance to the position of the PAP. However, opposite to the baseline case, the DRL method performs occasional repositioning to maintain better FEE. Finally, the TD3-DRL implementation keeps the PAP in constant movement with speeds around its most energy efficient velocity. We note that the most energy efficient velocity varies with the aerodynamics of the specific UAV and can thus be different with UAVs from different manufacturers. Opposed to the suburban scenario, in Fig. G.7 we can see in the subplots c) and e) that the PAP maintains much more dynamic movements for the urban and dense urban scenarios respectively. This is a superior approach to the stop-and-hover one, due to the short bursts of better LoS connectivity when the PAP travels above each GN. However, these bursts need to be balanced over the longer period of service and thus the PAP is always kept on the move. Finally, it is noticeable that in these two scenarios, the PAP flies off to a position that is far from the center of the area of service. This is significant for keeping the nodes equally serviced, and thus have improved FEE, in the more challenging propagation environments.

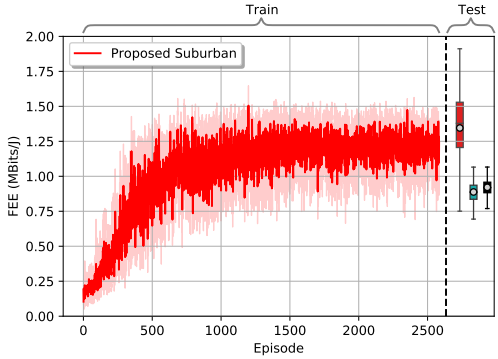
Figs. G.8a-G.9b show the respective improvement in the fair energy efficiency, fairness index, energy efficiency, and PAP air time as the training progresses. As seen in the figures, initially, the agent tries random trajectories to explore the state-action space causing relatively shorter episodes with low FEE, FI, and EE values. Later, this experience helps the ML model to converge to a better policy that improves the optimization metric. Moreover, the testing values (the values after 1000 episodes) using the proposed algorithm outperform the baseline scenarios in all the considered environments, as given in Table G.5. The maximum gain over Baseline1 is achieved when the PAP is deployed in a dense urban scenario. This is because the considered setup places a subset of GNs in the NLoS regime of the PAP's hovering point, thereby giving a low baseline FI value. The proposed algorithm improves the FI value by moving the PAP around the GNs, as shown in Fig. G.7e. The performance gain achieved by Baseline2 is competitive in urban and dense urban scenarios, with regards to Baseline1, because the corresponding trajectory improves the service fairness among all the GNs. Interestingly, the FEE

performances of Baseline1 and Baseline 2 in a suburban scenario are comparable since all the GNs are in LoS with the PAP throughout the respective placement policy, giving high FI values. This leads us to the conclusion that the suburban environment is not challenging enough for the problem of trajectory with regards to the scale of our implementation. . In all the trajectories, the PAP climbs to the maximum altitude after leaving the starting point and then follows a horizontal flight during the remaining endurance: as the altitude increases, the throughput between the PAP and a node increases due to an improved LoS probability between them; furthermore, the PAP power consumption during a vertical flight is much higher compared to a horizontal flight as shown in Fig. G.6. Hence, flying horizontally at the maximum altitude increases the PAP air time as well as the number of bits transmitted to the GNs thereby improving the FEE value of the system. Also, the speed plots of Fig. G.7 show that the actor proposes to fly the PAP at the optimal flying speed that maximizes the air-time of the PAP.

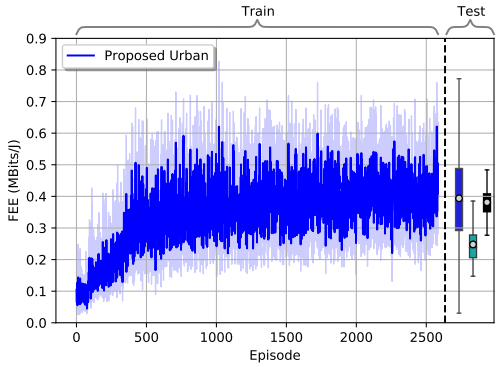
4.2 Randomized Uniform User Positions (Online RL)

In this section, we describe the method adopted from [35] to generalize the training so that the trained actor network performs well for any set of user positions. The system is trained for a fixed number of episodes N_{train}, where the x and y coordinates of each GN are uniformly distributed for each training episode. After every 10 training episodes, we evaluate the actor network on a total of N_{eval} evaluation episodes with disabled learning to assess the current performance of the actor network. We repeat this training procedure for N_{seed} times, each with a different random seed. The average FEE value after each evaluation phase is used as a metric to select the best-performing parameter (θ) of the actor network (μ_θ). At the end of the training procedure, we further evaluate the learned policy by assessing its performance in N_{test} episodes, each with a different placement of GNs that the agent has not seen during the entire training process. Moreover, this test phase happens without exploration and learning.

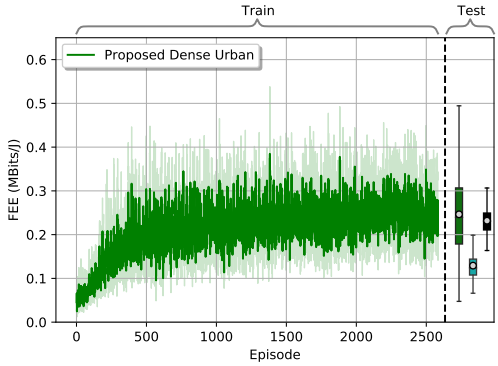
Fig. G.10 shows the training and testing performances for suburban, urban, and dense urban scenarios with $N_{\text{train}} = 1000$, $N_{\text{eval}} = 16$, and $N_{\text{seed}} = 8$. The shadow regions around the plots show a 95% confidence interval across randomized repetitions. In all the scenarios, the FEE value improves with the training as seen in Fig. G.8a. The mean and median FEE values obtained after testing the learned policy over $N_{\text{test}} = 512$ episodes outperform the mean baseline performances. As observed previously, the maximum and minimum performance gains are observed in dense urban and suburban scenarios, respectively. Additionally, the test performances (with learning disabled) are comparable with the performances at the end of the training phase. Thus the learned policy can be used to design an energy-efficient 3D trajectory for a PAP deployed to serve any given distribution of the GNs while guaranteeing per-user service fairness.



(a) FEE Suburban



(b) FEE Urban



(c) FEE Dense urban

Fig. G.10: Training and testing FEE plots. Circles inside the box plots represent the average test FEE values. The second and third box plots represent Baselines 1 and 2, respectively.

5 Conclusion

In this paper, we considered a UAV in the role of a portable access point (PAP) that aims to maximize the novel fairness-based energy efficiency metric, fair energy efficiency (FEE). Optimizing the energy-efficiency of PAPs is important but this should not come at the expense of service fairness. The method we propose here strikes a good balance between both. Moreover, we defined a pragmatic non-linear discharge behavior of the PAP battery, as the Peukert effect. As the first work to investigate the Peukert effect in PAP 3D trajectory optimization for wireless IoT services, we initially investigated the impact of the non-linearity of the energy storage. As such, we deducted that neglecting the Peukert effect overestimates the PAP air time which could force the PAP to perform an early landing. Given the non-convex FEE maximization problem with non-tractable constraints we proposed an adapted implementation of a twin delayed deep deterministic policy gradient deep reinforcement learning (TD3-DRL) framework. The optimal solutions provided by TD3-DRL varied by the properties of the propagation environment. The improvements of using the TD3-DRL in suburban scenarios are moderate with a gain up to 80% in suburban over the baseline scenarios and around 200% and 300% in the urban-and dense urban scenarios respectively. Finally, we generalize the network to any set of GN positions. Thus, we can summarize, that our TD3-DRL implementation provides a robust solution for PAP trajectory optimization in both strongly LoS and strongly NLoS environments.

References

[1] Fotouhi, Azade, et al. “Survey on UAV Cellular Communications: Practical Aspects, Standardization Advancements, Regulation, and Security Challenges,” in *IEEE Commun. Surv. and Tut. 21.4 (2019)*: 3417-3442.

[2] “Study on Enhanced LTE support for Aerial Vehicles (Release 15),” 3GPP, Sophia Antipolis, France, Rep. 3GPP TR 36.777, Dec. 2017.

[3] Filippone A, “Flight Performance of Fixed and Rotary Wing Aircraft,” in *Elsevier*; 2006 May 10.

[4] Khan, K. A., et al. “PKL Electrochemical Cell and the Peukert’s Law,” in *IJARIIE* 4.2 (2018): 4219-4227.

[5] A. Al-Hourani, S. Kandeepan, and A. Jamalipour, “Modeling Air-to-Ground Path Loss for Low Altitude Platforms in Urban Environments,” in *Globecom 2014, Symposium on Selected Areas in Communications: Satellite & Space Communication,* Austin, USA, Dec. 2014.

[6] A. Al-Hourani, S. Kandeepan and S. Lardner, "Optimal LAP Altitude for Maximum Coverage," in *IEEE Wireless Commun. Lett.,* vol. 3, no. 6, pp. 569-572, Dec. 2014

[7] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Efficient Deployment of Multiple Unmanned Aerial Vehicles for Optimal Wireless Coverage," in *IEEE Commun. Lett.,* vol. 20, no. 8, pp. 1647–1650, Aug. 2016.

[8] M. Alzenad, A. El-Keyi, F. Lagum, and H. Yanikomeroglu, "3-D Placement of an Unmanned Aerial Vehicle Base Station (UAV-BS) for Energy Efficient Maximal Coverage," in *IEEE Wireless Commun. Lett.,* vol. 6, no. 4, pp. 434–437, Aug. 2017.

[9] M. Alzenad, A. El-Keyi and H. Yanikomeroglu, "3-D Placement of an Unmanned Aerial Vehicle Base Station for Maximum Coverage of Users With Different QoS Requirements," in *IEEE Wireless Commun. Lett.,* vol. 7, no. 1, pp. 38-41, 2018.

[10] I. Donevski and J. J. Nielsen, "Dynamic Standalone Drone-Mounted Small Cells," in *2020 European Conference on Networks and Communications (EuCNC),* 2020, pp. 342-347.

[11] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Mobile Unmanned Aerial Vehicles (UAVs) for Energy-Efficient Internet of Things Communications," in *IEEE Trans. Wireless Commun.,* vol. 16, no. 11, pp. 7574–7589,Nov. 2017.

[12] J. Cui, H. Shakhatreh, B. Hu, S. Chen and C. Wang, "Power-Efficient Deployment of a UAV for Emergency Indoor Wireless Coverage," in *IEEE Access,* vol. 6, pp. 73200-73209, 2018.

[13] Q. Zhang, M. Mozaffari, W. Saad, M. Bennis and M. Debbah, "Machine Learning for Predictive On-Demand Deployment of Uavs for Wireless Communications," in *2018 IEEE Global Communications Conference (GLOBECOM),* 2018, pp. 1-6.

[14] S. Eom, H. Lee, J. Park, and I. Lee, "UAV-Aided Wireless Communication Designs with Propulsion Energy Limitations," in *IEEE Trans. on Veh. Technol.*, 2019.

[15] C. You and R. Zhang, "Hybrid Offline-Online Design for UAV-Enabled Data Harvesting in Probabilistic LoS Channels," in *IEEE Trans. on Wireless Commun.,* vol. 19, no. 6, pp. 3753-3768, June 2020.

[16] I. Valiulahi and C. Masouros, "Multi-UAV Deployment for Throughput Maximization in the Presence of Co-Channel Interference," in *IEEE Internet of Things Journal,* vol. 8, no. 5, pp. 3605-3618, Mar., 2021.

[17] Wu, Qingqing, and Rui Zhang, "Common Throughput Maximization in UAV-Enabled OFDMA Systems with Delay Consideration," in *IEEE Trans. on Commun. 66.12 (2018)*: 6614-6627.

[18] W. Shi, J. Li, H. Wu, C. Zhou, N. Cheng and X. Shen, "Drone-Cell Trajectory Planning and Resource Allocation for Highly Mobile Networks: A Hierarchical DRL Approach," in *IEEE Internet of Things Journal,* vol. 8, no. 12, pp. 9800-9813, June 15, 2021.

[19] Wu, H., Lyu, F., Zhou, C., Chen, J., Wang, L., & Shen, X., "Optimal UAV Caching and Trajectory in Aerial-Assisted Vehicular Networks: A Learning-Based Approach," in *IEEE Journal on Selected Areas in Commun.,* 38(12), 2783-2797, 2020.

[20] Y. Zeng, J. Xu and R. Zhang, "Energy Minimization for Wireless Communication with Rotary-Wing UAV," in *IEEE Trans. on Wireless Commun.*, vol. 18, no. 4, pp. 2329-2345, April 2019.

[21] Y. Zeng and R. Zhang, "Energy-Efficient UAV Communication With Trajectory Optimization," in *IEEE Trans. on Wireless Commun.*, vol. 16, no. 6, pp. 3747-3760, June 2017.

[22] N. Babu, C. B. Papadias and P. Popovski, "Energy-Efficient 3D Deployment of Aerial Access Points in a UAV Communication System," in *IEEE Commun. Lett.*, vol. 24, no. 12, pp. 2883-2887, Dec. 2020.

[23] S. Shakoor, Z. Kaleem, M. I. Baig, O. Chughtai, T. Q. Duong and L. D. Nguyen, "Role of UAVs in Public Safety Communications: Energy Efficiency Perspective," in *IEEE Access,* vol. 7, pp. 140665-140679, 2019.

[24] N. Babu, M. Virgili, C. B. Papadias, P. Popovski and A. J. Forsyth, "Cost- and Energy-Efficient Aerial Communication Networks With Interleaved Hovering and Flying," in *IEEE Trans. on Veh. Technol.,* vol. 70, no. 9, pp. 9077-9087, Sept. 2021.

[25] I. Donevski, N. Babu, J. J. Nielsen, P. Popovski and W. Saad, "Federated Learning With a Drone Orchestrator: Path Planning for Minimized Staleness," in *IEEE Open Journal of the Commun. Society,* vol. 2, pp. 1000-1014, 2021.

[26] C. H. Liu, Z. Chen, J. Tang, J. Xu and C. Piao, "Energy-Efficient UAV Control for Effective and Fair Communication Coverage: A Deep Reinforcement Learning Approach," in *IEEE Journal on Selected Areas in Commun.,* vol. 36, no. 9, pp. 2059-2070, Sept. 2018.

[27] R. Ding, F. Gao and X. S. Shen, "3D UAV Trajectory Design and Frequency Band Allocation for Energy-Efficient and Fair Communication: A Deep Reinforcement Learning Approach," in *IEEE Trans. on Wireless Commun.,* vol. 19, no. 12, pp. 7796-7809, Dec. 2020.

[28] Mazaherifar A, Mostafavi S., "UAV Placement and Trajectory Design Optimization: A Survey," in *Wireless Personal Commun.,* 2021 Dec 1:1-20.

[29] S. Shakoor, Z. Kaleem, M. I. Baig, O. Chughtai, T. Q. Duong and L. D. Nguyen, "Role of UAVs in Public Safety Communications: Energy Efficiency Perspective," in *IEEE Access,* vol. 7, pp. 140665-140679, 2019.

[30] https://files.gwl.eu/inc/_doc/attach/StoItem/5230/Datasheet_LG_MH1.pdf

[31] B. R. Marks and G. P. Wright, "Technical Note- A General Inner Approximation Algorithm for Nonconvex Mathematical programs," in *Oper.Res.,* vol. 26, no. 4, pp. 681–683, Aug. 1978.

[32] Wang Z, Li H, Wu Z, Wu H, "A Pretrained Proximal Policy Optimization Algorithm with Reward Shaping for Aircraft Guidance to a Moving Destination in Three-Dimensional Continuous Space," in *International Journal of Advanced Robotic Systems,* January 2021.

[33] Fujimoto, Scott, Herke Hoof, and David Meger, "Addressing Function Approximation Error in Actor-Critic Methods," in *International Conference on Machine Learning, PMLR,* 2018.

[34] Lillicrap, Timothy P., et al. "Continuous Control with Deep Reinforcement Learning," in *arXiv preprint arXiv:1509.02971 (2015).*

[35] M. P. Mota, A. Valcarce, J. -M. Gorce and J. Hoydis, "The Emergence of Wireless MAC Protocols with Multi-Agent Reinforcement Learning," in *2021 IEEE Globecom Workshops (GC Wkshps),* 2021, pp. 1-6

Paper H

Sustainable Wireless Services with UAV Swarms Tailored to Renewable Energy Sources

Igor Donevski, Marco Virgili, Nithin Babu, Jimmy Jessen Nielsen, Andrew J. Forsyth, Constantinos B. Papadias, Petar Popovski

The paper has been submitted in the
IEEE Transactions on Smart Grid, 2022.

The layout has been revised.

Abstract

Unmanned Aerial Vehicle (UAV) swarms are often required in off-grid scenarios, such as disaster-struck, war-torn or rural areas, where the UAVs have no access to the power grid and instead rely on renewable energy. Considering a main battery fed from two renewable sources, wind and solar, we scale such a system based on the financial budget, environmental characteristics, and seasonal variations. Interestingly, the energy source is correlated with the energy expenditure of the UAVs, since strong winds cause UAV hovering to become increasingly laborious. The aim is to maximize the cost efficiency of coverage at a particular location, which is a combinatorial optimization problem for dimensioning of the multivariate energy generation system under non-convex criteria. We have devised a customized algorithm by lowering the processing complexity and reducing the solution space through sampling. Evaluation is done with condensed real-world data on wind, solar energy as well as traffic load per unit area, driven by vendor provided prices. The implementation was tested in four locations, with varying wind or solar intensity. The best results were achieved in locations with mild wind presence and strong solar irradiation, while locations with strong winds and low solar intensity require higher CAPEX allocation.

1 Introduction

The use of UAVs, in particular the multi-copter drones, has been praised for the ability of providing modular, adaptable and scalable wireless communications services as they can easily be redeployed, target specific users and load balance existing cellular architectures, [1, 2]. Unfortunately, UAV-mounted small base stations (UAVSBSs) are not a feasible replacement to traditional base stations in urban areas, mainly due to the safety, privacy and noise concerns. Opposed to this, UAVSBSs are crucial in scenarios that result in service outages such as war-torn or disaster struck areas [3] and traffic surges in weakly serviced areas [4]. In these cases, it should be expected that the existing infrastructure is unable to support the energy requirements of the UAVSBS system. Moreover, to satisfy the service constraints of the area, that generally vary during the day [5], the deployments require multiple UAVs (a.k.a. swarm).

Since UAVs require a lot of energy (stored in a battery) to fly, the goal of this work is to evaluate the feasibility of self-sustainable energy implementation for long-term persistent (uninterrupted) operation of UAV swarms. This is targeted for areas where fixed infrastructure is unavailable and a UAVBS-based solution is deemed acceptable. As such we aim to provide a solution to finding the ideal scale of the system, for a particular location, through maximizing the coverage area discounted by its financial cost. This is a nuanced problem as it solves complex interactions between the energy generation and consumption systems.

1.1 Literature Overview

The effects of using UAVSBSs that are positioned to offer service to ground customers has already been well investigated in [2, 6–13]. In [6–8] the focus is on improving spectral efficiency when exploiting the temporal and spatial mobility of UAVs for servicing user hotspots. In our previous works [9, 10], we demonstrated the benefit of horizontally positioning a standalone UAVSBS, equipped with a tilting directional antenna. Moreover, the work in [11] focused on the energy efficiency for UAVSBS deployment, while the authors in [14] and [15] studied the problem of placement optimization of a single cell and interference-limited multi UAVSBS deployments, respectively. While the aforementioned works are concerned with optimizing deployment locations of the UAVs once they are in the air, they generally ignore the problem of short service durations.

Ever since the proliferation of drones into the mass market, there has been a strive towards persistent UAV services [16] with several methods. The most prominent method assumes automated battery swapping [17]. In [18] the authors solve the optimal trajectory for patrolling UAVs that exactly exploit the battery swapping mechanism that is connected to the grid mains. In [19] the authors consider a mothership-like UAV that houses and orchestrates the deployment of a swarm of smaller UAVs, where the mothership ensures that the energy requirements for the entire system is satisfied. While such mothership systems are genuinely useful for achieving unlimited mobility, the creation of one is complex and assume technical innovation on several fronts which is a significant shortcoming and would become very costly to implement. On the other hand, the authors of [20] consider a ground based central unit that serves as a backbone to the UAVs and has solar panels to manage the energy requirements on the ground. The shortcoming of the previous work is that it does not consider the impact of wind and offsetting the influence of air movement. In [21] the authors propose a cost efficient UAV system for data harvesting from IoT systems, and is not directly related to general communication services. In [22] we previously investigated the optimal arrangement for UAVs that need to provide persistent service by interleaved recharging at a ground station, however the analysis was limited to a single UAV and without the impact of wind.

We note that per [2], UAV Base Stations are able to alleviate capital and operating expenditures (CAPEX & OPEX) of telecom operators up to 52% and 42% respectively. To serve disaster struck, remote, or underdeveloped areas, we focus on works that involve dimensioning sustainable energy generation systems for wireless communications. As such, the authors of [23] and [24] proposed alleviating the energy requirements of multi-tier cellular implementations supported with renewable energy. The work of [25] comes the closest to our goal of providing cellular connectivity in rural zones. Moreover, the authors consider an architecture composed of UAV-based BSs to provide cellular coverage, ground sites to connect the UAVs with the rest of the network, solar panels, and batteries to recharge the UAVs. The [25] approach is generally simplistic, does not maximize coverage, and does not account for the impact of wind. And finally, the

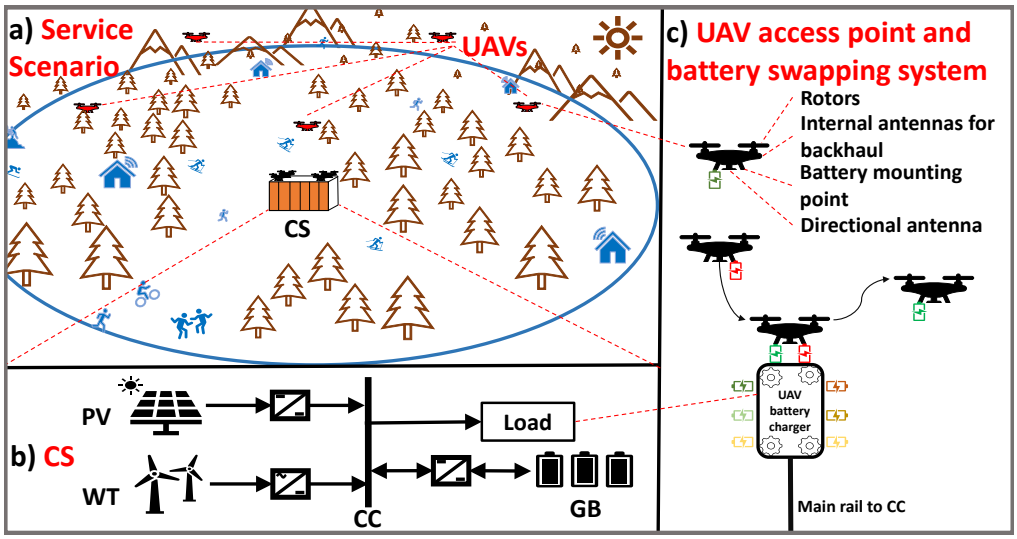


Fig. H.1: Schematic of the energy system at the central station (CS) that consists of wind turbines (WT), photovoltaic (PV) panels, a central circuit (CC), ground battery (GB) and a UAV battery charger (UAVBC) that represents the load. The service scenario illustrates a deployment in a mountainous region.

work [26] analyzes a mothership-orchestrated UAV swarm for wireless communications, with the goal is to minimize the overall weighted distance traveled by the mothership for UAV recharge. This work not give realistic overview of the capital expenses, and omits the impact of wind.

1.2 Off-grid Redeployable UAV Communications System

The proposed communications system is shown in Fig. H.1, and is intended to provide persistent services to rural, suburban and low-rise urban areas, by deploying a central station (CS) that supports and coordinates the UAV swarm. Once deployed, the UAVs hover and provide a satisfactory service rate for the entire area. When a UAV nearly depletes its battery, it flies back to the CS to replace the energy spent. As in [27], we consider an automatic battery swapping mechanism that replaces the depleted batteries, as shown in c) in Fig. H.1.

To ensure a long-term persistent deployment, the system needs to compensate the power requirements. As Illustrated in Fig. H.1, our considered implementation of a CS has five energy modules. The wind turbine (WT) and photovoltaic (PV) panels generate energy to be stored into a central ground battery (GB). The system is interconnected by a central circuit (CC) module that directly links the load of the system, which is an automatic UAV battery charger (UAVBC) that charges the hot-swappable UAV

batteries. Finally, we assume that the CS acts as a sink/middlehaul for the wireless service that the UAVs offer, which is overprovisioned and provided by low earth orbit (LEO) satellites [28].

1.3 Main Contributions & Paper Outline

This work provides a fresh perspective on UAV swarm implementations for persistent wireless service such as:

- We consider long-term standalone deployment of UAVs for remote areas with a realistic model for the impact of wind onto the energy consumption of the UAVs. This is the first work in the area to consider the wind intensity giving importance to the altitude and terrain roughness.
- We model the system as totally self-sustainable and including PV- and WT-based energy generation modules, where this is the first work to introduce WTs that have the capability of offsetting the UAV's energy expenditure due to wind strength.
- We aim to provide coverage in the most efficient manner for a potential capital investment. To achieve this, we formulate a novel problem that maximizes the wireless coverage area discounted by the cost of the system. The goal of solving the problem as coverage maximization, is to understand the scalability of such a complex system in the many different possible deployment environments. This is nuanced multi-variate optimization problem and is entirely based on real world data and current commercial climate.
- We propose a computationally light algorithm that uses greedy sampling and binary search to find the optimal configuration that is a combination of area covered, wind turbines, PV panels, cells in the ground battery, and UAVs in the swarm.

Common abbreviations are contained in Tab. H.1, and commonly used symbols are contained in Tab. H.2. Symbols introduced later in the papers are contained in a table within that section. The rest of the paper is organized as follows. In Section 2 we describe the UAV based communications services and its energy expenditure. In Section 3 we introduce the energy generation and management system. In Section 4 we define both the formal problem and the proposed algorithmic solution. In Section 5 we display the results of the implementation, and finally draw the conclusions in Section 6.

2 Modeling UAV Service and Energy

The coverage area (CA), a circle with radius $D_{\max}$, contains an arbitrary number of users that we model in terms of zonal datarate density (ZDD). The ZDD is defined as

Table H.1: Abbreviations used in this work.

Abbreviation	Meaning
BS	Base Station
CA	Coverage Area
CC	Central Circuit
CCEE	Cheapest Combination of Energy Elements
CS	Central Station
GSS	Greedy and Sparse Search
IoT	Internet-of-Things
LoS	Line-of-Sight
MEL	Minimum Energy Load
MPPT	Maximum Power Point Tracker
NLOS	No-Line-of-Sight
NOC	Nominal Operating Conditions
PV	Photovoltaic
QoS	Quality of Service
ST	Standard Testing conditions
UAV	Unmanned Aerial Vehicle
UAVBC	UAV Battery Charger
UAVSBS	UAV-mounted Small Base Station
WT	Wind Turbine
ZDD	Zonal Datarate Density
CAPEX	Capital Expenditure

λ_h which represents the requested datarate per unit of area, in Mbps/m^2, for hour of the day $h = 1, 2...24$, that is uniform for the entire area. The goal of ZDD is to properly scale it for larger time-lengths in the order of hours and adapt it per type of residency area, such as in [5, 29]. This allows scaling the traffic demand for different sizes of $D_{\max}$, without having to assume a stochastic point process. As a result, the minimum datarate requested for the entire CA $R_{h,\min}$, for hour h, is: $R_{h,\min}(D_{\max}) = \lambda_h \pi D_{\max}^2$. Note that we do not consider different rates between days of the year $i = 1, 2, ..365$, since such a metric is difficult to obtain and challenges the privacy of users. Considering a fleet of available drones n_{UAV}, a swarm size of $k_h \leq n_{\text{UAV}}$ UAVs is released so that each UAV j is given an equal amount of area to serve with rate $R(k_h, D_{\max})$. We can thus linearly scale the traffic load on each UAV with the swarm size, so that it satisfies:

$$\frac{R_{h,\min}(D_{\max})}{k_h} \leq R(k_h, D_{\max}),\ \forall h \tag{H.1}$$

under the condition that $\max_h (R_{h,\min}) \leq n_{\text{UAV}} \cdot R(n_{\text{UAV}}, D_{\max})$ is satisfied. The data rate depends on k_h and $D_{\max}$ because the radius of coverage of each UAV in the swarm

Table H.2: Nomenclature of the symbols used in this work.

Symbol	Meaning and unit of measurement
A_{eff}	Antennas' effectiveness in fitting the CA
C_{bat}	Battery capacity (Wh)
D_j	Horizontal Distance of the j-th UAV from CS (m)
$D_{\max}$	Diameter of coverage area (m)
$E_{\text{UAVs},h,i}$	UAV swarm energy consumption at time h and day i (Wh)
EEAC	Energy Efficiency of Annual Coverage (m^2/Wh)
F	Total cost of the system (€)
F_a	Cost of each type of wind turbine (€)
F_{E}	Cost of battery system (€)
F_{PV}	Cost of photovoltaic system (€)
F_{UAV}	Cost of UAVs (€)
F_{W}	Cost of wind system (€)
h	Hour of the day
i	Day of the year
k_h	Number of UAVs in a swarm
ℓ	Path loss (dB)
n_a	Number of wind turbines of each type
n_{PV}	Number of solar panels
n_{UAV}	Fleet size - Number of available UAVs
$\mathbf{p}_j$	Hovering location of the j-th UAV
R	Instantaneous Data rate (Mbps)
$R_{h,\min}$	Minimum datarate requirement (Mbps)
$v_{h,i}^{\text{wind}}$	Wind speed at time h and day i (m/s)
η	Mean large scale fading coefficient (dB)
θ	Elevation angle at the cell edge (°)
λ_h	Zonal datarate density (Mbps/m^2)
τ_{fly}	Air-time of a UAV (% of hour)

varies within the bounds of $0 < D(k_h, D_{\max}) \leq D_{\max}$.

2.1 UAV Hovering Locations

The coverage region for each UAV in the swarm is a circle of radius $D(k_h, D_{\max})$ which is derived from a packing algorithm [30]. In order to avoid leaving any part of the area without service, the circles of individual UAV coverage are packed in an overlapping manner that fully covers the CA. Making each UAV $j \in \{1, 2, .. k_h\}$ equally relevant, we assign the same radius $D(k_h, D_{\max}) = D_j \ \forall j$. Thus, as per the packing provided in [30], the radius occupies discrete values $D(k_h, D_{\max}) = \frac{D_{\max}}{\gamma_{k_h}}$, where $\gamma_{k_h} =$

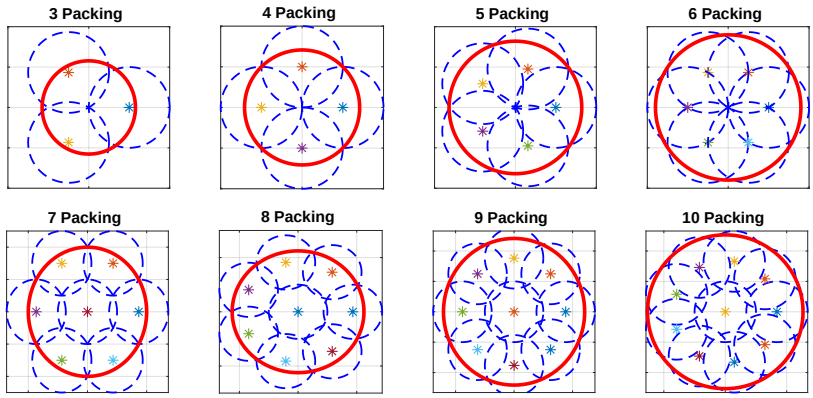


Fig. H.2: Overlapping packing patterns for UAV regions (blue) with radius $D(k_h, D_{\max})$ fully covering the CA (red) with radius $D_{\max}$.

$1, 1, 1.1547, \sqrt{2}, 1.641, 1.7988, 2,$ for $k_h = 1, 2, 3, 4, 5, 6, 7,$ respectively, and $\frac{D_{\max}}{1+2\cos\left(\frac{2\pi}{k_h-1}\right)}$ for $k_h = 8, 9, 10$.

Setting the center of the CA as the center of our coordinate system (0,0), the centers of the $k_h = \{3, 4, 5, 6\}$ circles are located at $\{\mathbf{p}_j(k_h, D_{\max}) = (x_j, y_j)\}$ where,

$$x_j = D(k_h, D_{\max})\cos\left(\frac{2\pi(j-1)}{k_h - 1}\right) \ \forall\, j \in \{1, 2, ...k_h\}\,, \tag{H.2}$$

$$y_j = D(k_h, D_{\max})\sin\left(\frac{2\pi(j-1)}{k_h - 1}\right) \ \forall\, j \in \{1, 2, ...k_h\}\,. \tag{H.3}$$

For the case of 7, the centers of the smaller circles of radius $D(7, D_{\max})$ that cover the region have coordinates $\{\mathbf{p}_j(7, D_{\max}) = (x_j, y_j)\}$ where,

$$x_j = D(7, D_{\max})\sqrt{3}\cos(\frac{2\pi(j-1)}{6}) \ \forall\, j \in \{1, 2, ...6\}\,, \tag{H.4}$$

$$y_j = D(7, D_{\max})\sqrt{3}\sin(\frac{2\pi(j-1)}{6}) \ \forall\, j \in \{1, 2, ...6\}\,, \tag{H.5}$$

$$(x_7, y_7) = (0, 0). \tag{H.6}$$

As such, the horizontal distance from the CS can be calculated as $d_j(k_h, D_{\max}) = \sqrt{x_j^2 + y_j^2}$. Finally, for $k_h = \{8, 9, 10\}$, one circle is concentric with the region and the centers of the other circles are situated in the vertices of a regular $(n-1)$-gon at a distance of $d_j(k_h) = 2\sin(\frac{\pi}{(j-1)})$ for $j \in \{1, 2, ...(k_h - 1)\}$ from the center of the region. The circle packing formations are shown in Fig. H.2. In order to achieve coverage

regions with radius $D(k_h, D_{\max})$ we adjust the UAV hovering height $H(k_h, D_{\max})$, which is dependent on the propagation environment in the CA and it is covered below.

2.2 Propagation Characteristics with a Directional Antenna

UAV based communication links discriminate two propagation groups, users with direct line-of-sight (LoS) or no-LoS (NLoS). As such, the path loss ℓ is a sum of the free space path loss (FSPL) and the additional large-scale shadowing coefficient for each one of the propagation groups. The mean large scale fading coefficients for each propagation group are η_{LoS} and η_{NLoS} and come as a consequence of the typology's features [12]. Thus, the path loss between a user at horizontal distance D and a UAV with altitude H can be expressed as:

$$\ell_{\text{LoS}} = -10\log(G_{\text{t}}) + 20\log(\sqrt{D^2+H^2}) + C + \eta_{\text{LoS}}, \tag{H.7}$$

$$\ell_{\text{NLoS}} = -10\log(G_{\text{t}}) + 20\log(\sqrt{D^2+H^2}) + C + \eta_{\text{NLoS}}, \tag{H.8}$$

where, G_{t} is the antenna gain, log is a shortened version of the common logarithm $\log_{10}$, and the term C is a substitute for the carrier frequency f_{c} constant in FSPL $C = 20\log(\frac{f_{\text{c}}4\pi}{c})$. Finally, averaging the two propagation groups by the probability of a LoS occurring gives:

$$10\log[L] = P_{\text{LoS}}(\eta_{\text{LoS}} - \eta_{\text{NLoS}}) + \ell_{\text{NLoS}}, \tag{H.9}$$

where the LoS probability is given by the s-curve model [13]:

$$P_{\text{LoS}} = \frac{1}{1 + a\exp(-b[\arctan\left(\frac{H}{D}\right) - a])}, \tag{H.10}$$

where a and b are constants dependent on the topological setting.

Each UAV has a downwards facing antenna with gain $G_{\text{t}} = A_{\text{eff}}10\log(G_{\text{I}})$, where the ideal conical antenna has gain:

$$G_{\text{I}} = \frac{2}{1 - \sin(\theta\frac{\pi}{180})}, \tag{H.11}$$

where $\theta = \arctan(\frac{H}{D})$ is the elevation angle at the cell edge and A_{eff} is the antennas' effectiveness in fitting an ideal conical beamwidth. This results in the final path loss expression:

$$10\log(L) = \frac{\eta_{\text{LoS}} - \eta_{\text{NLoS}}}{1 + a\exp\{-b[\theta - a]\}} + 20\log\left(\sqrt{D^2+H^2}\right) - A_{\text{eff}}10\log\left[\frac{2}{1 - \sin(\theta\frac{\pi}{180})}\right] + C + \eta_{\text{NLoS}}. \tag{H.12}$$

In order for all the users within the area with radius D to be served, we optimize the elevation angle of $\theta = \arctan(\frac{H}{D})$ from the perspective of a user located exactly at distance D. Thus, as in [9] we can extract an optimal ratio of D and H, through the angle θ, by solving:

$$0 = \frac{\pi \tan\left(\theta \frac{\pi}{180}\right)}{9 \log(10)} + \frac{a\, b(\eta_{\text{LoS}} - \eta_{\text{NLoS}}) \exp(-b(\theta - a))}{a \exp(-b(\theta - a) + 1)^2} - A_{\text{eff}} \frac{\pi \cos\left(\theta \frac{\pi}{180}\right)}{18 \log(10)(1 - \sin\left(\theta \frac{\pi}{180}\right))}. \quad \text{(H.13)}$$

This makes it easy to calculate the hovering height as $H = D \tan(\theta)$, which formulates the pathloss only as a function of the horizontal distance $L(D)$. Finally, the serving rate for a user at distance $D = D(k_h, D_{\max})$ becomes:

$$R(k_h, D_{\max}) = B \log_2 \left[1 + \frac{P_{\text{t}}}{B N_0 L(D(k_h, D_{\max}))}\right], \quad \text{(H.14)}$$

where P_{t} is the transmission power that is assumed to be identical at both user and UAV side, while N_0 is the noise spectral density linearly scaling the noise with the channel bandwidth B. Since the packing is done in an overlapping manner, we must account for a total available spectrum of $B_{\text{tot}} \geq 3 \cdot B$ to avoid inter-UAV-cell interference. Finally, we note that even though the coverage circles of two UAVs using the same bandwidth may overlap, such overlap occurs outside both coverage regions, and is thus not harmful towards the spectrum reuse in the packing algorithm, as it can be seen in Fig. H.2.

2.3 UAV Power Consumption Model

Most of the UAV's power consumption is absorbed by its rotors, while the power spent for communications is negligible. To hover, the UAV may have to counteract the wind speed $v_{h,i}^{\text{wind}}$, for hour h at day i, to achieve net zero speed is remarked as flying horizontally with non-zero velocity. Here we differentiate the wind intensity with regards to the daily variations since such data is readily available, and has very strict seasons. We also expect that the horizontal speed required to counteract the wind speed increases with altitude [31]:

$$v_{\text{hov},h,i} = v_{h,i}^{\text{wind}} \left[\frac{H(k_h, D_{\max})}{H_0}\right]^{E_{\text{w}}}, \quad \text{(H.15)}$$

where H_0 is the measurement altitude of the wind velocity $v_{h,i}^{\text{wind}}$, and E_{w} is the empirically measured indicator derived from the roughness of the surface in the area. To reach the hovering position $\mathbf{p}_j(k_h, D_{\max})$, the UAV ascends vertically with a velocity of v_{c} to the designated height $H(k_h, D_{\max})$, and flies horizontally with a velocity of v_{hfly} the horizontal distance $d_j(k_h, D_{\max})$. Near the end its air-time τ_{fly}, the UAV descends with $-v_{\text{c}}$ to represent negative velocity with regards to the coordinate system.

Table H.3: UAV Flight Parameters.

Label	Definition	Value
W	Weight of the UAV in Newton	23.84 N
N_{R}	Number of rotors	4
F_{n}	Upward thrust by the n^{th} rotor	-
v_{hfly}	UAV's horizontal flying velocity	10 m/s
v_{tip}	Tip speed of the rotor	102 m/s
A_{f}	Fuselage area	0.038 m^2
$\rho(H(k_h, D_{\max}))$	Air density	-
C_{D}	Drag Co-efficient	0.9
A_{r}	Rotor disc area	0.06 m^2
Δ	Profile drag coefficient	0.002
s	Rotor solidity	0.05
v_c	UAV's vertical flying velocity	10 m/s
P_{vfly}	Vertical flight power	-
P_{hfly}	Horizontal flight power	-
v_{hov}	Flying speed to counteract wind (m/s)	-

All the parameters used in the following equations are defined in Table. H.3, and with the goal to reduce equation clutter the variables $H(k_h, D_{\max})$ and $d_j(k_h, D_{\max})$ are reduced to H and d_j respectively. The power consumed by the UAV when flying horizontally with speed v is derived using the axial momentum theory, while assuming identical rotors [22] as,

$$P_{\mathrm{hfly}}(v) = \underbrace{N_{\mathrm{R}} P_{\mathrm{b}} \left(1 + \frac{3v^2}{v_{\mathrm{tip}}^2}\right)}_{P_{\mathrm{blade}}} + \underbrace{\frac{1}{2} C_{\mathrm{D}} A_{\mathrm{f}} \rho(H) v^3}_{P_{\mathrm{fuselage}}} + \underbrace{W \left(\sqrt{\frac{W^2}{4N_{\mathrm{R}}^2 \rho^2(H) A_{\mathrm{r}}^2} + \frac{v^4}{4}} - \frac{v^2}{2}\right)^{\frac{1}{2}}}_{P_{\mathrm{induce}}}, \tag{H.16}$$

where $P_{\mathrm{b}} = \frac{\Delta}{8} \rho(H) s A_{\mathrm{r}} v_{\mathrm{tip}}^3$, $\rho(H) = (1 - 2.2558.10^{-5} H)^{4.2577}$. P_{blade} and P_{fuselage} are the powers required to overcome the profile drag forces of the rotor blades and the fuselage of the aerial vehicle that oppose its forward movement, respectively, while P_{induce} represents the power required to lift the payload.

The power required by the aerial vehicle to climb vertically with a rate v_{c} m/s is

expressed as,

$$P_{\text{vfly}}(v_{\text{c}}) = \frac{W}{2}\left(v_{\text{c}} + \sqrt{v_{\text{c}}^2 + \frac{2W}{N_{\text{R}}\rho(H)A_{\text{r}}}}\right) + N_{\text{R}}P_{\text{b}}. \tag{H.17}$$

The energy consumption for the entire flight of UAV j occuring at hour h, day i, is $E_{j,h,i}(k_h, D_{\max})$ and can be thus segmented into the three parts, ascent, hovering, and descent:

$$\begin{aligned} E_{j,h,i}(k_h, D_{\max}) = & \underbrace{P_{\text{vfly}}(v_{\text{c}})\frac{H}{v_{\text{c}}} + P_{\text{hfly}}(v_{\text{hfly}})\frac{d_j}{v_{\text{hfly}}}}_{\text{ascent}} \\ & + \underbrace{P_{\text{vfly}}(-v_{\text{c}})\frac{H}{v_{\text{c}}} + P_{\text{hfly}}(v_{\text{hfly}})\frac{d_j}{v_{\text{hfly}}}}_{\text{descent}} \\ & + \underbrace{P_{\text{hfly}}(v_{\text{hov},h,i}) \cdot \left(\tau_{\text{fly}} - 2\left(\frac{H}{v_{\text{c}}} + \frac{d_j}{v_{\text{hfly}}}\right)\right)}_{\text{hover}}, \end{aligned} \tag{H.18}$$

where τ_{fly} is the designated flight time that the UAV must complete, and $2\left(\frac{H}{v_{\text{c}}} + \frac{d_j}{v_{\text{hfly}}}\right) < \tau_{\text{fly}}$. For convenience, we use a flight duration τ_{fly} of half an hour, which is reasonable for state-of-the-art UAV models, since our wind, solar and traffic data are quantized at each hour of the day. This means that at hour h on day i the UAV consumes a total energy of:

$$E_{\text{UAVs},h,i}(k_h) = \frac{1}{\tau_{\text{fly}}}\sum_{j=1}^{k_h} E_{j,h,i}(k_h). \tag{H.19}$$

Finally, we note that some of the flight time is spent on flying to and back from designated hovering positions. To avoid service outage and add leeway for battery swapping, we assume that the process of positioning occurs at different times for each UAV To afford such mobility, the system requires one spare auxiliary UAV.

3 Energy Generation and Management at the Central Unit

The electricity generated and stored in this system is proportional to its size, and therefore financial budget. The service availability detailed in the previous section thus becomes a function of the finantial budget, which is spent on energy generation and storage systems for the CS.

Load

Once the UAV lands on the CS, after spending τ_{fly} time in the air, it releases its depleted battery through an automated battery exchange system and receives a new, fully charged one, as shown back in Fig. H.1 part c). The old battery is then charged until full making each recharge cycle duration $\tau_{\text{charge}} = \frac{C_{\text{bat}}}{P_{\text{charge}}}$, where C_{bat} is the battery capacity, and P_{charge} is the charger power. These are lithium polymer (LiPo) batteries, which have a predominantly linear charging behaviour [32]. Therefore, the power drawn by a single battery unit is assumed to be constant, and the overall load profile will look like a step function directly dependent on the number of batteries recharging at the same time. The time required for each battery to be guaranteed operational for τ_{fly} must satisfy $\max\left(E_{j,h,i}(k_h, D_{\max})\right) \leq C_{\text{bat}} \;\; \forall\, j, h, i$. Where C_{bat} should be kept to a minimum with some margin for errors. Therefore, the number of UAV batteries per single UAV that are required by the system is defined by the $\tau_{\text{charge}}/\tau_{\text{fly}}$ ratio. The maximum number of UAV replaceable batteries is:

$$b_{\max} = \left\lceil n_{\text{UAV}} \left(\frac{\tau_{\text{charge}}}{\tau_{\text{fly}}} + 1 \right) \right\rceil, \tag{H.20}$$

which has to be reflected in the purchasing price per UAV in the fleet.

PV

The solar energy generation units are represented by a set of photovoltaic (PV) panels placed in parallel, all of the same type [33] and with the same working conditions. All parameters used in these equations are summarized in Table H.4. Their behaviour is simulated using a simplified version of the 5 parameters model [34], which neglects the shunt resistance and allows to calculate the maximum power voltage (V_{m}) and current (I_{m}) provided at any irradiation (G^{irr}) and ambient temperature (T_{a}) conditions. This is made possible using the conservative assumption of a maximum power point tracker (MPPT) with average efficiency $\epsilon_{\text{MPPT}} = 95\%$ [35].

$$V_{\text{m},h,i} = V_{\text{m,ST}} - \beta\left(T_{\text{C},h,i} - T_{\text{ST}}\right) + V_{\text{t},h,i} \log \frac{G^{\text{irr}}_{h,i}}{G^{\text{irr}}_{\text{ST}}}, \tag{H.21}$$

$$V_{\text{t},h,i} = n_{\text{cells}} \frac{k_{\text{b}} n_{\text{I}} T_{\text{C},h,i}}{q}, \tag{H.22}$$

$$I_{\text{m},h,i} = I_{\text{m,ST}} \left(\frac{G^{\text{irr}}_{\text{ST}}}{G^{\text{irr}}_{h,i}} \right) + \alpha\left(T_{\text{C},h,i} - T_{C,\text{ST}}\right), \tag{H.23}$$

$$T_{\text{C},h,i} = T_{\text{a},h,i} + \frac{T_{\text{C,NOC}} - T_{\text{a,NOC}}}{G^{\text{irr}}_{\text{NOC}}} G^{\text{irr}}_{h,i}. \tag{H.24}$$

Table H.4: PV Parameters from [33].

Label	Definition	Value
α	Thermal coefficient of SC current	0.0474 %/°C
β	Thermal coefficient of OC voltage	-0.285 %/°C
ϵ_{conv}	Converter efficiency	95%
n_{cells}	Number of PV cells	60
n_{I}	Diode ideality factor	1.5
k_{b}	Boltzmann constant	$1.380649 \cdot 10^{-23}$ J/K
q	Electrical charge of an electron	$1.602176634 \cdot 10^{-19}$ C
$G^{\text{irr}}_{\text{NOC}}$	Irradiation at NOC	800 W/m^2
$G^{\text{irr}}_{\text{ST}}$	Irradiation at ST	1000 W/m^2
$I_{\text{m,ST}}$	Maximum power current at ST	8.85 A
$T_{\text{a,NOC}}$	Ambient temperature at NOC	20°C
$T_{\text{C,NOC}}$	Cell temperature at NOC	45°C
$T_{\text{C,ST}}$	Ambient temperature at ST	25°C
$V_{\text{m,ST}}$	Maximum power voltage at ST	31.8 V
ϵ_{MPPT}	Maximum Power Point Tracker	95%

Knowing V_{m} and I_{m} from (H.21) and (H.23), as well as the cell temperature T_{C}, allows to calculate the output power as:

$$P_{\text{PV},h,i}(n_{\text{PV}}) = n_{\text{PV}} \cdot V_{\text{m},h,i} \cdot I_{\text{m},h,i} \cdot \epsilon_{\text{conv}} \cdot \epsilon_{\text{MPPT}}. \tag{H.25}$$

In the above equations, the subscript ST means standard test conditions ($G^{\text{irr}}_{\text{ST}} = 1000W/m^2$, $T_{\text{a,ST}} = 25°\text{C}$), whereas NOC stands for nominal operating conditions ($G^{\text{irr}}_{\text{NOC}} = 800$ W/m^2, $T_{\text{a,NOC}} = 20°\text{C}$). The cell temperature at standard test conditions T_{ST}, was calculated using (H.24), but using $T_{\text{a,SC}}$ and $G^{\text{irr}}_{\text{ST}}$ instead of T_{a} and G. The list price for a single panel, pre-VAT, is € 202 resulting in a PV system cost that scales linearly with the number of solar panels n_{PV}, as $F_{\text{PV}} = 202 \cdot n_{\text{PV}}$.

Wind

In favor of precision, the power output of a wind turbine is not calculated with an analytical model, but the interpolation of power curves found in the data sheet [36], and shown with blue in Fig. H.3. Two types of wind turbines are considered and treated as distinct elements of the system:

- A horizontal axis small-WT with standard power output of 500W (unit cost $F_{\text{W500}} = € 1,429.95$);
- A horizontal axis medium-WT with standard power output of 1kW (unit cost $F_{\text{W1000}} = € 2,738.76$);

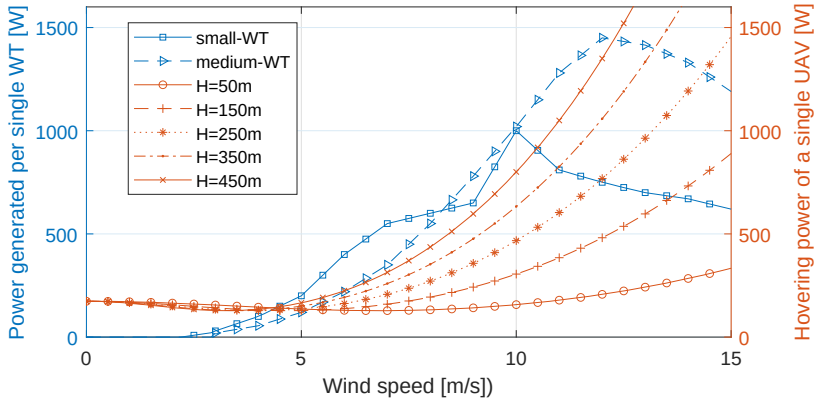


Fig. H.3: The wind power curves for two types of horizontal-axis WTs (blue), and UAV power consumption for hovering at different altitudes (red).

The list prices displayed above are pre-VAT, and were provided by Aeolos Wind Energy Ltd [36], and include a 9 m pole, as well as a rectifier and control system. Finally, the total power output of the system is scaled to:

$$P_{\mathrm{WT},h,i}(n_{\mathrm{W500}}, n_{\mathrm{W1000}}) = \sum_a n_a P_a(v_{w,h,i}), \tag{H.26}$$

where $a \in \{\mathrm{W500}, \mathrm{W1000}\}$ depicts the type of the turbine out of the two suggested ones, and n_a is the number of turbines of each type, giving a total financial cost of $F_{\mathrm{WT}} = \sum_a F_a \cdot n_a$.

Ground Battery

In order to provide continuous service, an energy storage system with capacity E_{cap} is needed. The load is powered by the generation elements when possible, with the storage elements receiving any excess energy and providing back-up when the generation rate is too low. Therefore, at the end of the timeslot the net energy in the system is $E_{\mathrm{net},h,i} = [P_{\mathrm{PV},h,i}(n_{\mathrm{PV}}) + P_{\mathrm{WT},h,i}(n_{\mathrm{W500}}, n_{\mathrm{W1000}})]\,\delta_t - E_{\mathrm{UAVs},h,i}(k)$, where $\delta_t = 2\tau_{\mathrm{fly}}$ is the length of the time interval. Thus, in case of net positive or negative energy, the battery capacity at the next time step $E_{(h+1)\%24,i+(h+1)/24}$ (where % is the modulo operator and / is integer division) will increase or decrease by $\min(E_{\mathrm{cap}}(n_{\mathrm{cell}}), E_{h,i} + \epsilon_{\mathrm{b},h,i} E_{\mathrm{net},h,i})$, with,

$$\epsilon_{\mathrm{b},h,i} = \begin{cases} \epsilon_{\mathrm{conv}}, & E_{\mathrm{net},h,i} \geq 0 \\ \frac{1}{\epsilon_{\mathrm{conv}}}, & E_{\mathrm{net,h,i}} < 0 \end{cases} \tag{H.27}$$

where, E_{cap} is the total capacity of the battery as a function of the number of cells in the system n_{cell}, and $\epsilon_{\mathrm{b},h,i}$ is the overall efficiency of the storage system. The Li-ion

battery cells are cylindrical LG MJ1, with unit cost of € 5.75 and capacity of 12.6 Wh, resulting in a maximum ground battery (GB) capacity of $E_{\text{cap}} = 12.6 \cdot n_{\text{cell}}$ requires spending $F_{\text{E}} = 5.75 \cdot n_{\text{cell}}$.

4 Problem Definition and Methodology

The goal of this paper is to find the best value for providing as much coverage as possible in a geographical region. We thus consider the problem of sizing the entire system as a combination of six variables: 1) number of communication UAVs in the fleet n_{UAV}; 2) number of 500W WTs n_{W500}; 3) number of 1kW WTs n_{W1000}; 4) number of PV panels n_{PV}; 5) number of battery cells in the GB n_{cell}; and 6) the radius of the circular CA $D_{\max}$. In order to evaluate the quality of the system, we use the area $\pi D_{\max}^2$, in which the guaranteed communications rate is satisfied, and the total upfront cost F for the system needed to provide that service.

Since accurately calculating the total capital expenditure of the system is crucial, the UAV swarm cost plays a big role. Since we use the DJI matrice 100/200 models as a reference, we take a reference price of € 4000 per UAV, resulting in a total cost for UAV equipment of $F_{\text{UAV}} = 4000 \cdot n_{\text{UAV}}$. Moreover, this budget also covers spare batteries $b_{\max} = 3 \cdot n_{\text{UAV}}$ that are required for battery swapping. Finally, to guarantee operability in case of defects in one of the UAVs in the fleet, and to offer better interleaving for battery swapping [22], there needs to be one spare UAV in the fleet $n_{\text{UAV}} \geq 2$. To avoid inconsistencies in the service, a simple timing difference in the UAV swapping time can be employed. To elaborate, all UAVs do not have to do the battery swap at the same exact instant which would result in a total outage of the system. To circumvent this issue, a desynchronization of UAV deployment by a few minutes is used. Finally, in case of a scheduling failure, the redundant UAV can substitute the designated UAV in the air.

$$
\begin{aligned}
\text{(P1)}: \quad & \underset{\{n_{\text{PV}}, n_{\text{W500}}, n_{\text{W1000}}, n_{\text{cell}}, n_{\text{UAV}}, D_{\max}\}}{\text{maximize}} \quad \frac{\pi D_{\max}^2}{F}, \\
\text{s.t.} \quad & R(k_h, D_{\max}) \leq R_{\min,h}(D_{\max}), && \text{(H.28)} \\
& 11 \geq n_{\text{UAV}} \geq 2, && \text{(H.29)} \\
& n_{\text{UAV}} \geq \max_h (k_h) + 1, && \text{(H.30)} \\
& E_{h,i} \geq 0, && \text{(H.31)} \\
& F = F_{\text{PV}} + F_{\text{WT}} + F_{\text{UAV}} + F_{\text{E}} \leq F_{\max}, && \text{(H.32)} \\
& D_{\max} \geq D_{\text{lb}}, && \text{(H.33)} \\
& D_{\max} \leq D_{\text{ub}}. && \text{(H.34)}
\end{aligned}
$$

The (P1) objective function maximizes the coverage of the deployment normalized by its CAPEX; (H.28) guarantees the quality of service for the whole area; (H.29) maintains eligibility of the number of UAVs in the swarm; (H.30) defines the size of the swarm; (H.31) guarantees no system outage due to lack of energy; (H.32) defines the financial budget; (H.33) and (H.34) define the minimum and maximum required coverage. We note that if the problem is infeasible, the system is inadequate for the application scenario. Finally, as per (H.28) and (H.31) the system does not allow for any outage tolerance given the provided data. This is because mismanagement of energy allocation will not result in a total outage but a reduced quality of service. Since our goal is to provide average service to the most users, accounting for outages would not be aimed towards constraining outages for that hour altogether, but a separate problem of QoS maximization instead of coverage maximization. So, in cases of sub-average performance of the system, it will operate in a best-effort mode.

Given a fixed coverage area, the problem can be separated into two sub-problems that construct the CAPEX efficient coverage maximization, and thus solution-searching can be done iteratively. The easier problem of the two is searching for the minimum energy load (MEL).

$$\begin{aligned}
(\text{MEL}) : \underset{\{k_h\}}{\text{minimize}} \quad & \min_{k_h} E_{\text{UAVs},h,i}(k_h, D_{\max}) \ \forall\, h, i, \\
\text{s.t.} \quad & R(k_h, D_{\max}) \leq R_{\min,h}(D_{\max}), && \text{(H.35)} \\
& 10 \geq k_h \geq 1. && \text{(H.36)}
\end{aligned}$$

The MEL problem guarantees coverage for a specific area by satisfying the lower datarate bound. Extracting the load profile of the entire system for a single area size is a constant complexity operation, since the number of hours and days for coverage are fixed. As such, the search of the entire space of eligible coverage areas has linear complexity where the complexity of that operation scales with the size of the eligible space between both boundaries (H.28) and (H.29).

The second sub-problem is finding the cheapest combination of energy elements (CCEE) W500 and W1000 WTs, PVs, and battery cells that satisfies the load profile. An exhaustive search on the CCEE problem has quartic complexity, which, summed with an exhaustively searched MEL, create an unreasonably complex problem. In addition, checking the MEL + CCEE sub-problems for every possible coverage multiplies the complexity by the size of the space $D_{\text{lb}} \leq D_{\max} \leq D_{\text{ub}}$. Therefore, it is necessary to find a more efficient way to solve (P1). We approach this by performing greedy sparse search to reduce the solution space, and simpler algorithms to find near-optimal solutions. Approximate methods, such as Genetic Algorithms implemented in [37], did not yield a satisfying performance and are thus left out of this work. However, the computational performance of GA is given at the end of Section 5.

To elaborate better, there are many challenges that come from the solving the realistic design for a combination of a UAVs swarm supplied with unreliable energy such

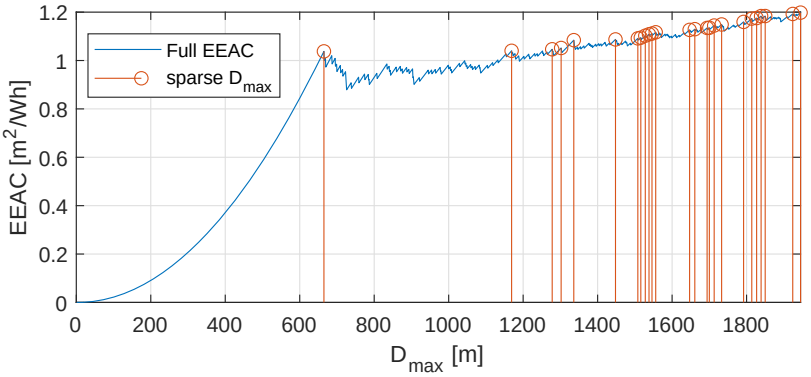


Fig. H.4: EEAC for servicing coverage area, at a suburban/remote setting, in the presence of weak wind with expected velocity of 3.6 m/s, overlay.

as renewables, in particular wind. First challenge is that the swarm has a varying size during the day. The second challenge is that the energy expenditure has non-monotone relationship the wind intensity. The third challenge is that the energy generation also varies in a non-monotone manner with the wind intensity. However, the biggest challenge of all is that all aforementioned challenges do not scale linearly with the size of the coverage area. Thus solving the optimal combination problem for all possible sizes of cellular coverage is not tractable. Nonetheless, we developed an efficient way to find the most economical system configuration based on the service demand and the available resources whose data is region-specific and is obtained from European Commission's information system. The methods are elaborated in the following subsection.

4.1 Greedy and Sparse Search (GSS) Algorithm

We define a search algorithm that uses sparse searching of coverage areas where maximum coverage per unit cost is likely to occur, and uncover a simplified way to solve CCEE. Specifically, we investigate the energy efficiency of annual coverage (EEAC) for each size of coverage area as a proxy-heuristic metric:

$$\mathrm{EEAC} = \frac{\pi D_{\max}^2}{\sum_{h,i} E_{\mathrm{UAVS},h,i}}, \tag{H.37}$$

where $E_{\mathrm{UAVS},h,i}$ is given by the MEL problem. Looking at Fig. H.4, it is noticeable that EEAC is not a monotonic nor a convex function of the coverage area. Therefore, it is convenient to sparsely search for a solution where EEAC is improving. Furthermore, we can use a greedy approach to shrink the number of samples that will be searched for a solution to the ones that offer the best improvement with regards to the last sample. Thus we select only the samples whose second order derivative is larger than zero. In this

Algorithm 4: GSS

IMPORT: {MEL, BINARY-SEARCH,
SAMPLE-mono, SAMPLE-2ndder , SAMPLE-comb}
Input: all-constants, all-data, D_{lb}, D_{ub} max_budget;
j=0
$D_{\max} = D_{\text{lb}}$
step_size=1;
while $D_{max} \leq D_{ub}$ *&&* $k_h \leq n_{UAV}$ **do**
  j=j+1
  ($\text{load}_{h,i}$ [j] ,F_{UAV} [j])
  =**MEL**($E_{\text{UAVs},h,i}(k_h, D_{\max})\ \forall\, h, i$)
  EEAC [j]= $\pi D^2_{\max} / \sum_{h,i} \text{load}_{h,i}$ [j]
  $D_{\max} = D_{\max}$ + step_size
SAMPLE-mono (EEAC,EEAC*):
$\text{EEAC}_{\text{mnt}} \leftarrow$ monotonic samples
SAMPLE-2ndder (EEAC_{mnt},EEAC*):
$D_{\text{max\_sparse}}, \text{addr} \leftarrow$ positive 2$^{\text{nd}}$ derivatives
j=0
for D_{max} **in** D_{max_sparse} **do**
  j=j+1
  $\text{load}_{h,i} = \text{load}_{h,i}$ [addr[j]]
  $F_{\text{UAV}} = F_{\text{UAV}}$ [addr[j]]
  $F_{\text{comb}} = 0$
  $F =$ max_budget
  solutions = []
  flag==True
  while *flag==True* **do**
    $(n_{\text{PV}}, n_{\text{W500}}, n_{\text{W1000}}), \text{flag} \leftarrow$ **SAMPLE-comb**
    $F_{\text{comb}} = F_{\text{PV}} + F_{\text{WT}} + F_{\text{UAV}}$
    $n_{\text{cell}} = \lfloor (F - F_{\text{comb}})/5.75 \rfloor$
    if $E_{h,i}(n_{cell}, n_{PV}, n_{W500}, n_{W1000}, load) \geq 0$ **then**
      BINARY-SEARCH$_{\text{minimize}}\ n_{\text{cell}}$ **s.t.**
        $E_{h,i}(n_{\text{cell}}, n_{\text{PV}}, n_{\text{W500}}, n_{\text{W1000}}, \text{load}) \geq 0$
      $F_{\text{comb}} = F_{\text{PV}} + F_{\text{WT}} + F_{\text{UAV}} + F_{\text{E}}$
      APPEND $(D_{\max}, n_{\text{cell}}, n_{\text{PV}}, n_{\text{W500}}, n_{\text{W1000}}, n_{\text{UAV}}, F_{\text{comb}})$
      TO solutions
      $F = F_{\text{comb}}$
  fin_sols[j] = $\min_{F_{\text{comb}}}$(solutions)
Output: fin_sols

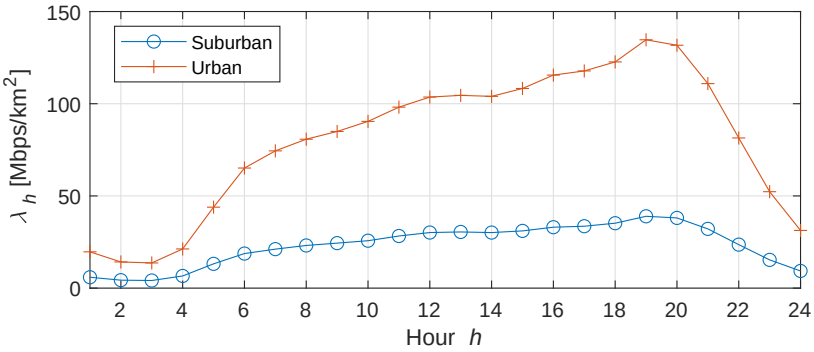


Fig. H.5: Daily evolution of requested data traffic [5, 29].

way, we still solve the MEL problem for the whole $D_{\text{lb}} \leq D_{\max} \leq D_{\text{ub}}$ space beforehand, with the goal of reducing the search space for the multi-variate CCEE sub-problem by a significant factor ranging between 100-1000, depending on the scenario we investigate.

The CCEE problem is harder to simplify. However, we can exploit the fact that the budget can be dedicated to two different purposes: energy generation and storage. We can easily reduce the complexity of searching the viability of storage once we have sufficient energy generation supporting the system. Thus, we decrease the dimensionality of the space by increasing the budget until battery storage becomes relevant, i.e. the power generation profile is able to keep up with the load profile. And since we are looking to minimize the financial cost, the first eligible solution where battery storage becomes relevant becomes our new and smaller search space. A simplified representation of the algorithm is shown in Alg. 4. The GSS approach does not guarantee to always find the global maximum for the coverage area due to the sampling of $D_{\max}$. Despite this, GSS managed to find the global optimum for all the scenarios that we tested. This is mostly due to the well sampled areas and the exhaustive search **SAMPLE-comb** function for sampling combinations of WTs, PVs, that are monotonously increasing in cost.

5 Numerical Results and Case Analysis

We aim to accurately evaluate the feasibility of the system across seasons or years. However, due to data sensitivity, the traffic data requested by the populace is only reflected on a daily cycle λ_h and does not vary with location, as shown in Fig. H.5. Moreover, we distinguish two possible types of areas that may need coverage: Suburban and Urban. These have the $(a, b, \eta^{\text{LoS}}, \eta^{\text{NLoS}})$ propagational parameters of values $(4.88, 0.43, 0.2, 24)$ and $(9.61, 0.16, 1.2, 23)$, for Suburban and Urban respectively [12]. The rest of the testing parameters are included in Table H.5. Antenna directivity A_{eff} is considered as a split variable, since it may impact the aerodynamics of the UAV in ways that the power

Table H.5: Simulation Parameters [38, 39]

Label	Definition	Value
f_c	Channel carrier frequency	5.8 GHz
c	Velocity of light	$3 \cdot 10^8$ m/s
B	Channel bandwidth	80 MHz
H_0	Nominal height for wind measurements	10 m
E_w	Environment roughness coefficient	0.335
B_{tot}	Available spectral width	480 MHz
N_0	Noise spectral power	-174 dBm/Hz
P_t	Transmission Power	23 dBm
D_{lb}	Lower bound of coverage size	0 m
D_{ub}	Upper bound of coverage size	∞ m
P_{charge}	Power of the charger	180 W
F_{max}	Total expendable budget	€ 100000

consumption model cannot predict, and a system integrator may only have few types available.

For the case analysis, four testing locations with diverse wind speed and solar irradiation patterns were chosen. Two locations are in regions that are prone to grid and system failures, like the earthquake ridden region around the Italian town of Amatrice and the fjord/floodplains of Western Denmark. We also suggest the placement of the system in common off-grid locations, such as sparsely populated areas in Western Texas and the touristic region of the Faroe Islands. We also refer to the Faroe Islands and Western Denmark as windy locations, and Amatrice and Western Texas as sunny locations.

In Fig. H.6 we illustrate the EEAC across the four scenarios for four different antenna directivity coefficients A_{eff} that have negligible impact to the UAVs' aerodynamics. More efficient antennas expect higher optimal altitudes, thus consuming more energy for vertical flights and expecting higher wind velocities. Additionally, larger $D_{\max}$ expects larger swarms, that have a lower flying altitude. As discussed in the previous section, thanks to the aerodynamics of the UAV, the presence of small horizontal wind reduces the power consumption for hovering and proves beneficial for the flying swarm. This effect provides interesting results in the case of the more windy locations, such as the Faroe Islands, which present a distinct improvement in energy efficiency for the coverage of areas with $3500\,\text{m} \leq D_{max} \leq 4250\,\text{m}$.

All implementations are tested for full annual service on a specified location. This is the most difficult test for the system as it considers diverse weather patterns of all four seasons. The productivity of each energy source, wind harvested by WTs and solar harvested by PV panels, is tied to the geographical location of the CS and the time of year. Therefore, we use measurement-driven data provided by the European Commision's

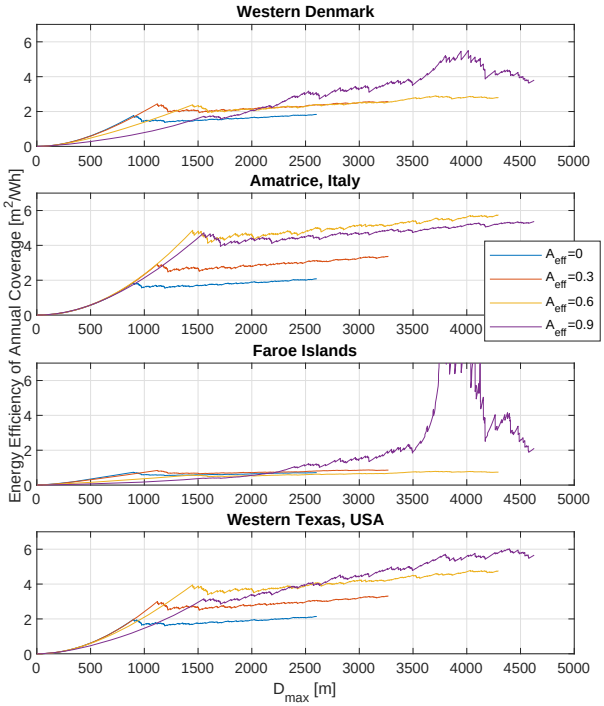


Fig. H.6: EEAC for all four locations in a suburban environment.

Photovoltaic Geographical Information System[1] to extract the annual measurements of 2015. In this way, by avoiding multi-year averages correlation between hourly samples is retained. Moreover, we expect solar irradiation and wind speed to be inversely correlated, as per the study in [40].

It is important to note that the reliability of the system entirely depends on the data taken in the testing system. In this work, we use four different sets of statistical data points, wind intensity, solar irradiation, temperature, data traffic pattern, based on averages over an hour. Thus, the solutions provided by our analysis will thus give average estimates on communication service performance. This does not mean that an unexpected fluctuation will result in total system failure but only below average performance in the communication service for that timeslot. In the same way, unexpected flux in energy supply will make the system capable of providing better service than needed, and therefore provide above average performance. These fluctuations in service are to be expected in an off-grid system, and the only way to guarantee better service is to provide stricter sampling (i.e. 75th percentile) of the dataset.

[1] https://re.jrc.ec.europa.eu/pvg_tools/en/#MR

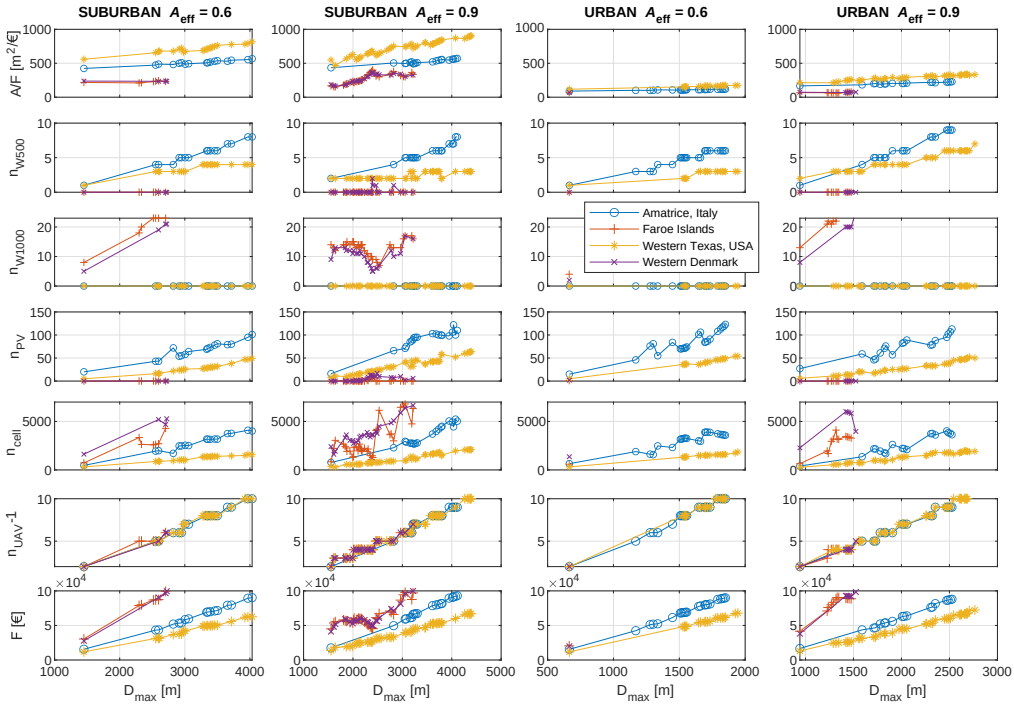


Fig. H.7: Implementation feasibility of the off-grid system in a suburban or urban environment.

Since we have set the bound for CA as infinite, in Fig. H.7 we plot the entire solution space of eligible area CA sizes searched by the GSS. This is done to better illustrate how the cost efficiency varies when different coverage bounds are imposed. The first impressions are that there is an obvious advantage of having the setup in an area where the use of solar panels is feasible. In both sunny locations, the cost-feasibility of the system is improving by better allocating the available budget, which in turn allows for exploiting the improved packing efficiency when using bigger swarms. The most cost efficient deployments are found in the Texian planes where energy can be captured through both wind and solar. Namely, bigger deployments in this setting do not need as much PV panels nor wind turbines as the other three locations, to satisfy the energy requirements of the UAV swarm.

Analyzing the impact of wind, we find that deployments in windy locations tend to have more volatile solution space opposed to the two sunny locations. The fluctuations in the curves as the CA gets larger are due to three non trivial effects of wind onto the energy generation and expenditure of the entire system. The first effect is the combination of the three non-monotonous curves for energy generation/expenditure in

Fig H.3. The second effect that causes the strong fluctuations are the variability in wind speed for energy generation, which in some cases warrants use of solar panels to satisfy average QoS. The final effect is that denser deployments (when CA is larger it requires a bigger UAV swarm) tend to have a lower flying altitude. This is particularly impactful in windy situations as the wind gets logarithmically stronger with the altitude of flying, and thus increases the energy expenditure of the swarm.

Furthermore, both windy locations do not find use for PV panels and generally use the bigger 1000W turbines. The first reason is that wind turbines offset the added wind expenditure of UAV hovering in high winds, and secondly, it makes more sense to exploit the natural resource with the higher energy output. We note that, the behaviour between W500 and W1000 WTs is in fact not linear (it has a specific profile) and W1000 can operate very efficiently in higher winds, as it shown back in Fig. H.3. To add, the UAV energy expenditure initially dips for weaker winds, as it negates negative effects from poor aerodynamics. This means that W1000 turbines are better suited for cancelling the high energy expenditure of winds stronger than 10 m/s. Even in the cases of sunny deployments, some energy captured from wind power is necessary usually with the W500 WTs. This is done to offset the higher energy expenditure in scenarios where the wind-speed becomes more challenging, such as when hovering at higher altitudes. As such, the flying altitude is the main culprit for the comparative efficiency between windy and sunny locations when the altitude of the UAVs becomes high, and wind speeds become the main cause of battery exhaustion.

The analytic impact of service in urban and suburban situations vary due to two effects. The first is the difference in the propagation properties of the environment that impact the large scale fading. Tougher propagation environments such as the urban environment have more solid structures and thus expect higher hovering altitudes. And the second effect is that, urban-type environments need to serve larger expected data traffic as represented in H.5, which results in smaller coverage areas for same swarm size.

As we can see in Fig. H.7 the suburban results are not so drastically different in efficiency (between the sunny and windy locations) in the case where the UAVs have antennas with high directivity $A_{\text{eff}} = 0.9$. In accord, deployment efficiency between the windy and the sunny locations is much closer, mostly due to the higher altitudes of the UAVs. Nonetheless, UAV swarm deployments are more costly in windy locations and generally require much higher budget than the € 100000 to achieve coverages bigger than 2700 m or 3200 m when using $A_{\text{eff}} = 0.6$ or $A_{\text{eff}} = 0.9$ respectively. We note that this is an already high cost and thus we do not recommend the use of UAV swarm wireless communications for covering big windy areas. In contrast, the scenario in Western Texas is not limited by the upper CAPEX limit, due to the mild wind presence and reliable solar irradiation.

Moving over to the Urban environment, we can see that due to the increased user density and the worse propagation environment, the coverage for the urban implemen-

tations usually have much smaller CA of radius between $2000 - 2700$ m. Moreover, the implementations' efficiency between the windy and sunny locations are much closer. However, the budget of scaling the system to a bigger coverage area in a windy location, takes a great toll on CAPEX and exceeds the initial allocation of $F = €\,100\,000$. Thus, serving windy urban areas sees no use for the case of $A_{\text{eff}} = 0.6$. Again, as seen in Fig. H.7 the performance for the windy location improves when using better antennas and allows for bigger swarms.

Finally, we compared the performance of our custom made GSS algorithm to an exhaustive search solution and a GA implementation. The performance comparison is in terms of processing time on the same machine with CPU execution, where the CPU was Intel(R) Xeon(R) Silver 4208 CPU @ 2.10GHz. The processing time for the case analysis of Western Texas was 51.88 minutes for GSS. Unrestrained exhaustive search solution containing the same solution space resolution as GSS was calculated to take around an unreasonable 3.64 years. A more practical, reduced resolution exhaustive search that was used to verify the GSS results took roughly 14 days of computation. And finally, a comparison with a GA with the following parameters was tested: population per generation of 30, number of parents in every generation 5, number of mutating offspring (in addition to the rest) 15, minimum consecutive generations when goal is reached 5, and minimum improvement desired by user 0.001 (the algorithm stops after no improvement higher than 0.001 has been made). The GA approach took 3.1 days of computation while providing poor results that were very far from the optimal values.

6 Conclusion

In this paper we considered deploying a UAV swarm that offers persistent wireless services in an entirely off-grid setting. We formulated the problem as CAPEX efficient coverage area maximization, which is a multi-variate optimization problem for solving the load profile based on real world data. We considered energy generation from two sources, wind and solar, that are also taken from real world data. In this paper we have emphasized the importance of accounting for the impact of wind onto the deployment. Moreover, we consider the hourly wind intensity as a function of elevation and terrain roughness, and account for its impact on UAV deployments for long duration hovering. We have proposed the GSS algorithm that is computationally easy, combining greedy sampling and binary search to find the optimal combination of wind turbines, PV panels, cells in the ground battery, and UAVs in the swarm. Using GSS we have calculated the feasibility of the system in four different locations where the deployment would have to balance wind or solar power generation. This work opens a plethora of directions for future works such as investigating the feasibility in very specific areas, specific short term periods, and different types of UAVs.

References

[1] M. Mozaffari, W. Saad, M. Bennis, Y. Nam, and M. Debbah, "A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems," *IEEE Communications Surveys Tutorials*, vol. 21, no. 3, pp. 2334–2360, Mar. , 2019.

[2] A. Fotouhi, H. Qiang, M. Ding, M. Hassan, L. G. Giordano, A. Garcia-Rodriguez, and J. Yuan, "Survey on UAV Cellular Communications: Practical Aspects, Standardization Advancements, Regulation, and Security Challenges," *IEEE Communications Surveys Tutorials*, vol. 21, no. 4, pp. 3417–3442, Mar. , 2019.

[3] M. Erdelj and E. Natalizio, "UAV-Assisted Disaster Management: Applications and Open Issues," in *2016 International Conference on Computing, Networking and Communications (ICNC)*, 2016, pp. 1–5.

[4] S. Sekander, H. Tabassum, and E. Hossain, "Multi-Tier Drone Architecture for 5G/B5G Cellular Networks: Challenges, Trends, and Prospects," *IEEE Communications Magazine*, vol. 56, no. 3, pp. 96–103, 2018.

[5] I. Donevski, G. Vallero, and M. A. Marsan, "Neural Networks for Cellular Base Station Switching," in *Proc. of IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops*, Paris, France, Apr. , 2019, pp. 738–743.

[6] A. Fotouhi, M. Ding, and M. Hassan, "Dynamic Base Station Repositioning to Improve Performance of Drone Small Cells," in *Proc. of IEEE Globecom Workshops (GC Wkshps)*, Washington DC, Dec. , 2016, pp. 1–6.

[7] B. Galkin, J. Kibiłda, and L. A. DaSilva, "A Stochastic Model for UAV Networks Positioned Above Demand Hotspots in Urban Environments," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 7, pp. 6985–6996, Jul. 2019.

[8] M. Mozaffari, A. Taleb Zadeh Kasgari, W. Saad, M. Bennis, and M. Debbah, "Beyond 5G With UAVs: Foundations of a 3D Wireless Cellular Network," *IEEE Transactions on Wireless Communications*, vol. 18, no. 1, pp. 357–372, Jan. 2019.

[9] I. Donevski and J. J. Nielsen, "Dynamic Standalone Drone-Mounted Small Cells," in *Proc. of European Conference on Networks and Communications (EuCNC)*, Dubrovnik, Croatia, Sep. , 2020, pp. 342–347.

[10] I. Donevski, J. J. Nielsen, and P. Popovski, "Standalone Deployment of a Dynamic Drone Cell for Wireless Connectivity of Two Services," in *2021 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2021, pp. 1–7.

[11] M. Alzenad, A. El-Keyi, F. Lagum, and H. Yanikomeroglu, "3-D Placement of an Unmanned Aerial Vehicle Base Station (UAV-BS) for Energy-Efficient Maximal Coverage," *IEEE Wireless Communications Letters*, vol. 6, no. 4, pp. 434–437, Aug. , 2017.

[12] A. Al-Hourani, S. Kandeepan, and A. Jamalipour, "Modeling Air-to-Ground Path Loss for Low Altitude Platforms in Urban Environments," in *Proc. of IEEE Global Communications Conference*, Austin, TX, Oct. , 2014, pp. 2898–2904.

[13] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP Altitude for Maximum Coverage," *IEEE Wireless Communications Letters*, vol. 3, no. 6, pp. 569–572, Dec. , 2014.

[14] N. Babu, K. Ntougias, C. B. Papadias, and P. Popovski, "Energy Efficient Altitude Optimization of an Aerial Access Point," in *Proc. of IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications*, London, Sep. , 2020, pp. 1–7.

[15] N. Babu, C. B. Papadias, and P. Popovski, "Energy-Efficient 3-D Deployment of Aerial Access Points in a UAV Communication System," *IEEE Communications Letters*, vol. 24, no. 12, pp. 2883–2887, Aug. , 2020.

[16] J. Scherer and B. Rinner, "Persistent Multi-UAV Surveillance with Energy and Communication constraints," in *2016 IEEE International Conference on Automation Science and Engineering (CASE)*, 2016, pp. 1225–1230.

[17] B. Michini, T. Toksoz, J. Redding, M. Michini, J. How, M. Vavrina, and J. Vian, "Automated Battery Swap and Recharge to Enable Persistent UAV Missions," in *Infotech@ Aerospace 2011*, Jun, 2011, p. 1405.

[18] X. Zhang and L. Duan, "Optimal Patrolling Trajectory Design for Multi-UAV Wireless Servicing and Battery Swapping," in *2019 IEEE Globecom Workshops (GC Wkshps)*, 2019, pp. 1–6.

[19] D. S. Lakew, W. Na, N.-N. Dao, and S. Cho, "Aerial Energy Orchestration for Heterogeneous UAV-Assisted Wireless Communications," *IEEE Systems Journal*, pp. 1–12, May 2021.

[20] J. Galán-Jiménez, E. Moguel, J. García-Alonso, and J. Berrocal, "Energy-Efficient and Solar Powered Mission Planning of UAV Swarms to Reduce the Coverage Gap in Rural Areas: The 3D Case," *Ad Hoc Networks*, vol. 118, p. 102517, Jul. 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S157087052100072X

[21] T. Li, K. Ota, T. Wang, X. Li, Z. Cai, and A. Liu, "Optimizing the Coverage via the UAVs with Lower Costs for Information-Centric Internet of Things," *IEEE Access*, vol. 7, pp. 15 292–15 309, Feb. 2019.

[22] N. Babu, M. Virgili, C. B. Papadias, P. Popovski, and A. J. Forsyth, "Cost- and Energy-Efficient Aerial Communication Networks with Interleaved Hovering and Flying," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 9, pp. 9077–9087, Jul. 2021.

[23] G. Piro, M. Miozzo, G. Forte, N. Baldo, L. A. Grieco, G. Boggia, and P. Dini, "HetNets Powered by Renewable Energy Sources: Sustainable Next-Generation Cellular Networks," *IEEE Internet Computing*, vol. 17, no. 1, pp. 32–39, Feb. 2013.

[24] M. A. Marsan, G. Bucalo, A. Di Caro, M. Meo, and Y. Zhang, "Towards Zero Grid Electricity Networking: Powering BSs with Renewable Energy Sources," in *2013 IEEE International Conference on Communications Workshops (ICC)*, 2013, pp. 596–601.

[25] L. Chiaraviglio, L. Amorosi, N. Blefari-Melazzi, P. Dell'Olmo, A. Lo Mastro, C. Natalino, and P. Monti, "Minimum Cost Design of Cellular Networks in Rural Areas With UAVs, Optical Rings, Solar Panels, and Batteries," *IEEE Transactions on Green Communications and Networking*, vol. 3, no. 4, pp. 901–918, Aug. 2019.

[26] L. Amorosi, J. Puerto, and C. Valverde, "Coordinating Drones with Mothership Vehicles: The Mothership and Multiple Drones Routing Problem with Graphs," 2021.

[27] K. Fujii, K. Higuchi, and J. Rekimoto, "Endless Flyer: A Continuous Flying Drone with Automatic Battery Replacement," in *2013 IEEE 10th International Conference on Ubiquitous Intelligence and Computing and 2013 IEEE 10th International Conference on Autonomic and Trusted Computing*, 2013, pp. 216–223.

[28] B. Soret, I. Leyva-Mayorga, S. Cioni, and P. Popovski, "5G Satellite Networks for Internet of Things: Offloading and Backhauling," *International Journal of Satellite Communications and Networking*, vol. 39, no. 4, pp. 431–444, Jul. 2021.

[29] A. P. Couto da Silva, D. Renga, M. Meo, and M. Ajmone Marsan, "The Impact of Quantization on the Design of Solar Power Systems for Cellular Base Stations," *IEEE Transactions on Green Communications and Networking*, vol. 2, no. 1, pp. 260–274, Oct, 2018.

[30] G. F. Tóth, "Thinnest Covering of a Circle by Eight, Nine, or Ten Congruent Circles," *Combinatorial and computational geometry*, vol. 52, no. 361, p. 59, 2005.

[31] *Wind Characteristics and Resources*. John Wiley & Sons, Ltd, 2009, ch. 2, pp. 23–89. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119994367.ch2

[32] S. Alessandrini, E. Rizzuto, and Z. Del Prete, "Characterizing Different Types of Lithium Ion Cells with an Automated Measurement System," *Journal of Energy Storage*, vol. 7, pp. 244–251, aug 2016.

[33] Futurasun, 280wp, polycrystalline photovoltaic module, fu 280p. [Online]. Available: https://www.futurasun.com/wp-content/uploads/2020/10/2020_FuturaSun_60p_260-285W_en.pdf?x97762

[34] A. Chouder, S. Silvestre, N. Sadaoui, and L. Rahmani, "Modeling and Simulation of a Grid Connected PV System Based on the Evaluation of Main PV Module Parameters," *Simulation Modelling Practice and Theory*, vol. 20, no. 1, pp. 46–58, jan 2012.

[35] M. Valentini, A. Raducu, D. Sera, and R. Teodorescu, "PV Inverter Test Setup for European Efficiency, Static and Dynamic MPPT Efficiency Evaluation," in *2008 11th International Conference on Optimization of Electrical and Electronic Equipment*, 2008, pp. 433–438.

[36] Aeolos wind energy ltd. [Online]. Available: http://www.windturbinestar.com

[37] M. Virgili, A. J. Forysth, and P. James, "A Multi-Objective Genetic Algorithm Methodology for the Design of Standalone Energy Systems," in *2021 IEEE Design Methodologies Conference (DMC)*, Jul. 2021, pp. 1–6.

[38] C. She, C. Liu, T. Q. Quek, C. Yang, and Y. Li, "Ultra-Reliable and Low-Latency Communications in Unmanned Aerial Vehicle Communication Systems," *IEEE Transactions on Communications*, vol. 67, no. 5, pp. 3768–3781, May. , 2019.

[39] I. Donevski, N. Babu, J. J. Nielsen, P. Popovski, and W. Saad, "Federated Learning with a Drone Orchestrator: Path Planning for Minimized Staleness," *IEEE Open Journal of the Communications Society*, vol. 2, pp. 1000–1014, Apr. 2021.

[40] J. Widén, "Correlations Between Large-Scale Solar and Wind Power in a Future Scenario for Sweden," *IEEE transactions on sustainable energy*, vol. 2, no. 2, pp. 177–184, Apr. 2011.

ISSN (online): 2446-1628
ISBN (online): 978-87-7573-846-5

AALBORG UNIVERSITY PRESS